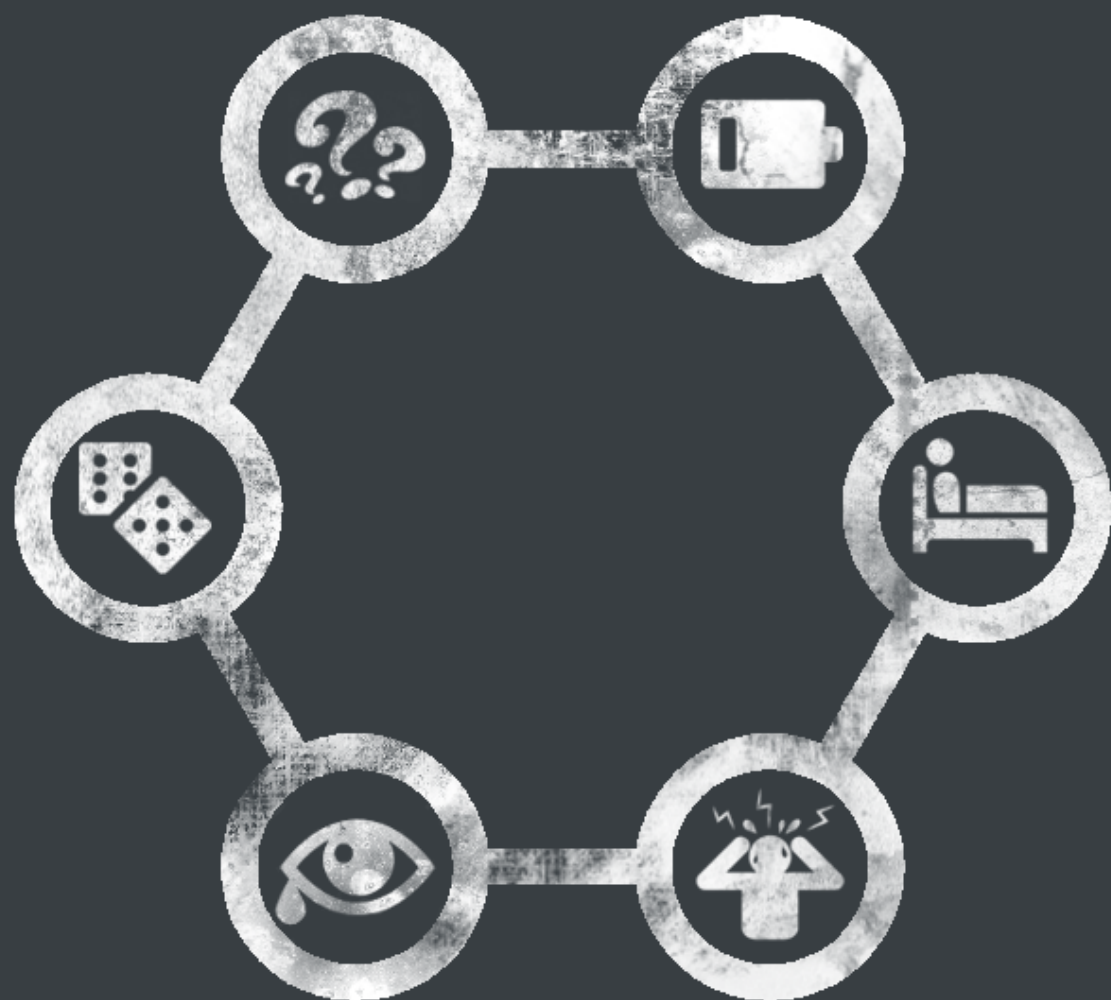# Network Psychometrics

Sacha Epskamp

Network Psychometrics

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor

aan de Universiteit van Amsterdam

op gezag van de Rector Magnificus

prof. dr. ir. K.I.J. Maex

ten overstaan van een door het College voor Promoties ingestelde

commissie, in het openbaar te verdedigen in de Agnietenkapel

op woensdag 5 April 2017, te 14:00 uur

door Sacha Epskamp

geboren te Blaricum

Promotiecommissie

In loving memory of

Janneke de Kort

26/01/1984 - 25/05/2016

Zara Bamdad

19/03/1990 - 15/07/2016

Contents

Introduction: Psychological Networks

1.1 Introduction

There are over 7 billion people in the world, each with a different brain containing 15 to 33 billion neurons. These people are intelligent entities who develop and change over time and who interact with each other in complicated social structures. Consequently, human behavior is likely to be complex. In recent years, research on dynamical systems in psychology has emerged, which is analogous to other fields such as biology and physics. One popular and promising line of research involves the modeling of psychological systems as causal systems or networks of cellular automata (Van Der Maas et al., 2006; Borsboom, 2008; Cramer, Waldorp, van der Maas, & Borsboom, 2010; Borsboom, Cramer, Schmittmann, Epskamp, & Waldorp, 2011). The general hypothesis is that noticeable macroscopic behavior—the co-occurrence of aspects of psychology such as cognitive abilities, psychopathological symptoms, or behavior—is not due to the influence of unobserved common causes, such as general intelligence, psychopathological disorders, or personality traits, but rather to emergent behavior in a network of interacting psychological, sociological, biological, and other components. I will term such networks *psychological networks* to distinguish these models from other networks used in psychology, such as social networks and neural networks.

Figure 1.1 shows an example of such a psychological network, estimated on the `bfi` dataset from the *psych* R package (Revelle, 2010). This dataset contains the responses of 2,800 people on 25 items designed to measure the Big Five personality traits (McCrae & Costa, 1997). The network shows many meaningful connections, such as "make friends easily" being linked to "make people feel at ease," "don't talk a lot" being linked to "find it difficult to approach others," and "carry the conversation to a higher level" being linked to "know how to captivate

---

Parts of this chapter have been adapted from: Epskamp, S., Borsboom, D., and Fried, E.I. (in press). Estimating Psychological Networks and their Accuracy: A Tutorial Paper. *Behavior Research Methods*.

people." A more detailed description on how to interpret such networks is included below and is discussed further (with respect to personality) in Chapter 10. Psychological networks are strikingly different to network models typically used in complexity research. These are networks between variables (nodes can take one of multiple states) rather than between concrete entities, such as people, computers, cities, and so forth. Furthermore, we do not know the structure of psychological networks. Due to the nascent status of this field, we do not yet know basic properties of psychological networks such as connectivity and clustering, which forces researchers to estimate network structures from data.

This dissertation addresses the problem of how to estimate network models from psychological data, and how such models should consequently be analyzed and interpreted. This is the first dissertation fully devoted to estimating and interpreting psychological networks. Some of the methods discussed (most notably the *qgraph* package; Chapter 9) have already grown to be commonly used psychological research. As a result, the number of researchers working on the estimation of psychological networks has grown substantively during the course of this PhD project. We can now speak of a new field of research: *network psychometrics*. The goal of this dissertation is to start of researchers in this field on the right foot.

Because most chapters of this dissertation utilize a certain class of network models, pairwise Markov random fields (PMRF; Lauritzen, 1996; Murphy, 2012), this introduction is followed by a general introduction to interpreting such models (adapted from Epskamp, Borsboom, & Fried, 2016, which is further used in Chapter 3). The introduction will conclude with a general outline of the dissertation.

1.2 Psychological Networks

A psychological network is a model in which *nodes* represent observed psychological variables, usually psychometric test items such as responses to questions about whether a person suffered from insomnia or fatigue in past weeks. These nodes are connected by *edges* which indicate some statistical relationship between them. These models are conceptually different from commonly used reflective latent variable models that explain the co-occurrence among symptoms (e.g., the fact that individuals often suffer from sadness, insomnia, fatigue, and concentration problems at the same time) by invoking an underlying unobserved latent trait (e.g., depression) as the common cause of all the symptoms. Psychological networks offer a different conceptual interpretation of the data and explain such co-occurrences via direct relationships between symptoms; for example, someone who sleeps poorly will be tired, and someone who is tired will not concentrate well (Fried et al., 2015; Schmittmann et al., 2013). Such relationships can then be more easily interpreted when drawn as a network structure where edges indicate pathways on which nodes can affect each other. The edges can differ in strength of connection, also termed *edge weight*, indicating if a relationship is strong (commonly visualized with thick edges) or weak (thin, less saturated edges) and positive (green edges) or negative (red edges). After a network structure is estimated, the visualization of the graph itself tells the researcher a detailed story of the multivariate dependencies in the data. Additionally, many inference methods from

Agreeableness
- A1: Am indifferent to the feelings of others.
- A2: Inquire about others' well–being.
- A3: Know how to comfort others.
- A4: Love children.
- A5: Make people feel at ease.

Conscientiousness
- C1: Am exacting in my work.
- C2: Continue until everything is perfect.
- C3: Do things according to a plan.
- C4: Do things in a half–way manner.
- C5: Waste my time.

Extraversion
- E1: Don't talk a lot.
- E2: Find it difficult to approach others.
- E3: Know how to captivate people.
- E4: Make friends easily.
- E5: Take charge.

Neuroticism
- N1: Get angry easily.
- N2: Get irritated easily.
- N3: Have frequent mood swings.
- N4: Often feel blue.
- N5: Panic easily.

Openness
- O1: Am full of ideas.
- O2: Avoid difficult reading material.
- O3: Carry the conversation to a higher level.
- O4: Spend time reflecting on things.
- O5: Will not probe deeply into a subject.

Figure 1.1: Example of a network model estimated the BFI personality dataset from the psych package in R. Nodes represent variables (in this case, personality inventory items) and links between the nodes (also termed *edges*) represent partial correlation coefficients. Green edges indicate positive partial correlations, red edges indicate negative partial correlations, and the width and saturation of an edge corresponds to the absolute value of the partial correlation. Estimation technique as outlined in Chapter 2 was used.

graph theory can be used to assess which nodes are the most important in the network, termed the *most central nodes*.

Directed and Undirected Networks

In general, there are two types of edges that can be present in a network: an edge can be directed, in which case one head of the edge has an arrowhead indicating a one-way effect, or an edge can be undirected, indicating some mutual relationship. A network that contains only directed edges is termed a *directed network*, whereas a network that contains only undirected edges is termed an *undirected network* (Newman, 2010). Many fields of science consider directed networks interesting because they can be used to encode causal structures (Pearl, 2000). For example, the edge insomnia → fatigue can be taken to indicate that insomnia causes fatigue. The work of Pearl describes that such causal structures can be tested using only observational cross-sectional data and can even be estimated to a certain extent (Kalisch, Mächler, Colombo, Maathuis, & Bühlmann, 2012; Scutari,

2010). However, when temporal information is lacking, there is only limited information present in cross-sectional observational data. Such estimation methods typically only work under two very strict assumptions (a) that all entities which play a causal role have been measured and (b) that the causal chain of cause and effect is not cyclic (i.e., a variable cannot cause itself via any path). Both assumptions are not very plausible in psychological systems. Furthermore, such directed networks suffer from the problem that many equivalent models can exist that feature the same relationships found in the data (MacCallum, Wegener, Uchino, & Fabrigar, 1993); this makes the interpretation of structures difficult. For example, the structure insomnia → fatigue → concentration is statistically equivalent to the structure insomnia ← fatigue → concentration as well as the structure insomnia ← fatigue ← concentration: All three only indicate that insomnia and concentration problems are conditionally independent after controlling for fatigue.

For the reasons outlined above, psychological networks estimated on cross-sectional data are typically undirected networks. The current state-of-the-art method for estimating undirected psychological network structures involves the estimation of PMRFs. A PMRF is a network model in which edges indicate the full conditional association between two nodes after conditioning on all other nodes in the network. This means when two nodes are connected, there is a relationship between these two nodes that cannot be explained by any other node in the network. Simplified, it can be understood as a partial correlation controlling for all other connections. The absence of an edge between two nodes indicates that these nodes are conditionally independent of each other given the other nodes in the network. Thus, a completely equivalent undirected structure (compared to the structures described above) would be insomnia — fatigue — concentration, indicating that insomnia and concentration problems are conditionally independent after controlling for fatigue.

Figure 1.2 shows a PMRF similar to the example described above. In this network, there is a positive relationship between insomnia and fatigue and a negative relationship between fatigue and concentration. The positive edge is thicker and more saturated than the negative edge, indicating that this interaction effect is stronger than that of the negative edge. This network shows that insomnia and concentration do not directly interact with each other in any way other than through their common connection with fatigue. Therefore, fatigue is the most important node in this network—a concept we will later quantify as *centrality*. These edges can be interpreted in several different ways.[1] First, as shown above, the model is in line with causal interpretations of associations among the symptoms. Second, this model implies that insomnia and fatigue predict each other after controlling for concentration; even when we know someone is concentrating poorly, that person is more likely to suffer from insomnia when we observe that person suffering from fatigue. Similarly, fatigue and concentration predict each other after controlling for insomnia. After controlling for fatigue, there is no longer any predictive quality between insomnia and concentration, even though these variables are correlated; fatigue now mediates the prediction between these

---

[1]A more detailed description of the interpretation of such models can be found in Chapter 6 and Chapter 8.
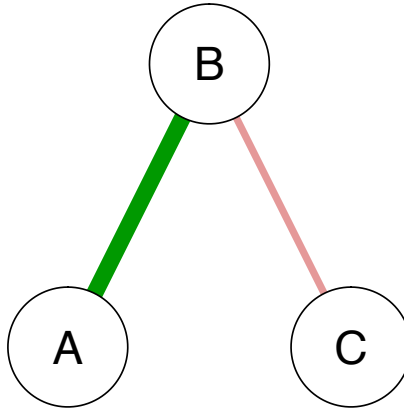
Figure 1.2: Example of a pairwise Markov random field. Node *A* positively interacts with node *B*, and node *B* negatively interacts with node *C*. Nodes *A* and *C* are *conditionally independent* given node *B*.

two symptoms. Finally, these edges can represent genuine symmetric causal interactions between symptoms (e.g., in statistical physics, a PRMF called the Ising model is used to model particles that cause neighboring particles to be aligned).

Network Inference

In the first step of network analysis, the obtained network is typically presented graphically to show the structure of the data. Afterwards, inference methods derived from graph theory can be applied to the network structure. The estimated PRMF is always a weighted network, which means that we not only look at the structure of the network (e.g., are two nodes connected or not) but also at the strength of connection between pairs of nodes. Because of this, many typical inference methods that concern the global structure of the network (e.g., small-worldness, density, and global clustering; Kolaczyk, 2009; Newman, 2010; Watts & Strogatz, 1998) are less useful in the context of psychological networks because they only take into account whether nodes are connected or not and not the strength of association among nodes. Because the global inference methods for weighted networks and PRMFs are still in development and no consensus has been reached, the network inference section focuses on local network properties: How are two nodes related, and what is the influence of a single node?

Relationship between two nodes. The relationship between two nodes can be assessed in two ways. First, we can directly assess the edge weight. This is

always a number that is nonzero because an edge weight of zero would indicate there is no edge. The sign of the edge weight (positive or negative) indicates the type of interaction, and the absolute value of the edge weight indicates the strength of the effect. For example, a positive edge weight of 0.5 is equal in strength to a negative edge weight of -0.5 and both are stronger than an edge weight of 0.2. Two strongly connected nodes influence each other more easily than two weakly connected nodes. This is similar to how two persons standing closer to each other can communicate more easily (via talking) than two people standing far away from each other (via shouting)—two strongly connected nodes are closer to each other. As such, the length of an edge is defined as the inverse of the edge strength. Finally, the distance between two nodes is equal to the sum of the lengths of all edges on the shortest path between two nodes (Newman, 2010).

Node centrality. The importance of individual nodes in the network can be assessed by investigating the node centrality. A visualization of a network, such as the one shown in Figure 1.2, is an abstract rendition of a high-dimensional space in two dimensions. Although visualizations of network models often aim to place highly connected nodes into the center of the graph, for instance using the Fruchterman-Reingold algorithm (Fruchterman & Reingold, 1991), the two-dimensional visualization cannot properly reflect the true space of the model. Thus, the metric distance between the placement of nodes in the two-dimensional space has no direct interpretation as it has in multidimensional scaling, for instance. Therefore, graph theory has developed several methods to more objectively quantify which node is most central in a network. Three such centrality measures have appropriate weighted generalizations that can be used with psychological networks (Opsahl, Agneessens, & Skvoretz, 2010). First, node strength, also called degree in unweighted networks (Newman, 2010), simply adds up the strength of all connected edges to a node; if the network is made up of partial correlation coefficients, the node strength equals the sum of absolute partial correlation coefficients between a node and all other nodes. Second, closeness takes the inverse of the sum of all the shortest paths between one node and all other nodes in the network. Thus, where node strength investigates how strongly a node is directly connected to other nodes in the network, closeness investigates how strongly a node is indirectly connected to other nodes in the network. Finally, betweenness looks at how many of the shortest paths between two nodes go through the node in question; the higher the betweenness, the more important a node is in connecting other nodes.

1.3 Outline

The above describes, in short, the basis of the methodology introduced in this dissertation. Although the methods and software discussed in this dissertation are applicable to many fields both in and outside of psychology[2], it is in three fields

---

[2]In recent literature, the methodology has been applied to diverse fields, such as attitude formation (Dalege et al., 2016), test validity (Ziegler, Booth, & Bensch, 2013), dog personality (Goold, Vas, Olsen, & Newberry, 2015), plant breeding (da Silva, Cecon, & Puiatti, 2015; Silva,

where the method has been most applied: clinical psychology, psychometrics, and personality research. As a result of this diversity, the audience of contributions in network psychometrics varies as well between empirical researchers without much programming knowledge, researchers with strong technical skills, and researchers that are familiar with programming in R. To this end, the dissertation is split in three parts: Part I is aimed at empirical researchers with an emphasis on clinical psychology, Part II is aimed at technical researchers with an emphasis on psychometrics, and Part III is aimed at R users with an emphasis on personality research.

Network models have gained extensive footing in clinical psychology for their ability to highlight the dynamics that may lead to someone developing or maintaining mental illness (Cramer et al., 2010; Borsboom & Cramer, 2013; Borsboom, in press). For this reason, Part I of this dissertation is aimed at empirical researchers, with an emphasis on clinical psychology. Chapter 2 will introduce the reader to the network estimation technique we now most often employ on cross-sectional data: model selection on a series of regularized networks. Chapter 3 continues on this topic and presents methods for assessing the accuracy of the network estimated as well as the stability of inferences made on the network structure after observing only subsets of people. Chapter 4 continuous discussing network estimation using regularization and provides a more critical note to argue that results from the methods described in this part are not without problems and certain conclusions cannot be drawn. Finally, Chapter 5 gives a conceptual introduction to time-series modeling on the data of a single patient in clinical practice.

Part II, on the other hand, is aimed at psychometricians and researchers with strong technical skills. Simply stated, these chapters contain equations. Chapter 6 will further outline the modeling of time-series data. This chapter will give the theoretical justification for the model described in Chapter 5 and extends this modeling framework to situations where time-series data is available from multiple people. Rather than contrasting network modeling to the latent variable modeling of classical psychometrics, Chapter 7 and Chapter 8 outline how these frameworks can be combined. Chapter 7 will introduce the network model for multivariate normal data as a formal psychometric model and will cast this model in the well-known structural equation modeling (SEM) framework. Combining network models into SEM allows for two new modeling frameworks, both of which show promise in psychometrics. Finally, Chapter 8 will outline the relationship between the best known network model for binary data, the Ising model, to well-known psychometric modeling frameworks such as logistic regression, loglinear modeling, and item-response theory.

Part III continues the combination of using networks to augment classical psychometrics in three tutorial papers aimed at empirical researchers familiar with programming in R. Two of these chapters are aimed at the third field in which network modeling has grown prominent: personality research (Cramer, Sluis, et al., 2012). These chapters are not technical and are all tutorials using R. Chapter 9 will outline how network-based visualizations can be used to gain insight

---

Rêgo, Pessoa, & Rêgo, 2016), and ecology (Wilkins, Shizuka, Joseph, Hubbard, & Safran, 2015; Gsell, Özkundakci, Hébert, & Adrian, 2016).

into high-dimensional correlational structures as well as large exploratory factor analysis results. Chapter 10 introduces the network perspective to personality researchers. Both Chapter 9 and Chapter 10 will show the use of these methods in personality inventory datasets. Finally, Chapter 11 will introduce an R package that can be used to draw path diagrams and to visualize correlational structures directly from SEM software output.

Part I

A Practical Guide to Network Psychometrics

Regularized Partial Correlation Networks

Abstract

Recent years have seen an emergence of network modeling applied to moods, attitudes, and problems in the realm of psychology. In this framework, psychological variables are understood to directly interact with each other rather than being caused by an unobserved latent entity. In this tutorial, we introduce the reader to estimating the most popularly used network model for psychological data: the partial correlation network. We describe how regularization techniques can be used to efficiently estimate a parsimonious and interpretable network structure on cross-sectional data. We show how to perform these analyses in R and demonstrate the method in an empirical example on post-traumatic stress disorder data. In addition, we discuss the effect of the hyperparameter that needs to be manually set by the researcher and provide a checklist with potential solutions for problems often arise when estimating regularized partial correlation networks. The chapter concludes with a simulation study that shows the performance of the discussed methodology using a plausible psychological network structure.

2.1 Introduction

Recent years have seen the emergence of the use of network modeling for exploratory studies of psychological behavior as an alternative to latent-variable modeling (Borsboom & Cramer, 2013; Schmittmann et al., 2013). In these so-called *psychological networks* (Epskamp, Borsboom, & Fried, 2016), nodes represent psychological variables such as mood states, symptoms or attitudes, and

---

This chapter has been adapted from: Epskamp, S., and Fried, E.I. (2016). A Tutorial on Regularized Partial Correlation Networks. *arXiv preprint*, arXiv:1607.01367, and: Epskamp, S. (2016). Regularized Gaussian Psychological Networks: Brief Report on the Performance of Extended BIC Model Selection. *arXiv preprint*, arXiv:1606.05771.

links between the nodes represent unknown statistical relationships that need to be estimated. As a result, this class of network models is strikingly different from e.g., social networks in which links are known (Wasserman & Faust, 1994), and poses novel problems of statistical inference. A great body of technical literature exists on the estimation of such network models (e.g., Meinshausen & Bühlmann, 2006; Friedman, Hastie, & Tibshirani, 2008; Hastie, Tibshirani, & Friedman, 2001; Hastie, Tibshirani, & Wainwright, 2015; Foygel & Drton, 2010). However, this line of literature often requires a more technical background than can be expected from psychological researchers and does not focus on the unique problems that come with analyzing psychological data, such as the handling of ordinal data, interpretability of networks based on different samples and attempting to find evidence for an underlying causal mechanism. While this tutorial is aimed at empirical researchers in psychology, it should be noted that the methodology can readily be applied to other fields of research as well.

The main type of model used to estimate psychological methods are so-called pairwise Markov random fields (PMRF; Lauritzen, 1996; Murphy, 2012). The present chapter will focus on the most common PMRF for continuous data: *partial correlation networks*. Partial correlation networks are usually estimated using *regularization*, an important statistical procedure that helps to recover the true network structure of the data. In this chapter, we present a tutorial on estimating such regularized partial correlation networks, using a methodology implemented in the *qgraph* package (Epskamp, Cramer, Waldorp, Schmittmann, & Borsboom, 2012) for the statistical programming language R (R Core Team, 2016). This methodology has already been used in a substantive number of publications in diverse fields, such as psychology, psychiatry, health sciences and more (e.g., Fried, Epskamp, Nesse, Tuerlinckx, & Borsboom, 2016; Isvoranu, van Borkulo, et al., 2016; Isvoranu, Borsboom, van Os, & Guloksuz, 2016; Knefel, Tran, & Lueger-Schuster, 2016; Levine & Leucht, 2016; Jaya, Hillmann, Reininger, Gollwitzer, & Lincoln, 2016; Deserno, Borsboom, Begeer, & Geurts, 2016; McNally, 2016; Kossakowski et al., 2016; Langley, Wijn, Epskamp, & Van Bork, 2015; van Borkulo et al., 2015). However, the methodology itself has not yet been introduced in psychological literature. In addition, because of the novelty of regularized partial correlation networks in psychological research, we are not aware of concise and clear introductions aimed at empirical researchers that explain regularization. The goal of this chapter is thus (1) to provide a short introduction to regularization partial correlation networks, (2) to outline the commands used in R to perform this procedure, and (3) to present a checklist for identifying the most common problems and questions arising from regularized networks. In addition, this chapter will present simulation results that show the described estimation method works well with plausible psychological networks on both continuous and ordinal data.

2.2 Partial Correlation Networks

The most commonly used framework for constructing a psychological network on data that can be assumed to be multivariate normal[1] is to estimate a network of *partial correlation coefficients* (McNally et al., 2015; Borsboom & Cramer, 2013). Such networks can also be termed *concentration graphs* (Cox & Wermuth, 1994) or *Gaussian graphical models* (Lauritzen, 1996). Each link in the network represents a partial correlation coefficient between two variables after conditioning on all other variables in the dataset. These coefficients range from −1 to 1 and encode the remaining association between two nodes after controlling for all other information possible, also known as conditional independence associations. Typically, the connections are visualized using red lines indicating negative partial correlations, green lines indicating positive partial correlations, and wider and more saturated connections indicate partial correlations that are far from zero (see Chapter 9). Whenever the partial correlation is exactly zero, no connection is drawn between two nodes, indicating that two variables are independent after controlling for all other variables in the network. This is of particular interest since such a missing connection indicates one of the two variables could not have caused the other (Pearl, 2000). As such, whenever there is a connection present, it highlights a potential causal pathway between two variables (see also Chapter 6).

Due to sampling variation, we do not obtain partial correlations that are exactly zero when estimating a partial correlation network. Instead, even when in reality two variables are conditionally independent, we still obtain partial correlations that are very small and are represented as very weak edges in the network. These connections are called *spurious* (Costantini, Epskamp, et al., 2015), as they represent relationships that are not true in reality. We wish to control for such spurious connections, especially considering the fact that we estimate a large number of parameters in partial correlation networks that can also lead to false positive associations. One way to do so is to test all partial correlations for statistical significance and remove all connections that fail to reach significance (Drton & Perlman, 2004). However, this poses a problem of multiple testing, and controlling for this problem (e.g., by using a Bonferroni correction) results in a loss of power (Costantini, Epskamp, et al., 2015).

2.3 LASSO Regularization

An increasingly popular method for controlling for spurious connections—as well as to obtain easier interpretable networks that may perform better in cross-validation prediction—is to use statistical *regularization* techniques originating in the field of machine learning. The goal here is to obtain a network structure in which as few connections as possible are required to parsimoniously explain the covariance among variables in the data. Especially prominent is to use of the 'least absolute shrinkage and selection operator' (LASSO; Tibshirani, 1996).

---

[1]The assumption of normality can be relaxed by applying a transformation when data are continuous but not normal (Liu, Lafferty, & Wasserman, 2009), or by basing the network estimation on polychoric correlations when the data are ordinal.

In essence, the LASSO shrinks partial correlation coefficients when estimating a network model, which means that small coefficients are estimated to be exactly zero. This results in fewer connections in the network, or in other words, a *sparse* network in which likely spurious connections are removed. The LASSO utilizes a tuning parameter λ (lambda) that needs to be set, controlling this level of sparsity. When the tuning parameter is low, only few connections are removed, likely resulting in too many spurious connections. When the tuning parameter is high, many connections are removed, likely resulting in too many true connections to be removed in addition to all spurious connections. More broadly, when λ equals zero every connection remains in the network and when λ is substantively high no connection remains in the network. As such, the tuning parameter needs to be carefully selected to result in a network structure that minimizes the number of spurious connections while maximizing the number of true connections (Foygel Barber & Drton, 2015; Foygel & Drton, 2010).

Typically, a range of networks is estimated under different values of λ (Zhao & Yu, 2006). The value for λ under which no edges are retained (the empty network), $\lambda_{\max}$, is set to the largest absolute correlation (Zhao et al., 2015). A minimum value can be chosen by multiplying some ratio R with this maximum value[2]:

$$\lambda_{\min} = R\lambda_{\max}.$$

A logarithmically spaced range of tuning parameters (typically 100 different values), ranging from $\lambda_{\min}$ to $\lambda_{\max}$, can be used to estimate different networks. To summarize, the LASSO can be used to estimate a *range* of networks rather than a single network, ranging from a fully connected network to a fully disconnected network. Next, one needs to select the best network out of this range of networks. This selection can be done by optimizing the fit of the network to the data (i.e. by minimizing some information criterion). Minimizing the Extended Bayesian Information Criterion (EBIC; Chen & Chen, 2008) has been shown to work particularly well in retrieving the true network structure (Foygel Barber & Drton, 2015; Foygel & Drton, 2010; van Borkulo et al., 2014), especially when the generating network is sparse (i.e., does not contain many edges). LASSO regularization with EBIC model selection has been shown to have high specificity all-around (i.e., does not estimate edges that are not in the true network) but a varying sensitivity (i.e., estimates edges that are in the true network) based on the true network structure and sample size. For example, sensitivity typically is less when the true network is dense (contains many connections) or features some nodes with many edges (hubs).

Many variants of the LASSO with different methods for selecting the LASSO tuning parameter have been implemented in open-source software (e.g., Krämer, Schäfer, & Boulesteix, 2009; Zhao et al., 2015). We suggest to use the variant termed the 'graphical LASSO' (glasso; Friedman et al., 2008), which is a fast variant of the LASSO specifically aimed at estimating partial correlation networks. The glasso algorithm has been implemented in the *glasso* package (Friedman, Hastie, & Tibshirani, 2014) for the statistical programming language R (R Core Team, 2016). An automatic function that uses this package in combination with

---

[2]The *qgraph* package uses $R = 0.01$ by default.
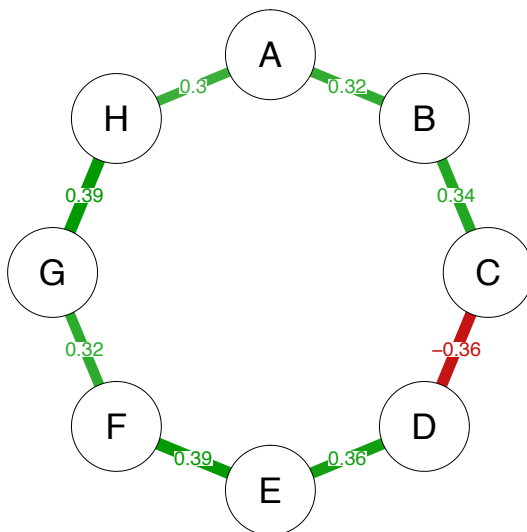
Figure 2.1: True network structure used in simulation example. The network represents a *partial correlation network*: nodes represent observed variables and links represent partial correlations between two variables after conditioning on all other variables. The simulated structure is a *chain graph* in which all absolute partial correlation coefficients were drawn randomly between 0.3 and 0.4.

EBIC model selection as described by Foygel and Drton (2010) has been implemented in the R package *qgraph* (Epskamp et al., 2012). We suggest using this routine because—in addition to simple input commands—it only requires an estimate of the covariance matrix and not the raw data, allowing one to use, e.g., polychoric correlation matrices when the data are ordinal.

The EBIC uses a hyperparameter γ (gamma) that controls how much the EBIC prefers simpler models (fewer connections). This hyperparameter γ should not be confused with the LASSO tuning parameter λ, and needs to be set manually. It typically is set between 0 and 0.5 (Foygel & Drton, 2010, suggest to use 0.5), with higher values indicating that simpler models (more parsimonious models with fewer connections) are preferred. Setting the hyperparameter to 0 errs on the side of discovery: more connections are estimated, including possible spurious ones (the network has a higher specificity). Setting the hyperparameter to 0.5, on the other hand, errs on the side of caution or parsimony: fewer connections are obtained including hardly any spurious connections but also less true connections (the network has a higher sensitivity). It is important to mention that even when setting the hyperparameter to 0, the network will still be sparser compared to a partial correlation network that does not employ any form of regularization; setting γ to 0 indicates that the EBIC reduces to the standard BIC, which is still a criterion that prefers simple models.

To exemplify the above-described method of selecting a best fitting regular-

ized partial correlation network, we simulated a dataset of 100 people and 8 nodes based on the *chain graph* shown in Figure 2.1. Such graphs are particularly suitable for our example because the true network (the one we want to recover with our statistical analysis) only features connections among neighboring nodes visualized in a circle. This makes spurious connections—any edge that connects non-neighboring nodes—easy to identify visually. We used the *qgraph* package to estimate 100 different network structures, based on different values for λ, and compute the EBIC under different values of γ. Figure 2.2 depicts a representative sample of 10 of these networks. As can be seen, networks 1 through 7 feature spurious connections and err on the side of discovery, while networks 9 and 10 recover too few connections and err on the side of caution. For each network, we computed the EBIC based on γ of 0, 0.25 and 0.5 (the parameter the researchers needs to set manually). The boldface values show the best fitting models, indicating which models would be selected using a certain value of γ. When $\gamma = 0$ was used, network 7 was selected that featured three weak spurious connections. When γ was set to 0.25 or 0.5 (the default in *qgraph*) respectively, network 8 was selected, which has the same structure as the true network shown in Figure 2.1. These results show that in our case, varying γ changed the results only slightly. Importantly, this simulation does not imply that $\gamma = 0.5$ always leads to the true model; simulation work has shown that 0.5 is fairly conservative and may result in omitting true edges from the network, while it is very unlikely that spurious ones are obtained (Foygel & Drton, 2010). In sum, the choice of the hyperparameter is somewhat arbitrary and up to the researcher, and depending on the relative importance assigned to caution or discovery (Dziak, Coffman, Lanza, & Li, 2012). Which of these γ values work best is a complex function of the (usually unknown) true network structure.

A note on sparsity. It is important to note that both thresholding networks based on significance of edges or using LASSO regularization will lead to edges being removed from the network (termed a sparse network), but do not present evidence that these edges are, in fact, zero (see Chapter 4). This is because these methods seek to maximize *specificity*; that is, they all aim to include as few *false positives* (edges that are not in the true model) as possible. All these methods will return empty network structures when there is not enough data. It is important to note that observing a structure with missing edges, or even an empty network, is in no way evidence that there are, in fact, missing edges. This is because these methods do not try to keep the number of *false negatives* low, that is, the number of edges that are not present in the estimated network but are present in the true network. This is related to a well-known problem of null hypothesis testing (to which, roughly, all these methods correspond): *Not* rejecting the null-hypothesis is not evidence that the null hypothesis is true (Wagenmakers, 2007). That is, we might not include an edge because the data are too noisy or because the null hypothesis is true; classical tests and LASSO regularization cannot differentiate between these two reasons. Quantifying evidence for edge weights being zero is still a topic of future research (see Chapter 12).
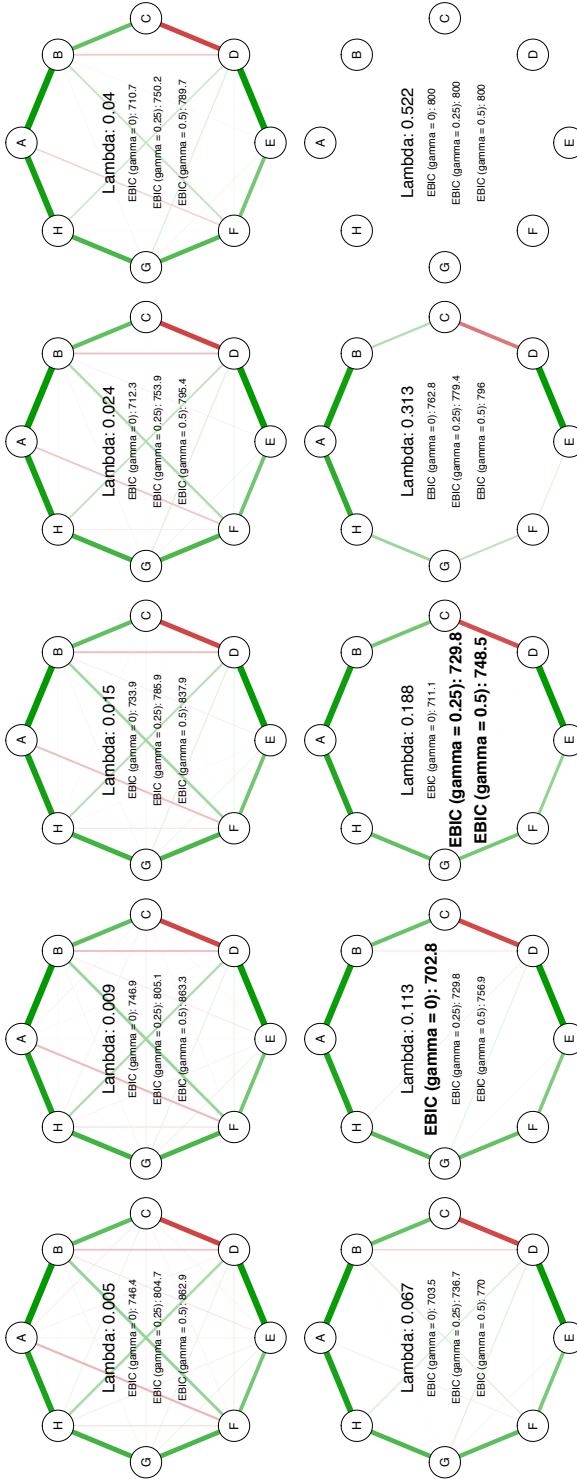
Figure 2.2: Ten different partial correlation networks estimated using LASSO regularization. Setting the LASSO tuningparameter λ that controls sparsity leads to networks ranging from densely connected to fully unconnected. Data were simulated under the network represented in Figure 2.1. The fit of every network was assessed using the EBIC, using hyperparameter γ set to 0, 0.25 or 0.5. The bold-faced EBIC value is the best, indicating the network which would be selected and returned using that γ value.

2.4 Example

In this paragraph, we use an example dataset to estimate a network on data of 221 people with a sub-threshold post-traumatic stress disorder (PTSD) diagnosis; the network features 20 PTSD symptoms. A detailed description of the dataset can be found elsewhere (Armour et al., 2016), and the full R codes for this analysis can be found in the supplementary materials.

The following R codes perform regularized estimation of a partial correlation network using EBIC selection (Foygel & Drton, 2010). These codes make use of the *qgraph* package (Epskamp et al., 2012), which in turns utilizes the *glasso* package for the glasso algorithm (Friedman et al., 2014). These codes assume data is present in R under the object name `Data`.

```
library("qgraph")
corMat <- cor_auto(Data)
graph <- qgraph(corMat,
    graph = "glasso",
    sampleSize = nrow(Data),
    layout = "spring",
    tuning = 0.5)
```

In these codes, `library("qgraph")` loads the package into R and the `cor_auto` function detects ordinal variables (variables with up to 7 unique integer values) and uses the *lavaan* package (Rosseel, 2012) to estimate polychoric, polyserial and Pearson correlations. The `qgraph` function estimates and plots the network structure. The argument `graph` specified that we want to use the glasso algorithm with EBIC model selection, the argument `sampleSize` specifies the sample size of the data, the argument `layout` specifies the node placement and the argument tuning specified the EBIC hyperparameter. The hyperparameter is here set to 0.5, which is also the current default value used in *qgraph*. For more control on the estimation procedure, one can use the `EBICglasso` function, which is automatically called when using `qgraph(..., graph = "glasso")`. Finally, the estimated weights matrix can be obtained either directly using `EBICglasso` or by using the `getWmat` function on the output of `qgraph`:

```
getWmat(graph)
```

Figure 2.3 shows the resulting network estimated under three different values of the hyperparameter 0, 0.25, and 0.5. Table 2.1 shows the description of the nodes. If we investigate the number of edges, we would expect that the network with the largest hyperparameter of 0.5 has the fewest connections. This is indeed the case: the network features 105 edges with $\gamma = 0$, 95 edges with $\gamma = 0.25$, and 87 edges with $\gamma = 0.5$.

We can further investigate properties of the network structures by investigating how important nodes are in the network using measures called centrality indices. A plot of these indices can be obtained as followed:

```
centralityPlot(graph)
```
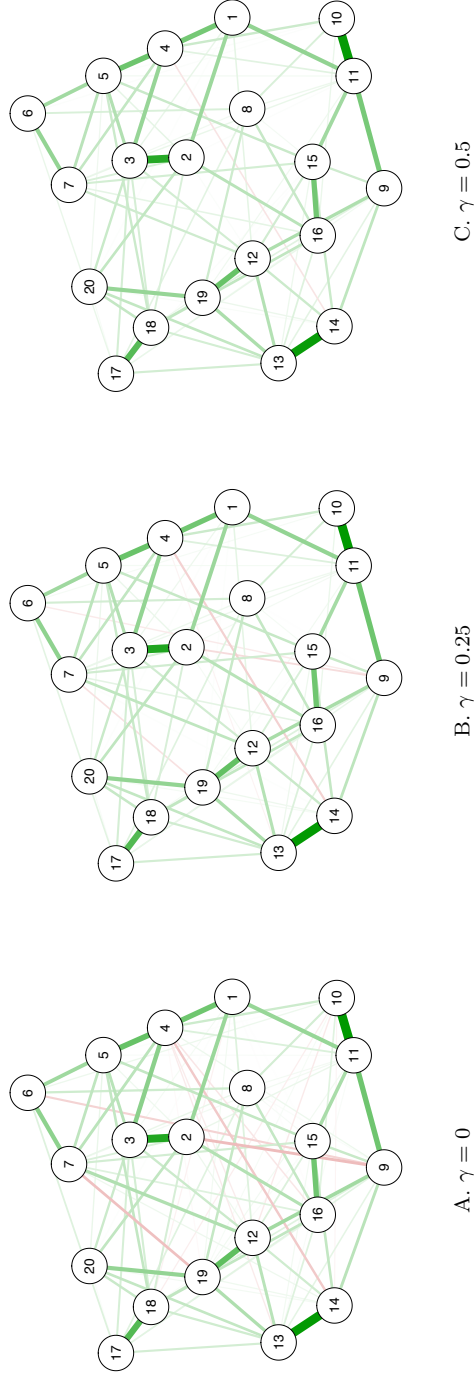
Figure 2.3: Partial correlation networks estimated on responses of 221 subjects on 20 PTSD symptoms, with increasing levels of the LASSO hyperparameter γ (from left to right: panel A = 0, panel B = 0.25, panel C = 0.5).

A. $\gamma = 0$

B. $\gamma = 0.25$

C. $\gamma = 0.5$

Table 2.1: Description of nodes shown in Figure 2.3

Node	Description
1	Intrusive Thoughts
2	Nightmares
3	Flashbacks
4	Emotional cue reactivity
5	Psychological cue reactivity
6	Avoidance of thoughts
7	Avoidance of reminders
8	Trauma-related amnesia
9	Negative beliefs
10	Blame of self or others
11	Negative trauma-related emotions
12	Loss of interest
13	Detachment
14	Restricted affect
15	Irritability/anger
16	Self-destructive/reckless behavior
17	Hypervigilance
18	Exaggerated startle response
19	Difficulty concentrating
20	Sleep disturbance

An overview of these measures and their interpretation can be found in Chapter 1 and Chapter 10. All measures indicate how important nodes are in a network, with higher values indicating that nodes are more important. Figure 2.4 was made using `centralityPlot` and shows the resulting centrality of all three networks shown in Figure 2.3. For a substantive interpretation of the network model obtained from this dataset we refer the reader to Armour et al. (2016).

2.5 Common Problems and Questions

The estimation of regularized networks is not always without problems and can sometimes lead to network structures that are hard to interpret. Here, we list several common problems and questions encountered when estimating these models.

1. The estimated network has no or very few edges. This can occur in the unlikely case when variables of interest do not exhibit partial correlations. More likely, it occurs when the sample size is too low for the number of nodes in the network. The EBIC penalizes edge weights based on sample size to avoid false positive associations, which means that with increasing sample size, the partial correlation network will be more and more similar to the regularized partial correlation network. The smaller the sample, however, the stronger the impact of the regularization on the network in terms of parsimony. Figure 2.5 (panel A) shows a network estimated on the same data
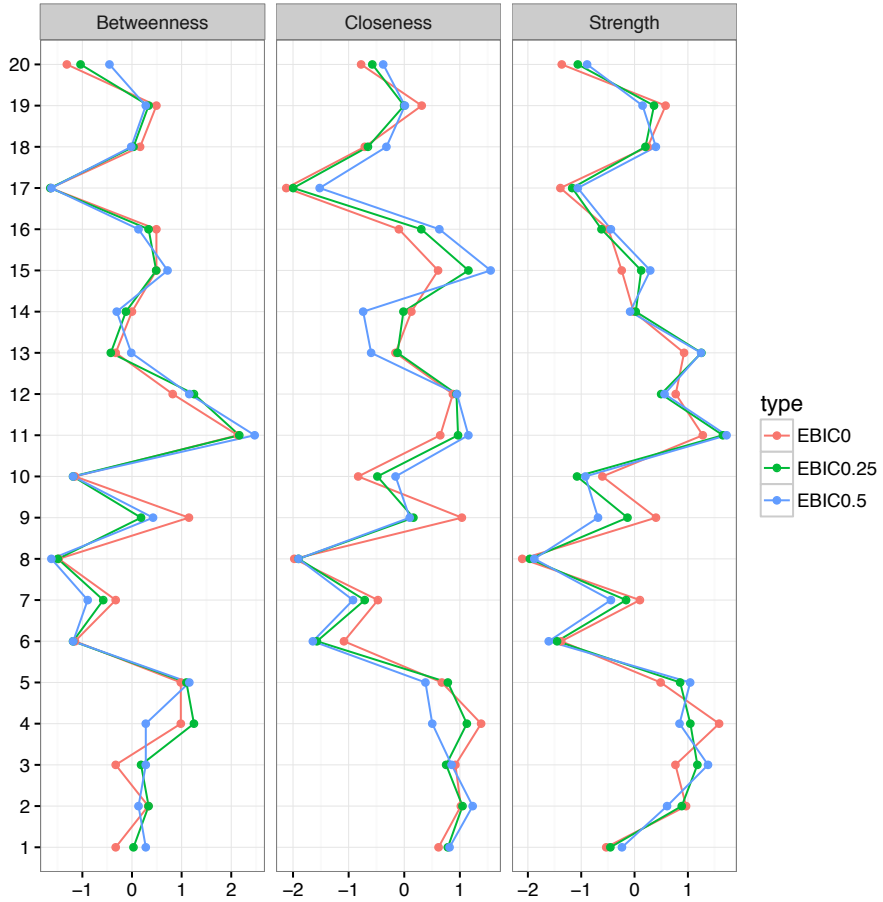
Figure 2.4: Closeness, betweenness, and degree centrality of the three networks described in Figure 2.3 with increasing levels of the LASSO hyperparameter γ. Values are standardized to z-scores.

as Figure 2.3, but this time with only 50 instead of the 221 participants. A
way to remediate this problem is by setting the hyperparameter lower (e.g.,
0; see Figure 2.5 panel B). Note that this likely leads to spurious connections.
An alternative solution is to make a selection of the variables of interest and
estimate a network based only on a subset of variables, as less nodes in the
network leads to less edges to be estimated, resulting in more observations
per parameter to be estimated.

2. The network is densely connected (i.e., many edges) including many unex-
pected negative connections and, in particular, including many implausibly
high partial correlations (e.g., higher than 0.8). As the LASSO aims to
remove connections and return a relatively sparse network, we would not
expect densely connected networks. In addition, we would not expect many
partial correlations to be so high, as (partial) correlations above 0.8 indi-
cate near-perfect collinearity between variables. These structures can occur
when the correlation matrix used as input is not *positive definite*, which in
turn can be a result of a too small sample size, or of the estimation of poly-
choric correlations. In the case of a non-positive definite correlation matrix,
`cor_auto` will warn the user and attempt to correct for this by searching
for a nearest positive definite matrix. This matrix, however, can still lead
to wildly unstable results. When the network looks very strongly connected
with few (if any) missing connections and partial correlations near 1 and -1,
the network structure is likely resulting from such a problem and should not
be interpreted. We suggest that researchers always compare networks based
on polychoric correlations with networks based on Spearman correlations
(they should look somewhat similar) to rule out if estimating the polychoric
correlations are the source of this problem.

3. While in general the graph looks as expected (i.e., relatively sparse), some
connections are extremely high and/or unexpectedly extremely negative.
This problem is related to the previous problem. The estimation of poly-
choric correlations relies on the pairwise cross-tables of variables in the
dataset. When the sample size is relatively low, some cells in the cross-
tables could be zero (e.g., nobody was observed that scored a 2 on one item
and a 1 on another item). This can lead to unstable estimated polychoric
correlations, and in turn to unstable partial correlations. Again, the network
based on polychoric correlations should be compared to a network based on
Spearman correlations. Obtaining very different networks indicates that the
estimation of the polychoric correlations may not be trustworthy.

4. Negative connections are found between variables where one would expect
positive connections. For example, two symptoms of the same disorder could,
unexpectedly, feature a negative partial correlation rather than a positive
one. This can occur artificially when one conditions on a *common effect*
(Pearl, 2000). Suppose one measures students' grades of a recent test, their
motivation, and the easiness of that test (Koller & Friedman, 2009). We
expect the grade to be positively influenced by the easiness of the test and
the motivation of the student, and we do not expect any correlation between
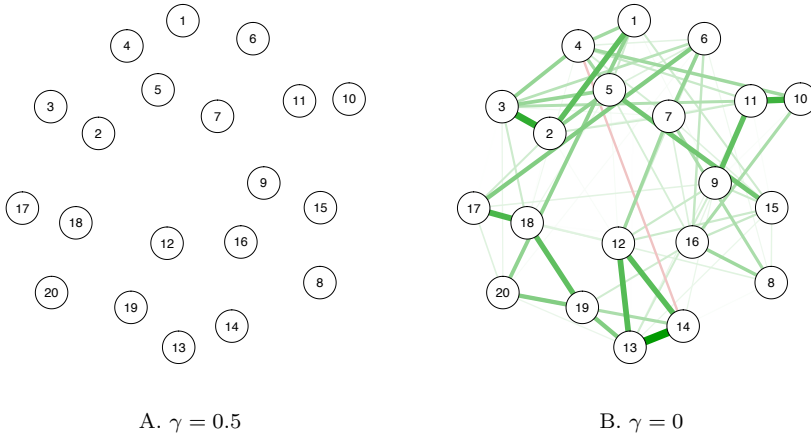
Figure 2.5: Network of 20 PTSD symptoms. Instead of the full data like in Figure 2.3 (221 subjects), only 50 subjects were used. Panel A: LASSO hyperparameter γ set to the default of 0.5; panel B: γ set to 0 for discovery.

motivation and easiness: knowing a student is motivated does not help us predict the easiness of a test. However, if we only look at students who obtained an A for the test (i.e., *conditioning* on grades), we now *can* predict that if the student is not at all motivated, the test must have been very easy. By conditioning on the common effect (grade) we artificially created a *negative* partial correlation between test easiness and student motivation. Because partial correlation networks indicate such conditional relationships, these negative relationships can occur when common effect relationships are present, and unexpected negative relationships might indicate common effect structures. Another way these relationships can occur is if the network is based on a subsample of the population, and that subsample is a common effect of the nodes in the network. For example, when one splits the sample based on the *sum score* of variables used also in the network, negative relationships could be induced. We recommend results based on such subsamples to be interpreted with care.

In addition to the above-mentioned problems, some questions are often encountered in network analysis:

1. How large does my sample have to be for a given number of nodes? Or in other words, how stable are the estimated network structures and centrality indices to sampling size? This topic goes beyond the scope of this chapter, and is further discussed in Chapter 3. In summary, networks are complicated models using many parameters, which can be unstable given relatively low sample sizes. The LASSO remedies this problem somewhat, and stable networks can be obtained with much smaller samples compared to unregularized networks. Nonetheless, network models estimate a large number of

parameters, implying that even when the LASSO is used, the models need considerable power to obtain stable parameter estimates. It is therefore advisable to always check for the accuracy and stability of edge weights and centrality measures when these are reported and substantively interpreted (c.f., Chapter 3).

2. Can we compare two different groups of people (e.g., clinical patients and healthy controls) regarding the connectivity or density of their networks (i.e. the number of connections)? The answer depends on the differences in sample size. As mentioned before, the EBIC is a function of the sample size: the lower the sample size, the more parsimonious the network structure. This means that comparing the connectivity of two networks is meaningful if they were estimated on roughly the same sample size, but that differences should not be compared if this assumption is not met (e.g., see Rhemtulla et al., 2016). A statistical test for comparing networks based on different sample sizes is currently being developed (Van Borkulo et al., 2016)[3].

3. Does the network structure provide evidence that the data are indeed causally interacting and derive from a true network model, and not from a common cause model where the covariance of symptoms is explained by one or more underlying latent variables (Schmittmann et al., 2013)? The short answer is no. While psychological networks have been introduced as an alternative modeling framework to latent variable modeling, and are capable of strongly changing the point of focus from the common shared variance to unique variance between variables (Costantini, Epskamp, et al., 2015), they do not necessarily disprove the latent variable model. There is a direct equivalence between network models and latent variable models (see Chapter 7 and Chapter 8), and if we generate data based on a true latent variable model, the corresponding network model will be fully connected. However, this does not mean that when the resulting network is not fully connected, the latent variable model must be false. LASSO estimation will *always* return a sparse network with at least some missing connections. As such, observing that there are missing connections does not indicate that the true model was a model without missing connections. Because of the equivalence stated above, observing a model with missing connections cannot be taken for evidence that a latent variable model was not true. A more detailed discussion on this topic can be found in Chapter 4 and a methodology on statistically comparing fit of a network model and latent variable model is described in Chapter 7. In addition, statistical tests to distinguish sparse networks from latent variable models are currently being developed (Van Bork, 2015).

2.6 Simulation Study

While partial correlation network estimation using EBIC model selection has already been shown to work well in retrieving the GGM structure (Foygel & Drton,

---

[3]github.com/cvborkulo/NetworkComparisonTest.
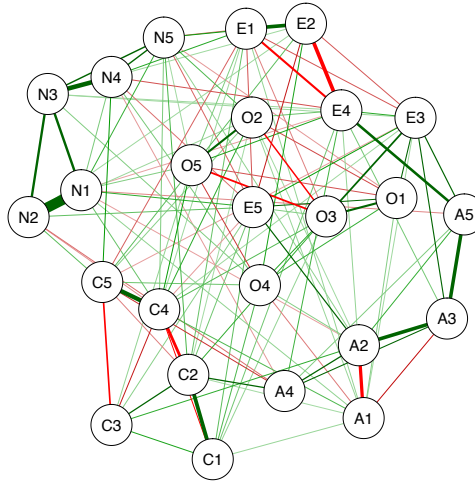
Figure 2.6: True Gaussian graphical model used in simulation study. The network was obtained by computing the (unregularized) sample partial correlation network on the BFI personality dataset from the psych package in R, followed by removing absolute edges below 0.05.

2010), it has not been validated in plausible scenarios for psychological networks. In addition, no simulation study has assessed the performance of using a polychoric correlation matrix in this methodology. To this end, this report presents a simulation study that assesses the performance in a plausible psychological network structure. In addition, the simulation study varied R and γ in order to provide recommendations of these parameters in estimating psychological networks. The simulation study makes use of the *qgraph* package, using the R codes described above.

Methods

To obtain a representative psychological network structure, the `bfi` dataset from the *psych* package (Revelle, 2010) was used on the Big 5 personality traits (Benet-Martinez & John, 1998; Digman, 1989; Goldberg, 1990a, 1993; McCrae & Costa, 1997). The `bfi` dataset consists of 2,800 observations of 25 personality inventory items. The network structure was obtained by computing the sample partial correlation coefficients (negative standardized inverse of the sample variance–covariance matrix; Lauritzen, 1996). Next, to create a sparse network all absolute edge weights below 0.05 were set to zero, thus removing edges from the network. Figure 2.6 shows the resulting network structure. In this network, 125 out of 300 possible edges were nonzero (41.6%). While this network is not the most appropriate network based on this dataset, it functions well as a proxy for psychological

network structures as it is both sparse (has missing edges) and has parameter values that are not shrunken by the LASSO.

In the simulation study, data was generated based on the network of Figure 2.6. Following, the network was estimated using the the *qgraph* codes described above. Sample size was varied between 50, 100, 250, 500, 1,000, and 2,500, γ was varied between 0, 0.25, 0.5, 0.75, and 1, and R was varied between 0.001, 0.01 and 0.1. The data was either simulated to be multivariate normal, in which case Pearson correlations were used in estimation, or ordinal, in which case polychoric correlations were used in the estimation. Ordinal data was created by sampling four thresholds for every variable from the standard normal distribution, and next using these thresholds to cut each variable in five levels. To compute polychoric correlations, the `cor_auto` function was used, which uses the `lavCor` function of the *lavaan* package (Rosseel, 2012). The number of different λ values used in generating networks was set to 100 (the default in *qgraph*).

For each simulation, in addition to the correlation between estimated and true edge weights, the sensetivity and specificity were computed (van Borkulo et al., 2014). The *sensitivity*, also termed the true-positive rate, indicates the proportion of edges in the true network that were estimated to be nonzero:

$$\text{sensitivity} = \frac{\#\text{ true positives}}{\#\text{ true positives} + \#\text{ of false negatives}}.$$

Specificity, also termed the true negative rate, indicates the proportion of true missing edges that were also estimated to be missing:

$$\text{specificity} = \frac{\#\text{ true negatives}}{\#\text{ true negatives} + \#\text{ false positives}}.$$

When specificity is high, there are not many false positives (edges detected to be nonzero that are zero in the true network) in the estimated network.

Results

Each of the conditions was replicated 1,000 times, leading to 180,000 simulated datasets. Figure 2.7 shows the sensitivity of the analyses. This figure shows that sensitivity increases with sample size and is high for large sample sizes. When $\gamma > 0$, small sample sizes are likely to result in empty networks (no edges), indicating a sensitivity of 0. When ordinal data is used, small sample sizes (50 and 100) resulted in far too densely connected networks that are hard to interpret. Setting γ to be higher remediated this by estimating empty networks. At higher sample sizes, γ does not play a role and sensitivity is comparable in all conditions. Using $R = 0.1$ remediates the poor performance of polychoric correlations in lower sample sizes, but also creates an upper bound to sensitivity at higher sample sizes.

Figure 2.8 shows the specificity of the analyses, which was all-around high except for the lower sample sizes in ordinal data using $R = 0.01$ or $R = 0.001$. Some outliers indicate that fully connected networks were estimated in ordinal data even when setting $\gamma = 0.25$ in small sample sizes. In all other conditions specificity was comparably high, with higher γ values only performing slightly
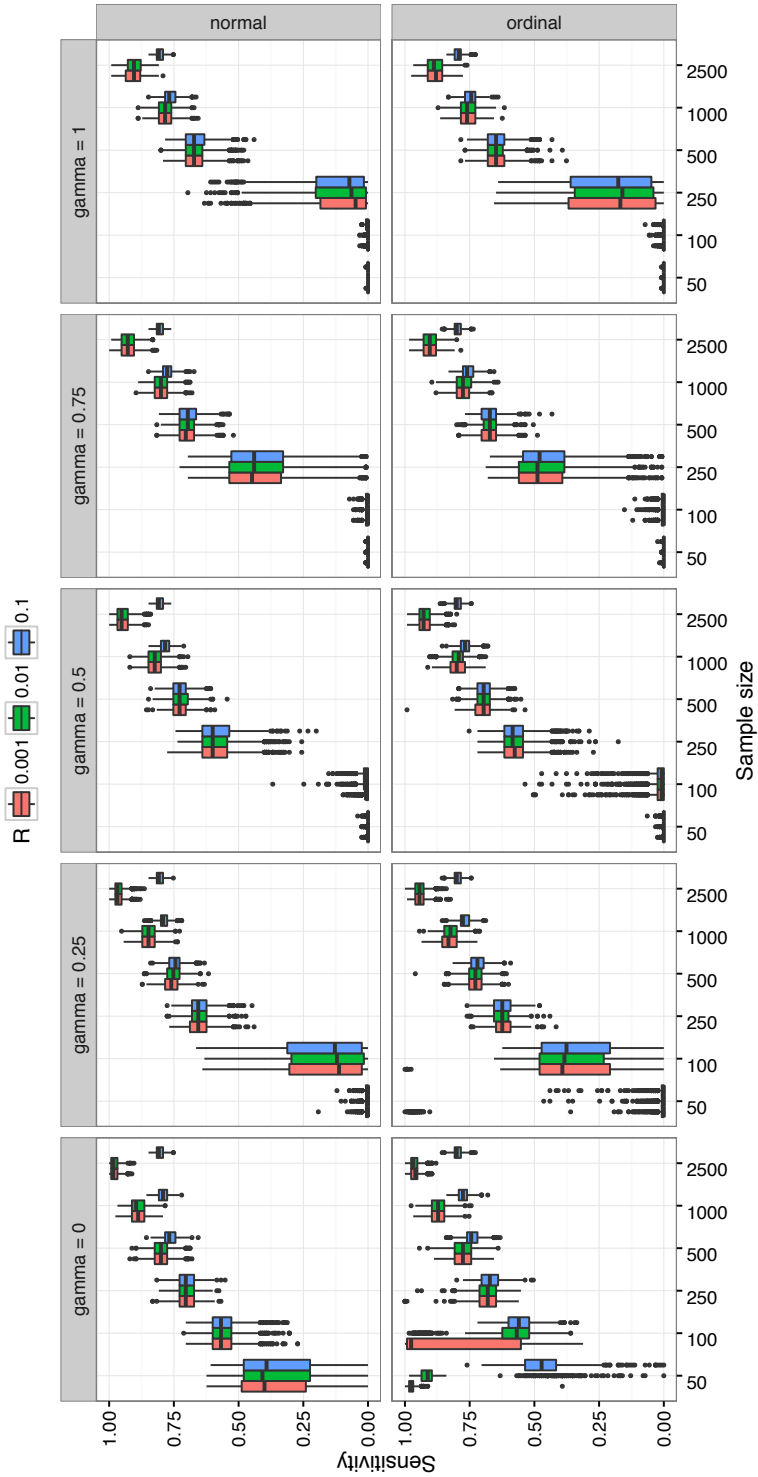
Figure 2.7: Sensitivity of the simulated datasets. Data is represented in standard boxplots. Horizontal panels indicate different EBIC hyperparameter values, vertical panels indicate if data was normal (Pearson correlations) or ordinal (polychoric correlations) and the color of the boxplots indicate the different ratio values used in setting the LASSO tuning parameter range. When sensitivity is high, true edges are likely to be detected.
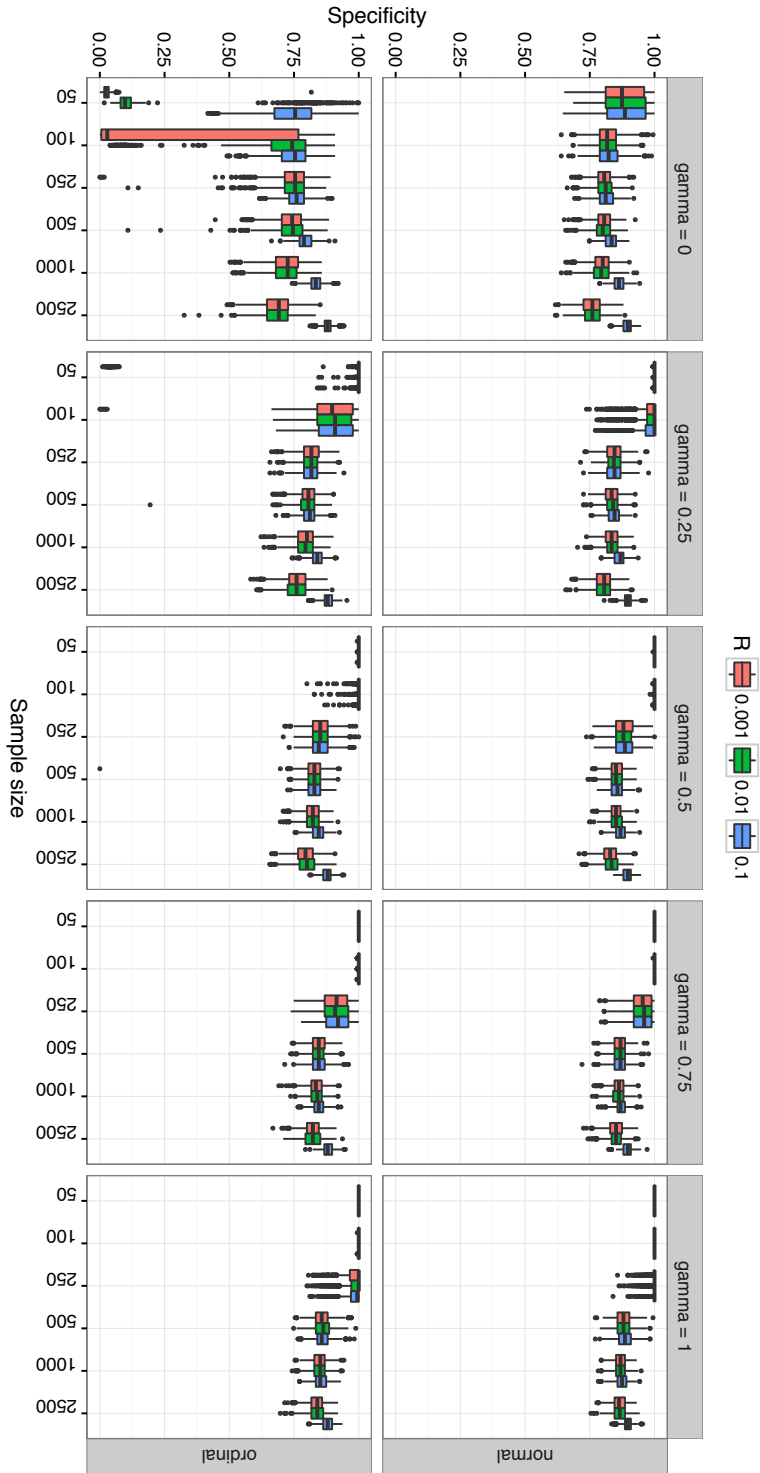
Figure 2.8: The specificity of the simulated datasets. When specificity is high, there are not many edges in the estimated network that are not present in the true network. See caption of Figure 2.7 for more details.
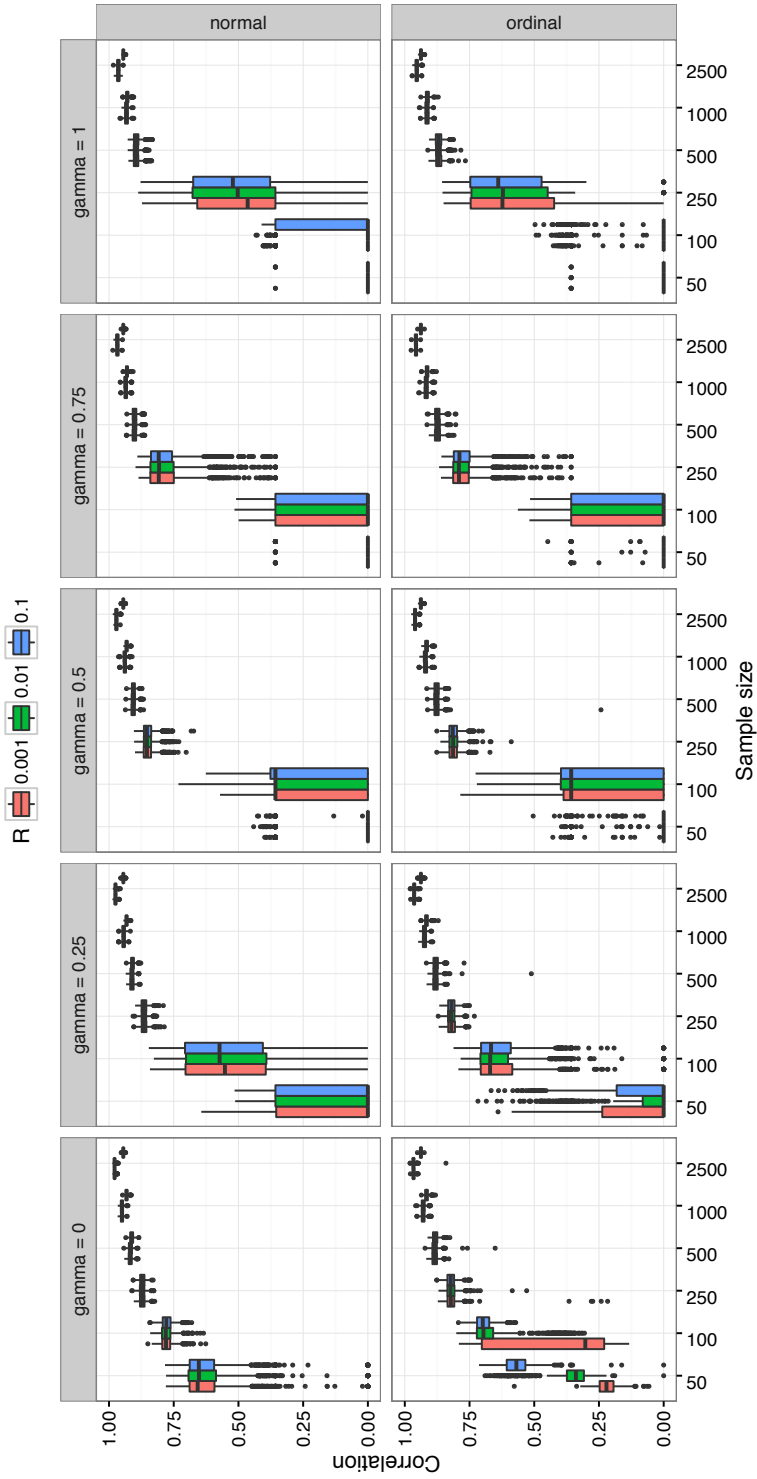
Figure 2.9: Correlation between true edge weights and estimated edge weights. See caption of Figure 2.7 for more details.

better. Figure 2.9 shows the correlation between true and estimated edge weights. This figure shows a comparable good performance from sample sizes of 250 and higher in all conditions, with γ values up to 0.5 outperforming the higher γ values. It should be noted that the correlation was set to zero if the estimated network had no edges (all edge weights were then zero).

2.7 Conclusion

This chapter presented tutorial on how to estimate psychological networks using a popular estimation technique: LASSO regularization with the EBIC model selection. The resulting network is a network of partial correlation coefficients controlled for spurious connections. One possibility to do so is provided by the *qgraph* R package that allows the estimation of network structure based on the correlation matrix of the data. The method also allows constructing partial correlation networks of ordered-categorical data by estimating the appropriate (in this case, polychoric) correlation matrix. The performance was assessed on 180,000 simulated datasets using a plausible psychological network structure. Results indicate that partial correlation networks could be well retrieved using either Pearson correlations or polychoric correlations. The default setup of *qgraph* uses $\gamma = 0.5$ and $R = 0.01$, which are shown to work well in all conditions. Setting $\gamma = 0.25$ improved the detection rate, but sometimes led to poorly estimated networks based on polychoric correlations. γ can be set to 0 to err more on the side of discovery (Dziak et al., 2012), but should be done with care in low sample polychoric correlation matrices. All conditions showed increasing sensitivity with sample size and a high specificity all-around. This is comparable to other network estimation techniques (van Borkulo et al., 2014), and shows that even though a network does not contain all true edges, the edges that are returned can usually be expected to be genuine. The high correlation furthermore indicated that the strongest true edges are usually estimated to be strong as well.

Many other estimation techniques exist. Regularized estimation of partial correlation networks can also be performed using the *huge* (Zhao et al., 2015) and *parcor* (Krämer et al., 2009) packages. When all variables are binary, one can estimate the Ising Model using, for instance, the *IsingFit* R package (van Borkulo & Epskamp, 2014). The resulting network has a similar interpretation as partial correlation networks, and is also estimated using LASSO with EBIC model selection (van Borkulo et al., 2014). When the data consist of both categorical and continuous variables, a state-of-the-art methodology is implemented in the *mgm* package (Haslbeck & Waldorp, 2016a) also making use of LASSO estimation with EBIC model selection. The *bootnet* package can subsequently be used to assess the accuracy of the estimated network structure obtained via *qgraph* or any of the other packages mentioned above (see also Chapter 3).

Important to note is that the methods described in this chapter are only appropriate to use when the cases in the data (the rows of the spreadsheet) can reasonably be assumed to be independent of one-another. Such is the case in cross-sectional analysis—where cases represent people that are measured only once—but not in longitudinal data where one person is measured on several occasions. In this

case, temporal information needs to be taken into account when estimating network structures. One way to do so is by using the *graphical vector-autoregression* model (graphical VAR; Wild et al., 2010). LASSO regularization making use of glasso in an iterative algorithm has been developed to estimate the network structures (Abegaz & Wit, 2013; Rothman, Levina, & Zhu, 2010). EBIC model selection using these routines has been implemented in the R packages *sparseTSCGM* (Abegaz & Wit, 2015; aimed at estimating genetic networks) and *graphicalVAR* (Epskamp, 2015; aimed at estimating $n = 1$ psychological networks).

In conclusion, while psychological network analysis is a novel field that is rapidly changing and developing, we have not seen an accessible description of the most commonly used estimation procedure in the literature: LASSO regularization using EBIC model selection to estimate a sparse partial correlation network. This chapter aimed to provide a short overview of this common and promising method.

Chapter 3

Accuracy of Psychological Networks

Abstract

The usage of *psychological networks* that conceptualize psychological be-
havior as a complex interplay of psychological and other components has
gained increasing popularity in various fields of psychology. While prior
publications have tackled the topics of estimating and interpreting such net-
works, little work has been conducted to check how *accurate* (i.e., prone to
sampling variation) networks are estimated, and how *stable* (i.e., interpre-
tation remains similar with less observations) inferences from the network
structure (such as centrality indices) are. In this chapter, we aim to in-
troduce the reader to this field and tackle the problem of accuracy under
sampling variation. We first introduce the current state-of-the-art of net-
work estimation. Second, we provide a rationale why researchers should
investigate the accuracy of psychological networks. Third, we describe how
bootstrap routines can be used to (A) assess the accuracy of estimated net-
work connections, (B) investigate the stability of centrality indices, and (C)
test whether network connections and centrality estimates for different vari-
ables differ from each other. We introduce two novel statistical methods: for
(B) the *correlation stability coefficient*, and for (C) the *bootstrapped differ-
ence test* for edge-weights and centrality indices. We conducted and present
simulation studies to assess the performance of both methods. Finally, we
developed the free R-package *bootnet* that allows for estimating psychological
networks in a generalized framework in addition to the proposed bootstrap
methods. We showcase *bootnet* in a tutorial, accompanied by R syntax, in
which we analyze a dataset of 359 women with posttraumatic stress disorder
available online.

3.1 Introduction

In the last five years, network research has gained substantial attention in psychological sciences (Borsboom & Cramer, 2013; Cramer et al., 2010). In this field of research, psychological behavior is conceptualized as a complex interplay of psychological and other components. To portray a potential structure in which these components interact, researchers have made use of *psychological networks*. Psychological networks consist of nodes representing observed variables, connected by edges representing statistical relationships. This methodology has gained substantial footing and has been used in various different fields of psychology, such as clinical psychology (e.g., Boschloo et al., 2015; Fried et al., 2015; McNally et al., 2015; Forbush, Siew, & Vitevitch, 2016), psychiatry (e.g., Isvoranu, van Borkulo, et al., 2016; Isvoranu, Borsboom, et al., 2016; van Borkulo et al., 2015), personality research (e.g., Costantini, Epskamp, et al., 2015; Costantini, Richetin, et al., 2015; Cramer, Sluis, et al., 2012), social psychology (e.g., Dalege et al., 2016), and quality of life research (Kossakowski et al., 2016).

These analyses typically involve two steps: (1) estimate a statistical model on data, from which some parameters can be represented as a weighted network between observed variables, and (2), analyze the weighted network structure using measures taken from graph theory (Newman, 2010) to infer, for instance, the most central nodes.[1] Step 1 makes psychological networks strikingly different from network structures typically used in graph theory, such as power grids (Watts & Strogatz, 1998), social networks (Wasserman & Faust, 1994) or ecological networks (Barzel & Biham, 2009) in which nodes represent entities (e.g., airports, people, organisms) and connections are generally observed and known (e.g., electricity lines, friendships, mutualistic relationships). In psychological networks, the strength of connection between two nodes is a parameter *estimated* from data. With increasing sample size, the parameters will be more accurately estimated (close to the true value). However, in the limited sample size psychological research typically has to offer, the parameters may not be estimated accurately, and in such cases, interpretation of the network and any measures derived from the network is questionable. Therefore, in estimating psychological networks, we suggest a third step is crucial: (3) assessing the accuracy of the network parameters and measures.

To highlight the importance of accuracy analysis in psychological networks, consider Figure 3.1 and Figure 3.2. Figure 3.1 (Panel A) shows a simulated network structure of 8 nodes in which each node is connected to two others in a *chain network*. The network model used is a Gaussian graphical model (Lauritzen, 1996), in which nodes represent observed variables and edges represent *partial correlation coefficients* between two variables after conditioning on all other variables in the dataset. A typical way of assessing the importance of nodes in this network is to compute *centrality indices* of the network structure (Costantini, Epskamp, et al., 2015; Newman, 2010; Opsahl et al., 2010). Three such measures are *node strength*, quantifying how well a node is directly connected to other nodes, *closeness*, quantifying how well a node is indirectly connected to other nodes, and *betweenness*,

---

[1] An introduction on the interpretation and inference of network models has been included in the online supplementary materials at http://sachaepskamp.com/files/bootnet_Supplementary.pdf.

A. Network structure   B. Centrality indices
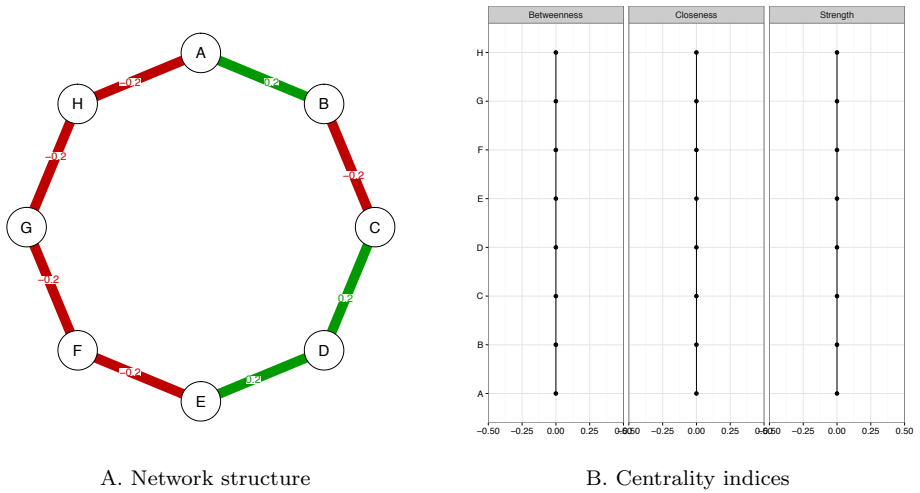
Figure 3.1: Simulated network structure (Panel A) and the importance of each node quantified in *centrality indices* (Panel B). The simulated network is a chain network in which each edge has the same absolute strength. The network model used was a Gaussian graphical model in which each edge represents partial correlation coefficients between two variables after conditioning on all other variables. Centrality indices are shown as standardized z-scores, which leads to all centrality indices to be equal to zero.

quantifying how important a node is in the average path between two other nodes. Figure 3.1 (Panel B) shows the centrality indices of the true network: all indices are exactly equal. We simulated a dataset of 500 individuals (typically regarded a moderately large sample size in psychology) using the network in Figure 3.1 and estimated a network structure based on the simulated data (as further described below). Results are presented in Figure 3.2; this is the observed network structure that researchers are usually faced with, without knowing the true network structure. Of note, this network closely resembles the true network structure.[2] As can be seen in Figure 3.2 (Panel B), however, centrality indices of the estimated network *do* differ from each other. Without knowledge on how *accurate* the centrality of these nodes are estimated, a researcher might in this case falsely conclude that nodes B and C play a much more important role in the network than other nodes.

Only few analyses so far have taken accuracy into account (e.g., Fried, Epskamp, et al., 2016), mainly because the methodology has not yet been worked out. This problem of accuracy is omnipresent in statistics. Imagine researchers employ a regression analysis to examine three predictors of depression severity, and identify one strong, one weak, and one unrelated regressor. If removing one of these three regressors, or adding a fourth one, substantially changes the regression

---

[2]Penalized maximum likelihood estimation used in this analysis typically leads to slightly lower parameter estimates on average. As a result, the absolute edge-weights in Figure 3.2 are all closer to zero than the absolute edge-weights in the true network in Figure 3.1.

A. Network structure

B. Centrality indices
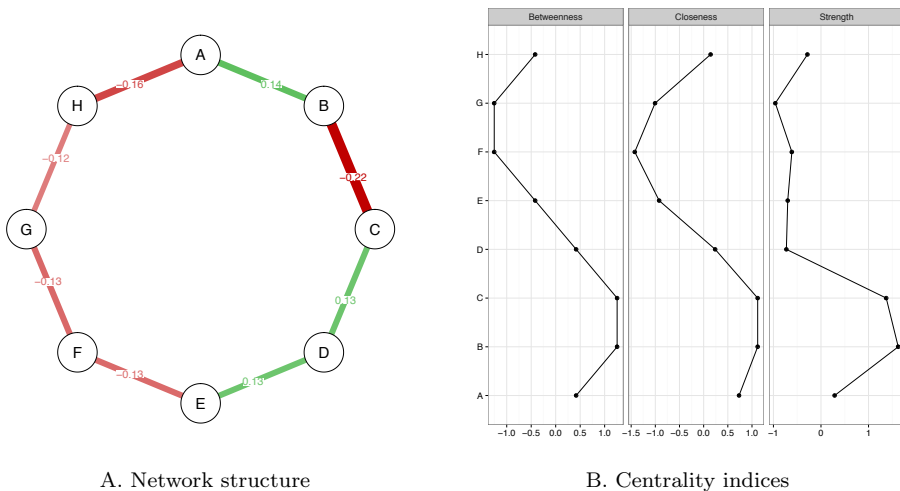
Figure 3.2: Estimated network structure based on a sample of 500 people simulated using the true model shown in Figure 3.1 (Panel A) and computed centrality indices (Panel B). Centrality indices are shown as standardized z-scores. Centrality indices show that nodes B and C are the most important nodes, even though the true model does not differentiate in importance between nodes.

coefficients of the other regressors, results are unstable and depend on specific decisions the researchers make, implying a problem of accuracy. The same holds for psychological networks. Imagine in a network of psychopathological symptoms that we find that symptom A has a much higher node strength than symptom B in a psychopathological network, leading to the clinical interpretation that A may be a more relevant target for treatment than the peripheral symptom B (Fried, Epskamp, et al., 2016). Clearly, this interpretation relies on the assumption that the centrality estimates are indeed different from each other. Due to the current uncertainty, there is the danger to obtain network structures sensitive to specific variables included, or sensitive to specific estimation methods. This poses a major challenge, especially when substantive interpretations such as treatment recommendations in the psychopathological literature, or the generalizability of the findings, are important. The current replication crisis in psychology (Open Science Collaboration, 2015) stresses the crucial importance of obtaining robust results, and we want the emerging field of psychopathological networks to start off on the right foot.

The remainder of the article is structured into three sections. In the first section, we give a brief overview of often used methods in estimating psychological networks, including an overview of open-source software packages that implement these methods available in the statistical programming environment R (R Core Team, 2016). In the second section, we outline a methodology to assess the accuracy of psychological network structures that includes three steps: (A) estimate confidence intervals (CIs) on the edge-weights, (B) assess the *stability* of central-

ity indices under observing subsets of cases, and (C) test for significant differences between edge-weights and centrality indices. We introduce the freely available R package, *bootnet*[3], that can be used both as a generalized framework to estimate various different network models as well as to conduct the accuracy tests we propose. We demonstrate the package's functionality of both estimating networks and checking their accuracy in a step-by-step tutorial using a dataset of 359 women with post-traumatic stress disorder (PTSD; Hien et al., 2009) that can be downloaded from the Data Share Website of the National Institute on Drug Abuse. Finally, in the last section, we show the performance of the proposed methods for investigating accuracy in three simulations studies. It is important to note that the focus of our tutorial is on cross-sectional network models that can readily be applied to many current psychological datasets. Many sources have already outlined the interpretation of probabilistic network models (e.g., Koller & Friedman, 2009; Lauritzen, 1996), as well as network inference techniques, such as centrality measures, that can be used once a network is obtained (e.g., Costantini, Epskamp, et al., 2015; Kolaczyk, 2009; Newman, 2004; Sporns, Chialvo, Kaiser, & Hilgetag, 2004).

To make this tutorial stand-alone readable for psychological researchers, we included a detailed description of how to interpret psychological network models as well as an overview of network measures in the online supplementary materials[4]. We hope that this tutorial will enable researchers to gauge the accuracy and certainty of the results obtained from network models, and to provide editors, reviewers, and readers of psychological network papers the possibility to better judge whether substantive conclusions drawn from such analyses are defensible.

3.2 Estimating Psychological Networks

As described in more detail in Chapter 1 and Chapter 2, a popular network model to use in estimating psychological networks is a pairwise Markov Random Field (PMRF; Costantini, Epskamp, et al., 2015; van Borkulo et al., 2014), on which the present chapter is focused. It should be noted, however, that the described methodology could be applied to other network models as well. A PMRF is a network in which nodes represent variables, connected by undirected edges (edges with no arrowhead) indicating conditional dependence between two variables; two variables that are not connected are independent after conditioning on all other variables. When data are multivariate normal, such a conditional independence would correspond to a partial correlation being equal to zero. Conditional independencies are also to be expected in many causal structures (Pearl, 2000). In cross-sectional observational data, causal networks (e.g. directed networks) are hard to estimate without stringent assumptions (e.g., no feedback loops). In addition, directed networks suffer from a problem of many equivalent models (e.g., a network $A \rightarrow B$ is not statistically distuinghuisable from a network $A \leftarrow B$; MacCallum et al., 1993, but see Mooij, Peters, Janzing, Zscheischler, & Schölkopf,

---

[3]CRAN link: `http://cran.r-project.org/package=bootnet`
Github link (developmental): `http://www.github.com/SachaEpskamp/bootnet`
[4]Included in this dissertation as Section 1.2

2016, when nonlinearities are included). PMRFs, however, are well defined and have no equivalent models (i.e., for a given PMRF, there exists no other PMRF that describes exactly the same statistical independence relationships for the set of variables under consideration). Therefore, they facilitate a clear and unambiguous interpretation of the edge-weight parameters as strength of unique associations between variables, which in turn may highlight potential causal relationships.

When the data are binary, the appropriate PMRF model to use is called the Ising model (van Borkulo et al., 2014), and requires binary data to be estimated. When the data follow a multivariate normal density, the appropriate PMRF model is called the Gaussian graphical model (GGM; Costantini, Epskamp, et al., 2015; Lauritzen, 1996), in which edges can be interpreted as *partial correlation coefficients*. The GGM requires an estimate of the covariance matrix as input,[5] for which polychoric correlations can also be used in case the data are ordinal (see Chapter 2). For continuous data that are not normally distributed, a transformation can be applied (e.g., by using the *nonparanormal transformation*; Liu, Han, Yuan, Lafferty, & Wasserman, 2012) before estimating the GGM. Finally, mixed graphical models can be used to estimate a PMRF containing both continuous and categorical variables (Haslbeck & Waldorp, 2016b).

Dealing with the problem of small N in psychological data. Estimating a PMRF features a severe limitation: the number of parameters to estimate grows quickly with the size of the network. In a 10-node network, 55 parameters (10 threshold parameters and $10 \times 9/2 = 45$ pairwise association parameters) need be estimated already. This number grows to 210 in a network with 20 nodes, and to 1275 in a 50-node network. To reliably estimate that many parameters, the number of observations needed typically exceeds the number available in characteristic psychological data. To deal with the problem of relatively small datasets, recent researchers using psychological networks have applied the 'least absolute shrinkage and selection operator' (LASSO; Tibshirani, 1996). This technique is a form of *regularization*. The LASSO employs such a regularizing penalty by limiting the total sum of absolute parameter values—thus treating positive and negative edge-weights equally—leading many edge estimates to shrink to exactly zero and dropping out of the model. As such, the LASSO returns a *sparse* (or, in substantive terms, conservative) network model: only a relatively small number of edges are used to explain the covariation structure in the data. Because of this sparsity, the estimated models become more interpretable. The LASSO utilizes a tuning parameter to control the degree to which regularization is applied. This tuning parameter can be selected by minimizing the Extended Bayesian Information Criterion (EBIC; Chen & Chen, 2008). Model selection using the EBIC has been shown to work well in both estimating the Ising model (Foygel Barber & Drton, 2015; van Borkulo et al., 2014) and the GGM (Foygel & Drton, 2010). The remainder of this chapter focuses on the GGM estimation method proposed

---

[5] While the GGM requires a covariance matrix as input, it is important to note that the model itself is based on the (possibly sparse) *inverse* of the covariance matrix. Therefore, the network shown does not show marginal correlations (regular correlation coefficients between two variables). The inverse covariance matrix instead encodes *partial* correlations.

by Foygel & Drton (2010; see also Chapter 2, for a detailed introduction of this method for psychological researchers).

Estimating regularized networks in R is straightforward. For the Ising model, LASSO estimation using EBIC has been implemented in the *IsingFit* package (van Borkulo et al., 2014). For GGM networks, a well established and fast algorithm for estimating LASSO regularization is the *graphical LASSO* (glasso; Friedman et al., 2008), which is implemented in the package *glasso* (Friedman et al., 2014). The *qgraph* package utilizes *glasso* in combination with EBIC model selection to estimate a regularized GGM. Alternatively, the *huge* (Zhao et al., 2015) and *parcor* (Krämer et al., 2009) packages implement several regularization methods— including also glasso with EBIC model selection—to estimate a GGM. Finally, mixed graphical models have been implemented in the *mgm* package (Haslbeck & Waldorp, 2016a).

3.3 Network Accuracy

The above description is an overview of the current state of network estimation in psychology. While network inference is typically performed by assessing edge strengths and node centrality, little work has been done in investigating how accurate these inferences are. In this section, we outline several methods that should routinely be applied after a network has been estimated. These methods will follow three steps: (A) estimation of the accuracy of edge-weights, by drawing bootstrapped CIs; (B) investigating the stability of (the order of) centrality indices after observing only portions of the data; and (C) performing bootstrapped difference tests between edge-weights and centrality indices to test whether these differ significantly from each other. We introduced these methods in decreasing order of importance: while (A) should always be performed, a researcher not interested in centrality indices might not perform other steps, whereas a researcher not interested in testing for differences might only perform (A) and (B). Simulation studies have been conducted to assess the performance of these methods, which are reported in a later section in the chapter.

Edge-weight Accuracy

To assess the variability of edge-weights, we can estimate a CI: in 95% of the cases such a CI will contain the true value of the parameter. To construct a CI, we need to know the *sampling distribution* of the statistic of interest. While such sampling distributions can be difficult to obtain for complicated statistics such as centrality measures, there is a straight-forward way of constructing CIs many statistics: *bootstrapping* (Efron, 1979). Bootstrapping involves repeatedly estimating a model under sampled or simulated data and estimating the statistic of interest. Following the bootstrap, a $1 - \alpha$ CI can be approximated by taking the interval between quantiles $1/2\alpha$ and $1 - 1/2\alpha$ of the bootstrapped values. We term such an interval a *bootstrapped CI*. Bootstrapping edge-weights can be done in two ways: using non-parametric bootstrap and parametric bootstrap (Bollen & Stine, 1992). In *non-parametric* bootstrapping, observations in the data are

resampled with replacement to create new plausible datasets, whereas *parametric* bootstrapping samples new observations from the parametric model that has been estimated from the original data; this creates a series of values that can be used to estimate the sampling distribution. Bootstrapping can be applied as well to LASSO regularized statistics (Hastie et al., 2015).

Non-parametric bootstrapping can always be applied, whereas parametric bootstrapping requires a parametric model of the data. When we estimate a GGM, data can be sampled by sampling from the multivariate normal distribution through the use of the R package *mvtnorm* (Genz et al., 2008); to sample from the Ising model, we have developed the R package *IsingSampler* (Epskamp, 2014). Using the GGM model, the parametric bootstrap samples continuous multivariate normal data—an important distinction from ordinal data if the GGM was estimated using polychoric correlations. Therefore, we advise the researcher to use the non-parametric bootstrap when handling ordinal data. Furthermore, when LASSO regularization is used to estimate a network, the edge-weights are on average made smaller due to shrinkage, which biases the parametric bootstrap. The non-parametric bootstrap is in addition fully data-driven, whereas the parametric bootstrap is more theory driven. As such, we will only discuss the non-parametric bootstrap in this chapter and advice the researcher to only use parametric bootstrap when no regularization is used and if the non-parametric results prove unstable or to check for correspondence of bootstrapped CIs between both methods.

It is important to stress that the bootstrapped results should *not* be used to test for significance of an edge being different from zero. While unreported simulation studies showed that observing if zero is in the bootstrapped CI does function as a valid null-hypothesis test (the null-hypothesis is rejected less than α when it is true), the utility of testing for significance in LASSO regularized edges is questionable. In the case of partial correlation coefficients, without using LASSO the sampling distribution is well known and p-values are readily available. LASSO regularization aims to estimate edges that are not needed to be exactly zero. Therefore, observing that an edge is not set to zero already indicates that the edge is sufficiently strong to be included in the model. In addition, as later described in this chapter, applying a correction for multiple testing is not feasible, In sum, the edge-weight bootstrapped CIs should not be interpreted as significance tests to zero, but only to show the accuracy of regularized edge-weights and to compare edges to one-another.

Centrality Stability

While the bootstrapped CIs of edge-weights can be constructed using the bootstrap, we discovered in the process of this research that constructing CIs for centrality indices is far from trivial. As discussed in more detail in the online supplementary materials, both estimating centrality indices based on a sample and bootstrapping centrality indices result in biased sampling distributions, and thus the bootstrap cannot readily be used to construct true 95% CIs even without regularization. To allow the researcher insight in the accuracy of the found centralities, we suggest to investigate the stability of the order of centrality indices based on *subsets* of the data. With *stability*, we indicate if the order of centrality indices re-

mains the same after re-estimating the network with fewer cases or nodes. A *case* indicates a single observation of all variables (e.g., a person in the dataset) and is represented by *rows* of the dataset. Nodes, on the other hand, indicate *columns* of the dataset. Taking subsets of cases in the dataset employs the so called *m out of n bootstrap*, which is commonly used to remediate problems with the regular bootstrap (Chernick, 2011). Applying this bootstrap for various proportions of cases to drop can be used to assess the correlation between the original centrality indices and those obtained from subsets. If this correlation completely changes after dropping, say, 10% of the cases, then interpretations of centralities are prone to error. We term this framework the *case-dropping subset bootstrap*. Similarly, one can opt to investigate the stability of centrality indices after dropping nodes from the network (*node-dropping subset bootstrap*; Costenbader & Valente, 2003), which has also been implemented in *bootnet* but is harder to interpret (dropping 50% of the nodes leads to entirely different network structures). As such, we only investigate stability under case-dropping, while noting that the below described methods can also be applied to node-dropping.

To quantify the stability of centrality indices using subset bootstraps, we propose a measure we term the *correlation stability coefficient*, or short, the *CS*-coefficient. Let $CS(\text{cor} = 0.7)$ represent the maximum proportion of cases that can be dropped, such that with 95% probability the correlation between original centrality indices and centrality of networks based on subsets is 0.7 or higher. The value of 0.7 can be changed according to the stability a researcher is interested in, but is set to 0.7 by default as this value has classically been interpreted as indicating a very large effect in the behavioral sciences (Cohen, 1977). The simulation study below showed that to interpret centrality differences the *CS*-coefficient should not be below 0.25, and preferably above 0.5. While these cutoff scores emerge as recommendations from this simulation study, however, they are somewhat arbitrary and should not be taken as definite guidelines.

Testing for Significant Differences

In addition to investigating the accuracy of edge weights and the stability of the order of centrality, researchers may wish to know whether a specific edge *A–B* is significantly larger than another edge *A–C*, or whether the centrality of node A is significantly larger than that of node B. To that end, the bootstrapped values can be used to test if two edge-weights or centralities significantly differ from one-another. This can be done by taking the *difference* between bootstrap values of one edge-weight or centrality and another edge-weight or centrality, and constructing a bootstrapped CI around those difference scores. This allows for a null-hypothesis test if the edge-weights or centralities differ from one-another by checking if zero is in the bootstrapped CI (Chernick, 2011). We term this test the *bootstrapped difference test*.

As the bootstraps are functions of complicated estimation methods, in this case LASSO regularization of partial correlation networks based on polychoric correlation matrices, we assessed the performance of the bootstrapped difference test for both edge-weights and centrality indices in two simulation studies below. The edge-weight bootstrapped difference test performs well with Type I error rate

close to the significance level (α), although the test is slightly conservative at low sample sizes (i.e, due to edge-weights often being set to zero, the test has a Type I error rate somewhat less than α). When comparing two centrality indices, the test also performs as a valid, albeit somewhat conservative, null-hypothesis test with Type I error rate close to or less than α. However, this test does feature a somewhat lower level of power in rejecting the null-hypothesis when two centralities do differ from one-another.

A null-hypothesis test, such as the bootstrapped difference test, can only be used as evidence that two values differ from one-another (and even then care should be taken in interpreting its results; e.g., Cohen, 1994). *Not* rejecting the null-hypothesis, however, does not necessarily constitute evidence for the null-hypothesis being true (Wagenmakers, 2007). The slightly lower power of the bootstrapped difference test implies that, at typical sample sizes used in psychological research, the test will tend to find fewer significant differences than actually exist at the population level. Researchers should therefore not routinely take non-significant centralities as evidence for centralities being equal to each other, or for the centralities not being accurately estimated. Furthermore, as described below, applying a correction for multiple testing is not feasible in practice. As such, we advise care when interpreting the results of bootstrapped difference tests.

A note on multiple testing. The problem of performing multiple significance tests is well known in statistics. When one preforms two tests, both at $\alpha = 0.05$, the probability of finding at least *one* false significant result (Type I error) is *higher* than 5%. As a result, when performing a large number of significance tests, even when the null-hypothesis is true in all tests one would likely find several significant results purely by chance. To this end, researchers often apply a *correction for multiple testing*. A common correction is the 'Bonferroni correction' (Bland & Altman, 1995), in which α is divided by the number of tests. To test, for example, differences between all edge-weights of a 20-node network requires 17,955 tests, leading to a Bonferroni corrected significance level of 0.000003.[6] Testing at such a low significance level is *not* feasible with the proposed bootstrap methods, for three reasons:

1. The distribution of such LASSO regularized parameters is far from normal (Pötscher & Leeb, 2009), and as a result approximate p-values cannot be obtained from the bootstraps. This is particularly important for extreme significance levels that might be used when one wants to test using a correction for multiple testing. It is for this reason that this chapter does not mention bootstrapping p-values and only investigates null-hypothesis tests by using bootstrapped CIs.

2. When using bootstrapped CIs with N_B bootstrap samples, the widest interval that can be constructed is the interval between the two most extreme bootstrap values, corresponding to $\alpha = 2/N_B$. With 1,000 bootstrap samples, this corresponds to $\alpha = 0.002$. Clearly, this value is much higher than

---

[6]One might instead only test for difference in edges that were estimated to be non-zero with the LASSO. However, doing so still often leads to a large number of tests.

0.000003 mentioned above. Taking the needed number of bootstrap samples for such small significance levels is computationally challenging and not feasible in practice.

3. In significance testing there is always interplay of Type I and Type II error rates: when one goes down, the other goes up. As such, reducing the Type I error rate increases the Type II error rate (not rejecting the null when the alternative hypothesis is true), and thus reduces statistical power. In the case of $\alpha = 0.000003$, even if we could test at this significance level, we would likely find no significant differences due to the low statistical power.

As such, Bonferroni corrected difference tests are still a topic of future research.

Summary

In sum, the non-parametric (resampling rows from the data with replacement) bootstrap can be used to assess the *accuracy* of network estimation, by investigating the sampling variability in edge-weights, as well as to test if edge-weights and centrality indices significantly differ from one-another using the bootstrapped difference test. Case-dropping subset bootstrap (dropping rows from the data), on the other hand, can be used to assess the *stability* of centrality indices, how well the order of centralities are retained after observing only a subset of the data. This stability can be quantified using the CS-coefficient. The R code in the online supplementary materials show examples of these methods on the simulated data in Figure 3.1 and Figure 3.2. As expected from Figure 3.1, showing that the true centralities did not differ, bootstrapping reveals that none of the centrality indices in Figure 3.2 significantly differ from one-another. In addition, node strength ($CS(\text{cor} = 0.7) = 0.13$), closeness ($CS(\text{cor} = 0.7) = 0.05$) and betweenness ($CS(\text{cor} = 0.7) = 0.05$) were far below the thresholds that we would consider stable. Thus, the novel bootstrapping methods proposed and implemented here showed that the differences in centrality indices presented in Figure 3.2 were not interpretable as true differences.

3.4 Tutorial

In this section, we showcase the functionality of the *bootnet* package for estimating network structures and assessing their accuracy. We do so by analyzing a dataset ($N = 359$) of women suffering from posttraumatic stress disorder (PTSD) or sub-threshold PTSD. The *bootnet* package includes the bootstrapping methods, CS-coefficient and bootstrapped difference tests as described above. In addition, *bootnet* offers a wide range of plotting methods. After estimating nonparametric bootstraps, *bootnet* produces plots that show the bootstrapped CIs of edge-weights or which edges and centrality indices significantly differ from one-another. After estimating subset bootstrap, *bootnet* produces plots that show the correlation of centrality indices under different levels of subsetting (Costenbader & Valente, 2003). In addition to the correlation plot, *bootnet* can be used to plot the average

Default set	R chain
EBICglasso	Data %>% qgraph::cor_auto %>% qgraph::EBICglasso
pcor	Data %>% qgraph::cor_auto %>% corpcor::cor2pcor
IsingFit	Data %>% bootnet::binarize %>% IsingFit::IsingFit
IsingLL	Data %>% bootnet::binarize %>% IsingSampler::EstimateIsing(method = \ll")
huge	Data %>% as.matrix %>% na.omit %>% huge::huge.npn %>% huge::huge(method = \glasso") %>% huge::huge.select(criterion = \ebic")
adalasso	Data %>% parcor::adalasso.net

Table 3.1: R chains to estimate network models from data. The default sets `"EBICglasso"`, `"pcor"`, `"huge"` and `"adalasso"` estimate a Gaussian graphical model and the default sets `"IsingFit"` and `"IsingLL"` estimate the Ising model. The notation `package::function` indicates that the function after the colons comes from the package before the colons. Chains are schematically represented using *magrittr* chains: Whatever is on the left of `%>%` is used as first argument to the function on the right of this operator. Thus, the first chain corresponding to `"EBICglasso"` can also be read as `qgraph::EBICglasso(qgraph::cor_auto(Data))`.

estimated centrality index for each node under different sampling levels, giving more detail on the order of centrality under different subsetting levels.

With *bootnet*, users can not only perform accuracy and stability tests, but also flexibly estimate a wide variety of network models in R. The estimation technique can be specified as a *chain* of R commands, taking the data as input and returning a network as output. In bootnet, this chain is broken in several phases: data preparation (e.g., correlating or binarizing), model estimation (e.g., glasso) and network selection. The *bootnet* package has several *default sets*, which can be assigned using the `default` argument in several functions. These default sets can be used to easily specify the most commonly used network estimation procedures. Table 3.1 gives an overview of the default sets and the corresponding R functions called.[7]

Example: Post-traumatic Stress Disorder

To exemplify the usage of *bootnet* in both estimating and investigating network structures, we use a dataset of 359 women enrolled in community-based substance abuse treatment programs across the United States (study title: Women's Treatment for Trauma and Substance Use Disorders; study number: NIDA-CTN-0015).[8] All participants met the criteria for either PTSD or sub-threshold PTSD, according to the DSM-IV-TR (American Psychiatric Association, 2000). Details of the sample, such as inclusion and exclusion criteria as well as demographic variables, can be found elsewhere (Hien et al., 2009). We estimate the network

---

[7]The notation makes use of notation introduced by the *magrittr* R package (Bache & Wickham, 2014)

[8]https://datashare.nida.nih.gov/protocol/nida-ctn-0015

using the 17 PTSD symptoms from the PTSD Symptom Scale-Self Report (PSS-SR; Foa, Riggs, Dancu, & Rothbaum, 1993). Participants rated the frequency of endorsing these symptoms on a scale ranging from 0 (not at all) to 3 (at least 4 or 5 times a week).

Network estimation. Following the steps in the online supplementary materials, the data can be loaded into R in a data frame called `Data`, which contains the frequency ratings at the baseline measurement point. We will estimate a Gaussian graphical model, using the graphical LASSO in combination with EBIC model selection as described above (Foygel & Drton, 2010). This procedure requires an estimate of the variance-covariance matrix and returns a parsimonious network of *partial correlation coefficients*. Since the PTSD symptoms are ordinal, we need to compute a polychoric correlation matrix as input. We can do so using the `cor_auto` function from the *qgraph* package, which automatically detects ordinal variables and utilizes the R-package `lavaan` (Rosseel, 2012) to compute polychoric (or, if needed, polyserial and Pearson) correlations. Next, the `EBICglasso` function from the *qgraph* package can be used to estimate the network structure, which uses the *glasso* package for the actual computation (Friedman et al., 2014). In *bootnet*, as can be seen in Table 3.1, the `"EBICglasso"` default set automates this procedure. To estimate the network structure, one can use the `estimateNetwork` function:

```
library("bootnet")
Network <- estimateNetwork(Data, default = "EBICglasso")
```

Next, we can plot the network using the plot method:

```
plot(Network, layout = "spring", labels = TRUE)
```

The `plot` method uses *qgraph* to plot the network. Figure 3.3 (Panel A) shows the resulting network structure, which is parsimonious due to the LASSO estimation; the network only has 78 non-zero edges out of 136 possible edges. A description of the node labels can be seen in Table 3.2. Especially strong connections emerge among Node 3 (being jumpy) and Node 4 (being alert), Node 5 (cut off from people) and Node 11 (interest loss), and Node 16 (upset when reminded of the trauma) and Node 17 (upsetting thoughts/images). Other connections are absent, for instance between Node 7 (irritability) and Node 15 (reliving the trauma); this implies that these symptoms can be statistically independent when conditioning on all other symptoms (their partial correlation is zero) or that there was not sufficient power to detect an edge between these symptoms.

Computing centrality indices. To investigate centrality indices in the network, we can use the `centralityPlot` function from the *qgraph* package:

```
library("qgraph")
centralityPlot(Network)
```

A. Network structure   B. Centrality indices
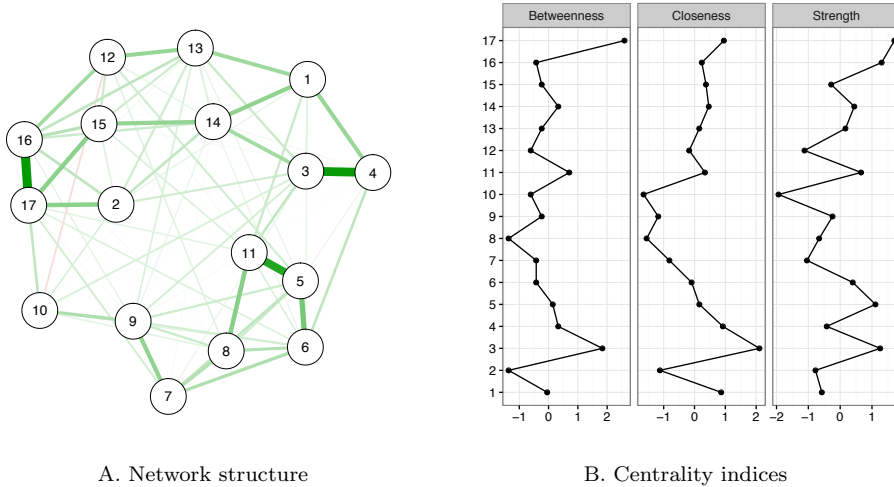
Figure 3.3: Estimated network structure of 17 PTSD symptoms (Panel A) and the corresponding centrality indices (Panel B). Centrality indices are shown as standardized z-scores. The network structure is a Gaussian graphical model, which is a network of partial correlation coefficients.

ID	Variable
1	Avoid reminds of the trauma
2	Bad dreams about the trauma
3	Being jumpy or easily startled
4	Being over alert
5	Distant or cut off from people
6	Feeling emotionally numb
7	Feeling irritable
8	Feeling plans won't come true
9	Having trouble concentrating
10	Having trouble sleeping
11	Less interest in activities
12	Not able to remember
13	Not thinking about trauma
14	Physical reactions
15	Reliving the trauma
16	Upset when reminded of trauma
17	Upsetting thoughts or images

Table 3.2: Node IDs and corresponding symptom names of the 17 PTSD symptoms.

The resulting plot is shown in Figure 3.3 (Panel B). It can be seen that nodes differ quite substantially in their centrality estimates. In the network, Node 17 (upsetting thoughts/images) has the highest strength and betweenness and Node 3 (being jumpy) has the highest closeness. However, without knowing the accuracy of the network structure and the stability of the centrality estimates, we cannot conclude whether the differences of centrality estimates are interpretable or not.

Edge-weight accuracy. The `bootnet` function can be used to perform the bootstrapping methods described above. The function can be used in the same way as the `estimateNetwork` function, or can take the output of the `estimateNetwork` function to run the bootstrap using the same arguments. By default, the nonparametric bootstrap with 1,000 samples will be used. This can be overwritten using the `nBoots` argument, which is used below to obtain more smooth plots.[9] The `nCores` argument can be used to speed up bootstrapping and use multiple computer cores (here, eight cores are used):

```
boot1 <- bootnet(Network, nBoots = 2500,  nCores = 8)
```

The `print` method of this object gives an overview of characteristics of the sample network (e.g., the number of estimated edges) and tips for further investigation, such as how to plot the estimated sample network or any of the bootstrapped networks. The `summary` method can be used to create a summary table of certain statistics containing quantiles of the bootstraps.

The `plot` method can be used to show the bootstrapped CIs for estimated edge parameters:

```
plot(boot1, labels = FALSE, order = "sample")
```

Figure 3.4 shows the resulting plots and reveals sizable bootstrapped CIs around the estimated edge-weights, indicating that many edge-weights likely do not significantly differ from one-another. The generally large bootstrapped CIs imply that interpreting the order of most edges in the network should be done with care. Of note, the edges 16 (upset when reminded of the trauma) – 17 (upsetting thoughts/images), 3 (being jumpy) – 4 (being alert) and 5 (feeling distant) – 11 (loss of interest), are reliably the three strongest edges since their bootstrapped CIs do not overlap with the bootstrapped CIs of any other edges.[10]

Centrality stability. We can now investigate the stability of centrality indices by estimating network models based on subsets of the data. The case-dropping bootstrap can be used by using `type = "case"`:

```
boot2 <- bootnet(Network, nBoots = 2500, type = "case",
nCores = 8)
```

---

[9]Using many bootstrap samples, such as the 2,500 used here, might result in memory problems or long computation time. It is advisable to first use a small number of samples (e.g., 10) and then try more. The simulations below show that 1,000 samples may often be sufficient.

[10]As with any CI, non-overlapping CIs indicate two statistics significantly differ at the given significance level. The reverse is not true; statistics with overlapping CIs might still significantly differ.
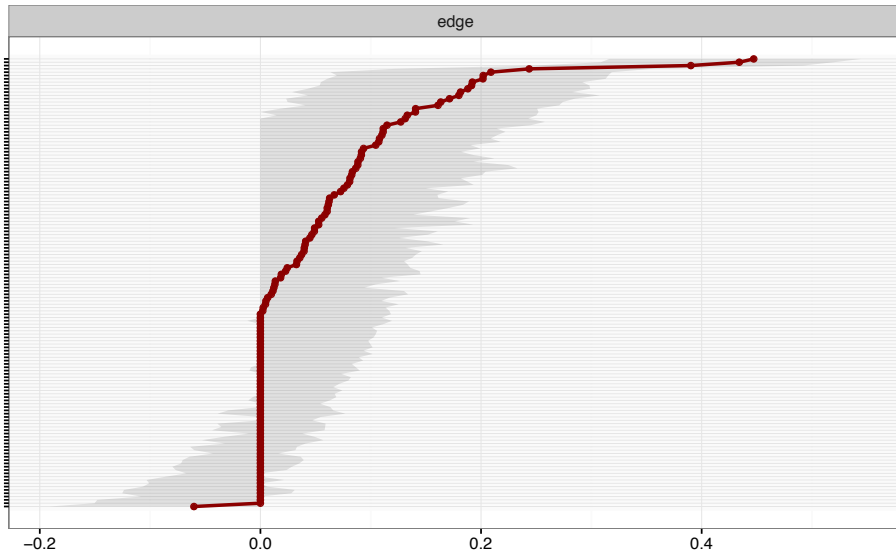
Figure 3.4: Bootstrapped confidence intervals of estimated edge-weights for the estimated network of 17 PTSD symptoms. The red line indicates the sample values and the gray area the bootstrapped CIs. Each horizontal line represents one edge of the network, ordered from the edge with the highest edge-weight to the edge with the lowest edge-weight. In the case of ties (for instance, multiple edge-weights were estimated to be exactly 0), the mean of the bootstrap samples was used in ordering the edges. y-axis labels have been removed to avoid cluttering.

To plot the stability of centrality under subsetting, the plot method can again be used:

```
plot(boot2)
```

Figure 3.5 shows the resulting plot: the stability of closeness and betweenness drop steeply while the stability of node strength is better. This stability can be quantified using the CS-coefficient, which quantifies the maximum proportion of cases that can be dropped to retain, with 95% certainty, a correlation with the original centrality of higher than (by default) 0.7. This coefficient can be computed using the `corStability` function:

```
corStability(boot2)
```

The CS-coefficient indicates that betweenness ($CS(\text{cor} = 0.7) = 0.05$) and ($CS(\text{cor} = 0.7) = 0.05$) closeness are not stable under subsetting cases. Node strength performs better ($CS(\text{cor} = 0.7) = 0.36$), but does not reach the cutoff of 0.5 from our simulation study required consider the metric stable. Therefore, we conclude that the order of node strength is interpretable with some care, while the orders of betweenness and closeness are not.
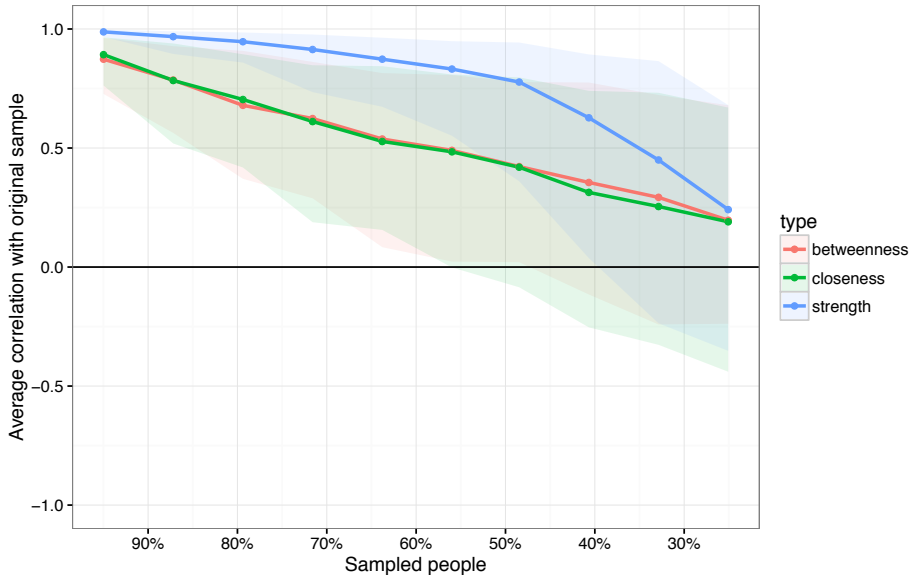
Figure 3.5: Average correlations between centrality indices of networks sampled with persons dropped and the original sample. Lines indicate the means and areas indicate the range from the 2.5th quantile to the 97.5th quantile.

Testing for significant differences. The `differenceTest` function can be used to compare edge-weights and centralities using the bootstrapped difference test. This makes use of the non-parametric bootstrap results (here named `boot1`) rather than the case-dropping bootstrap results. For example, the following code tests if Node 3 and Node 17 differ in node strength centrality:

```
differenceTest(boot1, 3, 17, "strength")
```

The results show that these nodes do not differ in node strength since the bootstrapped CI includes zero (CI: $-0.17, 0.37$). The plot method can be used to plot the difference tests between all pairs of edges and centrality indices. For example, the following code plots the difference tests of node strength between all pairs of edge-weights:

```
plot(boot1, "edge", plot = "difference", onlyNonZero = TRUE,
     order = "sample")
```

In which the `plot` argument has to be used because the function normally defaults to plotting bootstrapped CIs for edge-weights, the `onlyNonZero` argument sets so that only edges are shown that are nonzero in the estimated network, and `order = "sample"` orders the edge-weights from the most positive to the most negative edge-weight in the sample network. We can use a similar code for comparing node strength:

```
plot(boot1, "strength")
```

In which we did not have to specify the `plot` argument as it is set to the `"difference"` by default when the statistic is a centrality index.

The resulting plots are presented in Figure 3.6. Panel A shows that many edges cannot be shown to significantly differ from one-another, except for the previously mentioned edges 16 (upset when reminded of the trauma) – 17 (upsetting thoughts/images), 3 (being jumpy) – 4 (being alert) and 5 (feeling distant) – 11 (loss of interest), which significantly differ from most other edges in the network. Panel B shows that most node strengths cannot be shown to significantly differ from each other. The node with the largest strength, Node 17, is significantly larger than almost half the other nodes. Furthermore, Node 7 and Node 10 and also feature node strength that is significantly larger than some of the other nodes. In this dataset, no significant differences were found between nodes in both betweenness and closeness (not shown). For both plots it is important to note that *no* correction for multiple testing was applied.

3.5   Simulation Studies

We conducted three simulation studies to assess the performance of the methods described above. In particular, we investigated the performance of (1) the CS-coefficient and the bootstrapped difference test for (2) edge-weights and (3) centrality indices. All simulation studies use networks of 10 nodes. The networks were used as partial correlation matrices to generate multivariate normal data, which were subsequently made ordinal with four levels by drawing random thresholds; we did so because most prior network papers estimated networks on ordinal data (e.g., psychopathological symptom data). We varied sample size between 100, 250, 500, 1,000, 2,500 and 5,000, and replicated every condition 1,000 times. We estimated Gaussian graphical models, using the graphical LASSO in combination with EBIC model selection (Foygel & Drton, 2010; see also Chapter 2), using polychoric correlation matrices as input. Each bootstrap method used 1,000 bootstrap samples. In addition, we replicated every simulation study with 5-node and 20-node networks as well, which showed similar results and were thus not included in this chapter to improve clarity.

CS-**coefficients.**   We assessed the CS-coefficient in a simulation study for two cases where: networks where centrality did not differ between nodes, and networks where centrality did differ. We simulated chain networks as shown in Figure 3.1 consisting of 10 nodes, 50% negative edges and all edge-weights set to either 0.25 or -0.25. Next, we randomly rewired edges as described by Watts and Strogatz (1998) with probability 0, 0.1, 0.5 or 1. A rewiring probability of 0.5 indicates that every edge had a 50% chance of being rewired to another node, leading to a different network structure than the chain graph. This procedure creates a range of networks, ranging from chain graphs in which all centralities are equal (rewiring probability = 0) to random graphs in which all centralities may be different (rewiring probability = 1). Every condition (rewiring probability × sample size) was replicated 1,000 times, leading to 24,000 simulated datasets. On each of

B. Significant differences in node strength

A. Significant differences in edge-weights
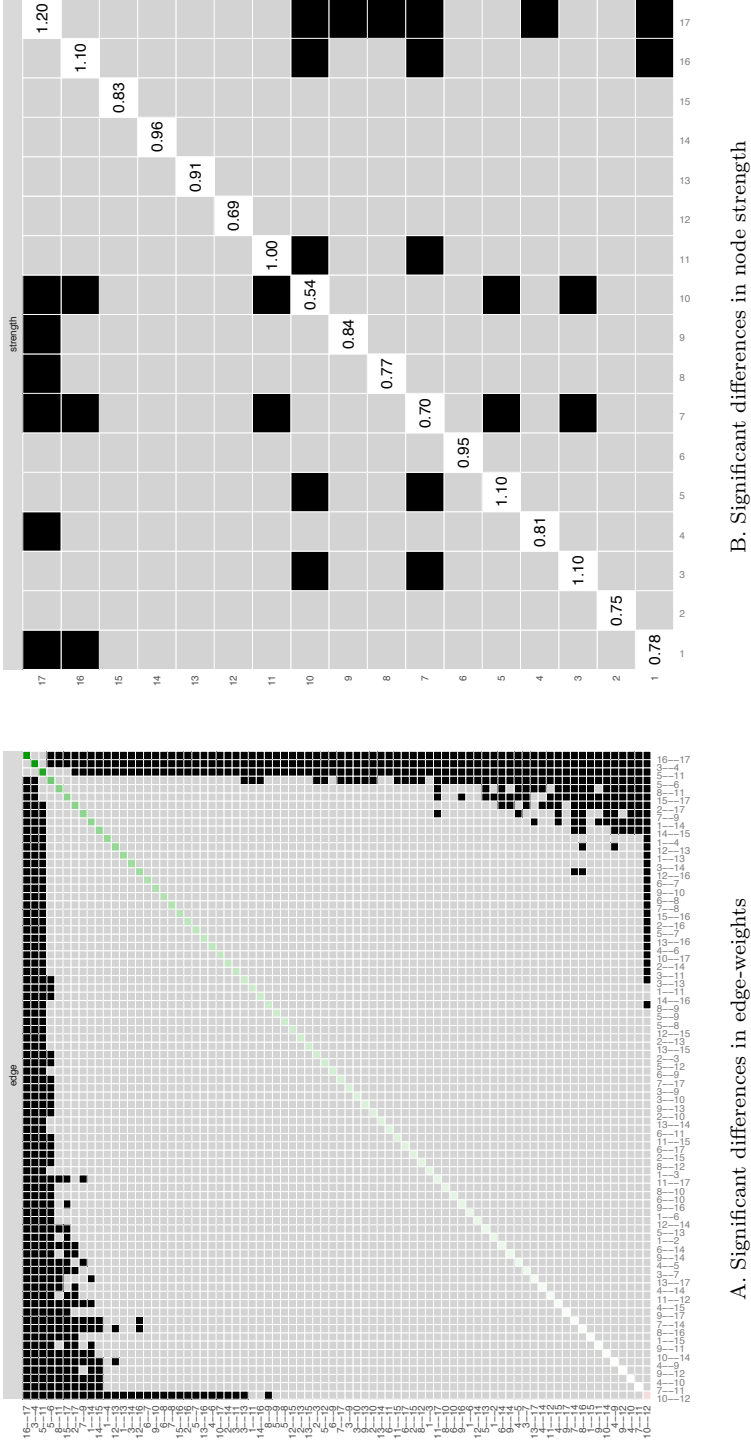
Figure 3.6: Bootstrapped difference tests ($\alpha = 0.05$) between edge-weights that were non-zero in the estimated network (Panel A) and node strength of the 17 PTSD symptoms (Panel B). Gray boxes indicate nodes or edges that do not differ significantly from one-another and black boxes represent nodes or edges that do differ significantly from one-another. Colored boxes in the edge-weight plot correspond to the color of the edge in Figure 3.3, and white boxes in the centrality plot show the value of node strength.

these datasets, case-dropping bootstrap was performed and the CS-coefficient was computed. Figure 3.7 shows the results, showing that the CS-coefficient remains low in networks in which centrality does not differ and rises as a function of sample size in networks in which centralities do differ. It can be seen that under a model in which centralities do not differ the CS-coefficient remains stable as sample size increases and stays mostly below .5, and roughly 75% stays below 0.25. Therefore, to interpret centrality differences the CS-coefficient should not be below 0.25, and preferably above 0.5.

Edge-weight bootstrapped difference test. We ran a second simulation study to assess the performance of the bootstrapped difference test for edge-weights. In this simulation study, chain networks were constructed consisting of 10 nodes in which all edge-weights were set to 0.3. Sample size was again varied between 100, 250, 500, 1,000, 2,500 and 5,000 and each condition was again replicated 1,000 times, leading to 6,000 total simulated datasets. Data were made ordinal and regularized partial correlation networks were estimated in the same manner as in the previous simulation studies. We only compared edges that were nonzero in the true network (thus, edges with a weight of 0.3 that were not different from one-another), and we investigated the rejection rate under different levels of α: 0.05, 0.01, 0.001 and 0.0001. For every significance level, the expected significance level given a certain number of bootstrap samples (in this case 1,000) was computed using the following R code:

```
alpha <- 0.05
mean(replicate(10000,quantile(runif(1000),alpha/2)) +
  (1 - replicate(10000,quantile(runif(1000),1-alpha/2))))
```

Figure 3.8 shows that rejection rate converged on the expected rejection rate with higher samples, and was lower than the expected rejection rate in the low sample condition of $N = 100$—a result of the LASSO pulling many edge-weights to zero in low sample sizes.

Centrality bootstrapped difference test. We conducted a third simulation study to assess the performance of the bootstrapped difference test for centrality indices. The design was the same as the first simulation study, leading to 24,000 total simulated datasets. We performed the bootstrapped difference test to all pairs of nodes in all networks and computed the rate of rejecting the null-hypothesis of centralities being equal. Figure 3.9 shows the results of this simulation study. It can be seen that the average rate of rejecting the null-hypothesis of two centrality indices being equal under a chain-network such as shown in Figure 3.1 stays below 0.05 at all sample sizes for all centrality indices. As such, checking if zero is in the bootstrapped CI on differences between centralities is a valid null-hypothesis test. Figure 3.9, however, also shows that the rejection rate often is below 0.05, leading to a reduced power in the test. As such, finding true differences in centrality might require a larger sample size. When centralities differ (rewiring probability > 0), power to detect differences goes up as a function of sample size. Unreported simulation studies showed that using Pearson or Spearman correlations on ordinal
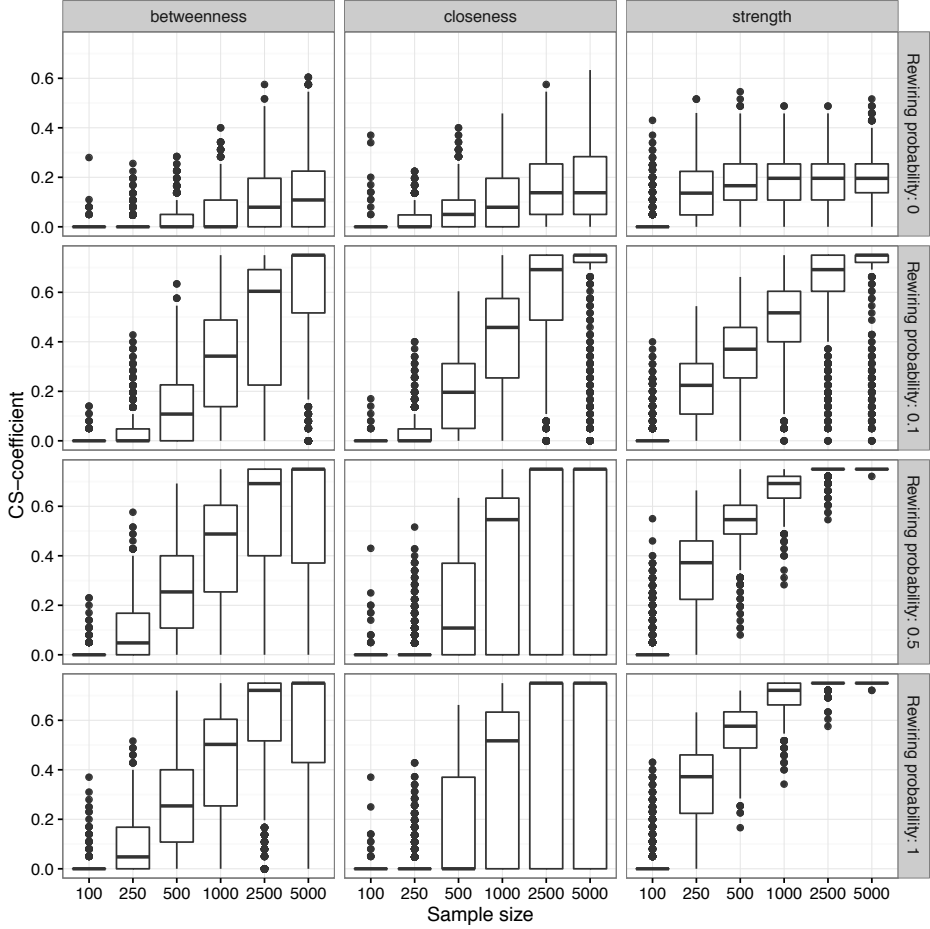
Figure 3.7: Simulation results showing the CS-coefficient of 24,000 simulated datasets. Datasets were generated using chain networks (partial correlations) of 10 nodes with edge-weights set to 0.25 or -0.25. Edges were randomly rewired to obtain a range from networks ranging from networks in which all centralities are equal to networks in which all centralities differ. The CS-coefficient quantifies the maximum proportion of cases that can be dropped at random to retain, with 95% certainty, a correlation of at least 0.7 with the centralities of the original network. Boxplots show the distribution of CS-coefficients obtained in the simulations. For example, plots on top indicate that the CS-coefficient mostly stays below 0.2 when centralities do not differ from one-another (chain graph as shown in Figure 3.1).
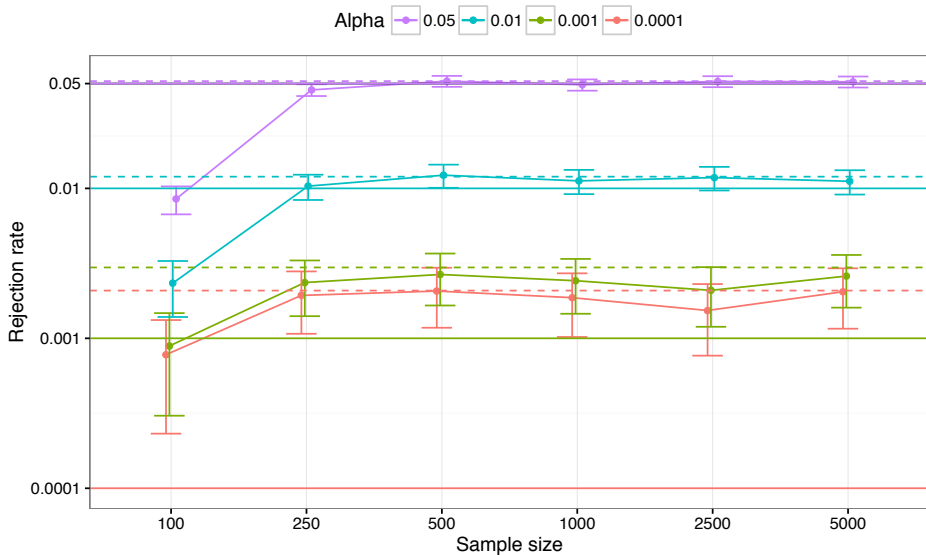
Figure 3.8: Simulation results showing the rejection rate of the bootstrapped difference test for edge-weights on 6,000 simulated datasets. Datasets were generated using chain networks (partial correlations) of 10 nodes with edge-weights set to 0.3. Only networks that were nonzero in the true network were compared to one-another. Lines indicate the proportion of times that two random edge-weights were significantly different (i.e., the null-hypothesis was rejected) and their CI (plus and minus 1.96 times the standard error). Solid horizontal lines indicate the intended significance level and horizontal dashed line the expected significance level given 1,000 bootstrap samples. The y-axis is drawn using a logarithmic scale.

data using this method leads to an inflated Type-I error rate. Our simulations thus imply that bootstrapped difference test for centrality indices for ordinal data should use polychoric correlations as input to the graphical LASSO.

3.6   Conclusion

In this chapter, we have summarized the state-of-the-art in psychometric network modeling, provided a rationale for investigating how susceptible estimated psychological networks are to sampling variation, and described several methods that can be applied after estimating a network structure to check the accuracy and stability of the results. We proposed to perform these checks in three steps: (A) assess the *accuracy* of estimated edge-weights, (B) assess the *stability* of centrality indices after subsetting the data, and (C) test if edge-weights and centralities differ from one-another. Bootstrapping procedures can be used to perform these steps. While bootstrapping edge-weights is straight-forward, we also introduced two new statistical methods: the *correlation stability coefficient* (*CS*-coefficient) and the
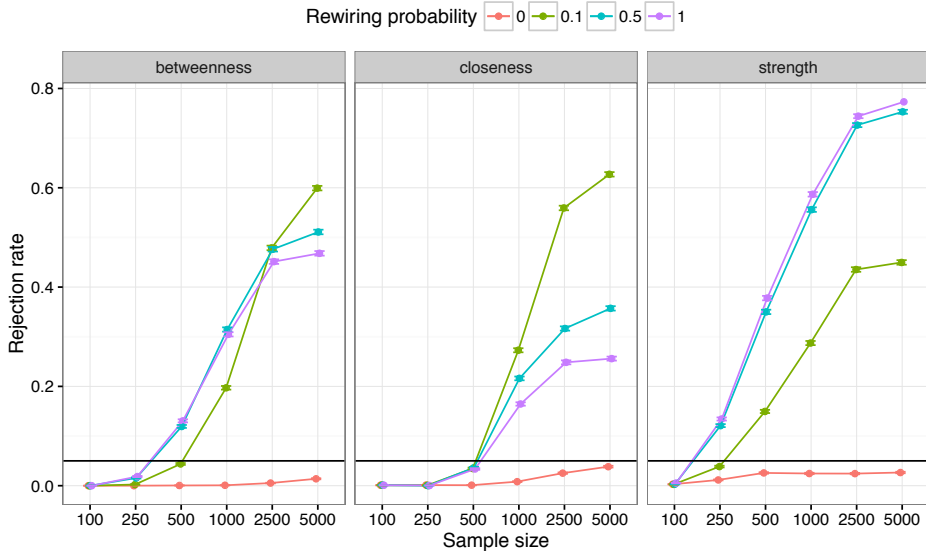
Figure 3.9: Simulation results showing the rejection rate of the bootstrapped difference test for centrality indices. Datasets were generated using the same design as in Figure 3.7. Lines indicate the proportion of times that two random centralities were significantly different (i.e., the null-hypothesis was rejected).

bootstrapped difference test for edge-weights and centrality indices to aid in steps 2 and 3 respectively. To help researchers conduct these analyses, we have developed the freely available R package *bootnet*, which acts as a generalized framework for estimating network models as well as performs the accuracy tests outlined in this chapter. It is of note that, while we demonstrate the functionality of *bootnet* in this tutorial using a Gaussian graphical model, the package can be used for any estimation technique in R that estimates an undirected network (such as the Ising model with binary variables).

Empirical example results. The accuracy analysis of a 17-node symptom network of 359 women with (subthreshold) PTSD showed a network that was susceptible to sampling variation. First, the bootstrapped confidence intervals of the majority of edge-weights were large. Second, we assessed the stability of centrality indices under dropping people from the dataset, which showed that only node strength centrality was moderately stable; betweenness and closeness centrality were not. This means that the order of node strength centrality was somewhat interpretable, although such interpretation should be done with care. Finally, bootstrapped difference tests at a significance level of 0.05 indicated that only in investigating node strength could statistical differences be detected between centralities of nodes, and only three edge-weights were shown to be significantly higher than most other edges in the network.

Limitations and Future Directions

Power-analysis in psychological networks. Overall, we see that networks with increasing sample size are estimated more accurately. This makes it easier to detect differences between centrality estimates, and also increases the stability of the order of centrality estimates. But how many observations are needed to estimate a reasonably stable network? This important question usually referred to as *power-analysis* in other fields of statistics (Cohen, 1977) is largely unanswered for psychological networks. When a reasonable prior guess of the network structure is available, a researcher might opt to use the *parametric* bootstrap, which has also been implemented in bootnet, to investigate the expected accuracy of edge-weights and centrality indices under different sample sizes. However, as the field of psychological networks is still young, such guesses are currently hard to come by. As more network research will be done in psychology, more knowledge will become available on graph structure and edge-weights that can be expected in various fields of psychology. As such, power calculations are a topic for future research and are beyond the scope of the current chapter.

Future directions. While working on this project, two new research questions emerged: is it possible to form an unbiased estimator for centrality indices in partial correlation networks, and consequently, how should true 95% confidence intervals around centrality indices be constructed? As our example highlighted, centrality indices can be highly unstable due to sampling variation, and the estimated sampling distribution of centrality indices can be severely biased. At present, we have no definite answer to these pressing questions that we discuss in some more detail in the online supplementary materials. In addition, constructing bootstrapped CIs on very low significance levels is not feasible with a limited number of bootstrap samples, and approximating p-values on especially networks estimated using regularization is problematic. As a result, performing difference tests while controlling for multiple testing is still a topic of future research. Finally, future research should focus on identifying why the bootstrapped difference test has low statistical power and extend the presented simulation studies in an attempt to identify if the test works under multiple conditions (e.g., different network structures or network models). Given the current emergence of network modeling in psychology, remediating these questions should have high priority.

Related research questions. We only focused on accuracy analysis of cross-sectional network models. Assessing variability on longitudinal and multi-level models is more complicated and beyond the scope of current chapter; it is also not implemented in *bootnet* as of yet. We refer the reader to Bringmann and colleagues (2015) for a demonstration on how confidence intervals can be obtained in a longitudinal multi-level setting. We also want to point out that the results obtained here may be idiosyncratic to the particular data used. In addition, it is important to note that the bootstrapped edge-weights should not be used as a method for comparing networks based on different groups, (e.g., comparing the bootstrapped CI of an edge in one network to the bootstrapped CI of the same edge

in another network) for which a statistical test is being developed.[11] Finally, we wish to point out promising research on obtaining exact p-values and confidence intervals based on the results of LASSO regularized analyses (see Hastie et al., 2015, for an overview), which may in the future lead to a lesser need to rely on bootstrapping methods.

Conclusion

In addition to providing a framework for network estimation as well as performing the accuracy tests proposed in this chapter, *bootnet* offers more functionality to further check the accuracy and stability of results that were beyond the scope of this chapter, such as the parametric bootstrap, node-dropping bootstrap (Costenbader & Valente, 2003) and plots of centrality indices of each node under different levels of subsetting. Future development of *bootnet* will be aimed to implement functionality for a broader range of network models, and we encourage readers to submit any such ideas or feedback to the Github Repository.[12] Network accuracy has been a blind spot in psychological network analysis, and the authors are aware of only one prior paper that has examined network accuracy (Fried, Epskamp, et al., 2016), which used an earlier version of *bootnet* than the version described here. Further remediating the blind spot of network accuracy is of utmost importance if network analysis is to be added as a full-fledged methodology to the toolbox of the psychological researcher.

---

[11]http://www.github.com/cvborkulo/NetworkComparisonTest
[12]http://www.github.com/sachaepskamp/bootnet

Chapter 4

Network Estimation and Sparsity

Abstract

Network models, in which psychopathological disorders are conceptualized as a complex interplay of psychological and biological components, have become increasingly popular in the recent psychopathological literature (Borsboom et al., 2011). These network models often contain significant numbers of unknown parameters, yet the sample sizes available in psychological research are limited. As such, general assumptions about the true network are introduced to reduce the number of free parameters. Incorporating these assumptions, however, means that the resulting network will always reflect the particular structure assumed by the estimation method—a crucial and often ignored aspect of psychopathological networks. For example, observing a sparse structure and simultaneously assuming a sparse structure does not imply that the true model is, in fact, sparse. To illustrate this point, we discuss a recently published paper that reveals a high-dimensional network of psychopathological symptoms (Boschloo et al., 2015). Furthermore, we show the effect of the assumption of sparsity in three simulation studies.

4.1 Introduction

Recent psychological literature has focused on a network approach to model many different psychological phenomena (Schmittmann et al., 2013). Such networks can be high-dimensional structures (i.e., the number of unknown parameters is much larger than the available data), which are hard to estimate without making general assumptions about the underlying true model structure. If the true model is assumed to be sparse, thus containing a small number of connections relative to the number of possible connections, a methodology can be applied that will return

---

This chapter has been adapted from: Epskamp, S., Kruis, J., and Marsman, M. (in press). Estimating psychopathological networks: be careful what you wish for. *PlosOne.*

such a sparse network structure. In other words, assuming a sparse network structure results in estimating a sparse network structure, which means that certain conclusions cannot be drawn from observing such a structure. In this chapter, we argue that care should be taken in interpreting the obtained network structure because the estimation procedure may pollute the results. We will illustrate this by showing examples of networks obtained when sparse networks are estimated even when the true network structure is dense.

4.2 Network Psychometrics

The network approach has been particularly promising in the field of psychopathology. Within this framework, symptoms (e.g., insomnia, fatigue, and concentration problems) are no longer treated as interchangeable indicators of some latent mental disorder (e.g., depression). Instead, symptoms play an active causal role. For example, insomnia leads to fatigue, fatigue leads to concentration problems, and so forth (Borsboom & Cramer, 2013). Psychopathological disorders, then, are not interpreted as the common cause of observed symptoms but rather as emergent behaviors that result from a complex interplay of psychological and biological components. To grasp such a complex structure, a network model can be used in which variables such as symptoms or moods are represented by nodes. Nodes are connected by edges that indicate associations between nodes. This line of research has led to intuitive new insights about various psychopathological concepts such as comorbidity (Borsboom et al., 2011; Cramer et al., 2010), the impact of life events (Cramer, Borsboom, Aggen, & Kendler, 2012; Fried et al., 2015), and sudden life transitions (e.g., sudden onset of a depressive episode; van de Leemput et al., 2014; Wichers, Groot, Psychosystems, ESM Group, & EWS Group, 2016).

The growing popularity of the network perspective on psychological phenomena has culminated in the emergence of a new branch of psychology dedicated to the estimation of network structures on psychological data—network psychometrics (Epskamp, Maris, Waldorp, & Borsboom, in press). This field focuses on tackling the problem of estimating network structures involving large numbers of parameters in high-dimensional models. When cross-sectional data are analyzed, the most popular models that are used are the Gaussian Graphical Model (GGM; Lauritzen, 1996) for continuous data and the Ising model (Ising, 1925) for binary data. Both the GGM and the Ising model fall under a general class of models called *Markov Random Fields*. These models represent variables as nodes which are connected by edges but only if the variables are *conditionally independent*. The strength of an edge (i.e., its absolute deviance from zero) demonstrates the strength of the association between two variables after conditioning on all other variables in the network; this is also termed *concentration* (Cox & Wermuth, 1993). In the GGM, edges directly correspond to partial correlation coefficients. The Ising model does not allow for such standardization, but edge weights can be similarly interpreted. A more detailed conceptual introduction of these models can be found in Chapter 1. A technical introduction to the GGM will follow in Chapter 6, and a technical introduction to the Ising Model will follow in Chapter 8.

In both models, we must estimate a weight matrix that contains $P(P-1)/2$

number of parameters, where P is the number of nodes, in order to encode the network structure. These parameters encompass the conditional relationship between two nodes after conditioning on all other nodes in the network and can be shown to be quite instable with relatively low sample sizes (see Chapter 3). "Relatively low sample sizes," is a loose description and has not yet been well-defined. A general rule would be to have at least as many observations as the number of parameters. But, as will be shown later, this general rule still results in unstable estimates. A common solution to overcome the problem of estimating many parameters is to reduce this number by using some form of regularization or penalization. A particularly promising technique is to apply the 'least absolute shrinkage and selection operator' (LASSO; Tibshirani, 1996) to the edge weights of the network. The LASSO penalizes the sum of absolute parameter values such that the estimated values shrink to zero. That is, the absolute parameter estimates will be small and will often equal exactly zero. Therefore, the resulting model is almost always sparse; only a relatively few number of parameters will be estimated to be nonzero. The use of LASSO typically leads to better performance in cross-validations (i.e., overfitting is prevented) and results in more easily interpretable models compared to nonregularized Ising models. Most important is that if the true network structure is sparse, the LASSO performs well in estimating this network structure and, more specifically, in estimating fewer edges to be nonzero that are actually zero in the true network (i.e., fewer false positives).

The LASSO uses a tuning parameter that controls the sparsity, which can be chosen to minimize some criterion such as the Extended Bayesian Information Criterion (EBIC; Chen & Chen, 2008). This methodology has been shown to work well for both the GGM (Foygel & Drton, 2010) and the Ising model (Foygel Barber & Drton, 2015; van Borkulo et al., 2014), has been implemented in easy-to-use software (e.g., Epskamp et al., 2012; van Borkulo & Epskamp, 2014), and has been utilized in an increasing number of publications (e.g., Dalege et al., 2016; Isvoranu, van Borkulo, et al., 2016; Kossakowski et al., 2016; Langley et al., 2015; Rhemtulla et al., 2016; van Borkulo et al., 2015). For a more thorough introduction to this methodology, we recommend reading van Borkulo et al. (2014) and Chapter 2. To date, the largest application of a psychological network estimated using the LASSO was carried out by Boschloo et al. (2015), who measured 120 psychiatric symptoms in 34,653 subjects and modeled these with an Ising model. We use their work in this chapter to illustrate our concerns regarding the interpretation of network structures that are the result of applying network methodology to data.

4.3 A Sparse Network Model of Psychopathology

The network of Boschloo et al. (2015) shows a network structure in which symptoms representative of a disorder strongly cluster together. Although Boschloo and her colleagues admitted that the found network structure closely represents the structure that is imposed by the Diagnostic and Statistical Manual of Mental Disorders (DSM; American Psychiatric Association, 2013), they concluded that the found structure indicates that symptoms are not interchangeable, which is presumed in the DSM. Commonly, a DSM diagnosis requires an individual to have X

out of Y symptoms, regardless of which specific symptoms. This means that two people with vastly different symptoms can be assigned the same diagnosis. This interchangeability results from an underlying causal notion of unobserved diseases causing symptoms rather than symptoms having an active causal role on each other—a notion more formally known as the common cause model (Schmittmann et al., 2013). Boschloo and her colleagues concluded that the network structure shows that symptoms are not interchangeable, mainly due to apparent differences in the number of connections and the strength of the connections between symptoms (a relative small number of pathways between disorders and the presence of some negative connections).

Although we do not necessarily disagree with the notion that symptoms play an active causal role in psychopathology, we wish to point out that the conclusion that symptoms are not interchangeable is difficult to ascertain from a sparse approximated network structure alone. This is because the LASSO relies on the assumption that the true network structure is sparse; the LASSO will always search for a model in which relatively few edges and paths explain the co-occurrence of all nodes. As a result, the LASSO can have a low sensitivity (i.e., not all true edges are detected) but always has a high specificity (i.e., fewer false positives; van Borkulo et al., 2014). It is this reason why network analysts prefer the LASSO; edges that are estimated by the LASSO are likely to represent true edges. Moreover, the LASSO returns a possible explanation of the data using only a few connections that can be interpreted as causal pathways (Lauritzen, 1996; Pearl, 2000). That the LASSO yields a possible explanation, however, does not mean that the LASSO provides the only explanation, nor does it indicate that other explanations are false. In the case of Boschloo et al., the sparse explanation found by the LASSO can give great insight regarding a possible way in which psychopathological symptoms interact with each other. However, merely finding a sparse structure does not mean that other explanations (e.g., a latent variable model with interchangeable symptoms) are disproved. Simply stated, using the LASSO always returns a sparse structure, that is what the LASSO does.

4.4 The Bet on Sparsity

The LASSO is capable of retrieving the true underlying structure but only if that true structure is sparse. Any regularization method makes the assumption that the true structure can be simplified in some way (e.g., is sparse) because otherwise too many observations are needed to estimate the network structure. This principle has been termed the *bet on sparsity* (Hastie et al., 2001). But what if the truth is not sparse, but dense?

Such a case would precisely arise if the true model were a latent common cause model in which one or several latent variables contribute to scores on completely interchangeable indicators. This is a feasible alternative because the Ising model can be shown to be mathematically equivalent to a certain type of latent variable model: the multidimensional item response model (MIRT; Reckase, 2009), with posterior normal distributions on the latent traits (Marsman, Maris, Bechger, & Glas, 2015; see also Chapter 8). The corresponding Ising model is a low-rank

network that will often be dense (i.e., all possible edges are present). Intuitively, this makes sense because the Ising model parameterizes conditional dependencies between items after conditioning on all other items, and no two items can be made conditionally independent if the common cause model is true. A low-rank weighted network will show indicators of a latent variable as clusters of nodes that are all strongly connected with each other. Therefore, if a common cause model is the true origin of the co-occurrences in the dataset, the corresponding Ising model should show the indicators to cluster together. Then if LASSO regularization is used, the corresponding network would likely feature sparsity but the nodes would still be clustered together—much like the results shown by Boschloo et al.

It is this relationship between the Ising model and MIRT that has led researchers to estimate the Ising model using a different form of regularization, by estimating a low-rank approximation of the network structure (Marsman et al., 2015). Such a structure is strikingly different than the sparse structure returned by LASSO estimation. Whereas the LASSO will always yield many edge parameters to be exactly zero, a low-rank approximation generally estimates no edge to be exactly zero. Thus a low-rank approximation will typically yield a dense network. On the other hand, this dense network is highly constrained by the eigenvector structure, leading many edge parameters to be roughly equivalent to each other rather than compared to the strongly varying edge parameters LASSO estimation allows. For example, the data can always be recoded such that a Rank 1 approximation only has positive connections. These are key points that cannot be ignored when estimating a network structure. Regardless of the true network structure that underlies the data, the LASSO will always return a sparse network structure. Similarly, a low-rank approximation will always return a dense low-rank network structure. Both methods tackle the bet on sparsity in their own way—sparsity in the number of nonzero parameters or sparsity in the number of nonzero eigenvalues—and both can lose the bet.

4.5 Estimating an Ising Model When the Truth Is Dense

Here we illustrate the effect that the estimation procedure has on the resulting Ising model in two examples. First, we simulated 1,000 observations from the true models shown in Figure 4.1. The first model is called a Curie-Weiss model (Kac, 1966), which is fully connected and in which all edges have the same strength (here set to 0.2). This network is a true Rank 1 network, which has been shown to be equivalent to a unidimensional Rasch model (Marsman et al., 2015). The Rasch model is a latent variable model in which all indicators are interchangeable. Figure 4.2 shows the results using three different estimation methods—sequential univariate logistic regressions for unregularized estimation (see Chapter 8), LASSO estimation using the *IsingFit*[1] R package (van Borkulo & Epskamp, 2014), and a Rank 2 approximation (Marsman et al., 2015)—on the first n number of rows in the simulated dataset. It can be seen that the unregularized estimation shows many spurious differences in edge strength, including many negative edges. The LASSO

---

[1] All LASSO analyses in this chapter make use of the default setup of *IsingFit*, using a hyperparameter (γ) value of 0.25 as well as the AND-rule.
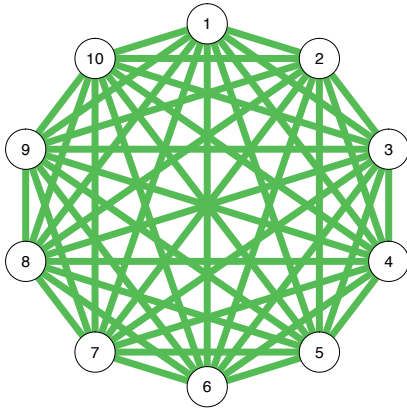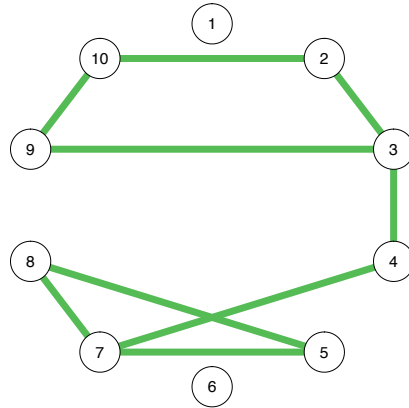
Figure 4.1: True network structures used in simulation study. The first network is a Curie-Weiss network: a fully connected network in which all edges have the same strength. The second network is a random sparse network. All edge weights are 0.2.

performs better but estimates a sparse model in which edge weights vary and in which many edges are estimated to be exactly zero. The Rank 2 approximation works best in capturing the model, which is not surprising because the true model is a Rank 1 network. At high sample sizes, all methods perform well in obtaining the true network structure.

The second model in Figure 4.1 corresponds to a sparse network in which 20% of randomly chosen edge strengths are set to 0.2 and in which the remaining edge strengths are set to 0 (indicating no edge). As Figure 4.3 shows, the LASSO now performs very well in capturing the true underlying structure. Because both the unregularized estimation and the Rank 2 approximation estimate a dense network, they have a very poor specificity (i.e., many false-positive edges). In addition, the Rank 2 approximation retains spurious connections even at high sample sizes (choosing a higher rank will lead to a better estimation). Thus, this example serves to show that the LASSO and low-rank approximations only work well when the assumptions on the true underlying model are met. In particular, using a low-rank approximation when the truth is sparse will result in many false positives, whereas using a LASSO when the truth is dense will result in many false negatives. Even when the true model is one in which every node represents an interchangeable symptom, the LASSO would still return a model in which nodes could be interpreted to not be interchangeable.

For the second example, we simulated data under the latent variable model as shown in Figure 4.4, using an MIRT model (Reckase, 2009). In this model, the symptoms for dysthymia and generalized anxiety disorder (GAD) were taken from the supplementary materials of Boschloo et al. (2015), with the exception of the GAD symptom "sleep disturbance," which we split in two: insomnia and
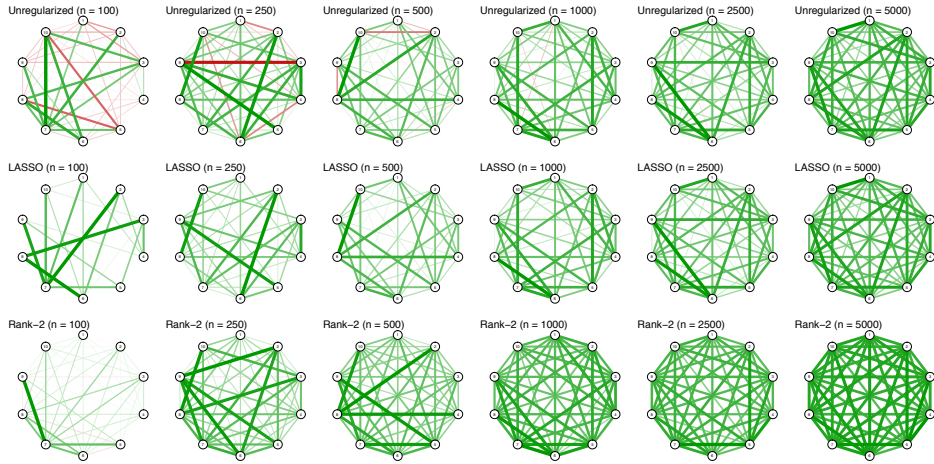
Figure 4.2: Examples of estimated network structures when the true network is a Curie-Weiss network, using different sample sizes and estimation methods. Graphs were drawn using the *qgraph* package without setting a maximum value (i.e., the strongest edge in each network has full saturation and width).

hypersomnia. The item discriminations of each symptom were set to 1, indicating that symptoms are interchangeable, and item difficulties were set to 0. All latent variables were simulated to be normally distributed with a standard deviation of 1, and the correlation between dysthymia and GAD was set to 0.55—similar to the empirically estimated comorbidity (Kessler, Chiu, Demler, & Walters, 2005). Nodes 2 and 3 in dysthymia and nodes 6 and 7 in GAD are mutually exclusive, which we modeled by adding orthogonal factors with slightly higher item discriminations of 1.1 and -1.1. Furthermore, nodes 7, 8, 9, and 10 of dysthymia are identical to nodes 6, 7, 8, and 9 of GAD respectively, which we modeled by adding orthogonal factors with item discriminations of 0.75. These nodes are typically not identical because a skip structure is imposed in datasets such as the one analyzed by Boschloo et al. That is, if someone does not exhibit the symptom "low mood," that person is never asked about insomnia in the depression scale because he or she is assumed to not have this symptom. We did not impose a skip structure to keep the simulation study simple. Such shared symptoms are termed *bridge symptoms* in network analysis because they are assumed to connect the clusters of disorders and explain comorbidity (Borsboom et al., 2011; Cramer et al., 2010). In sum, the model shown in Figure 4.4 generates data that are plausible given the latent disease conceptualization of psychopathology.

Figure 4.5 shows the simulated and recovered network structures. First we simulated 10 million observations from this model and estimated the corresponding Ising model using nonregularized estimation by framing the Ising model as a log-linear model (Agresti, 1990, see Chapter 8) (the estimation was done using the *IsingSampler* package, Epskamp, 2014). Panel A shows the results, which give a good proxy of the true corresponding Ising structure. It can be seen that the true

Figure 4.3: Examples of estimated network structures when the true network is sparse, using different sample sizes and estimation methods. Graphs were drawn using the *qgraph* package without setting a maximum value (i.e., the strongest edge in each network has full saturation and width).

Figure 4.4: A multidimensional IRT model (MIRT) used in simulating data. All latent variables were normally distributed with standard deviation of 1 and all symptoms were binary. The edges in this model correspond to item discrimination parameters.

model is dense, meaning that indicators of the disorders cluster together. Two negative connections are formed between the mutually exclusive indicators, and bridging connections are formed between the shared indicators. Next, we simulated 1,000 observations from the model in Figure 4.4 and estimated the Ising model in various ways. Panel B shows unregularized estimation via a log-linear model and shows many spurious strong connections, including many more neg-

ative connections than present in the true model. As such, Panel B highlights our need to regularize—even in a sizable dataset of 1,000 observations for a 19-node network. The simulated data has 22.2 observations for every parameter, far more than the 4.9 observations per parameter in Boschloo et al. (2015). Thus, even with a high sample size (34,653 subjects) and even when more subjects are measured than there are parameters present, it can still be advisable to use some form of regularization, as is done by Boschloo et al. (Boschloo et al., 2015) in using the LASSO. Panel C shows the result from using the LASSO, using the *IsingFit* package (van Borkulo et al., 2014). In this model, the clustering is generally retrieved—two of the bridging connections are retrieved and one negative connection is retrieved. However, the resulting structure is much more sparse than the true model, and interpreting this structure could lead to the same conclusions determined by Boschloo and her colleagues: The number of connections differed across symptoms, connection strengths varied considerably across symptoms, and relatively few connections connected the two disorders. Finally, Panel D shows the result of a Rank 2 approximation, which is equivalent to a two-factor model. Here, it can be seen that although a dense structure is retrieved that shows the correct clustering, violations of the clustering (the negative and bridging edges) are not retrieved. The online supplementary materials[2] show that with a higher sample size ($n = 5,000$) the estimation is more accurate and that the unregularized and LASSO estimations result in similar network structures.

Different Estimation Techniques

In light of the examples discussed in this chapter, researchers may wonder when they should and should not use a particular estimation method. For example, low-rank estimation is more suited in the example demonstrated in Figure 4.2, whereas LASSO estimation fits better in the example shown in Figure 4.3. These conclusions, however, depend on knowing the true network structure as shown in Figure 4.1—something a researcher will not know in reality. The choice of estimation method, therefore, is not trivial. Choosing the estimation method depends on three criteria: (1) the prior expectation of the true network structure, (2) the relative importance the researcher attributes to sensitivity (discovery) and specificity (caution), and (3) the practical applicability of an estimation procedure. When a researcher expects the true network to be low rank (e.g., due to latent variables), low-rank estimation should be preferred over LASSO regularization. On the other hand, when a researcher expects the network to be sparse, LASSO regularization should be used. In addition, LASSO regularization should be preferred when a researcher aims to have high specificity (i.e., to refrain from estimating an edge that is missing in the true model). Finally, practical arguments can play a role in choosing an estimation procedure as well. LASSO, particularly in combination with EBIC model selection, is relatively fast even with respect to large datasets. As a result, researchers could apply bootstrapping methods to the estimation procedure to further investigate the accuracy of parameter estimation (see Chapter 3), which may not be feasible for slower estimation procedures.

---

[2] http://sachaepskamp.com/files/S2_EstimatedNetworks.pdf
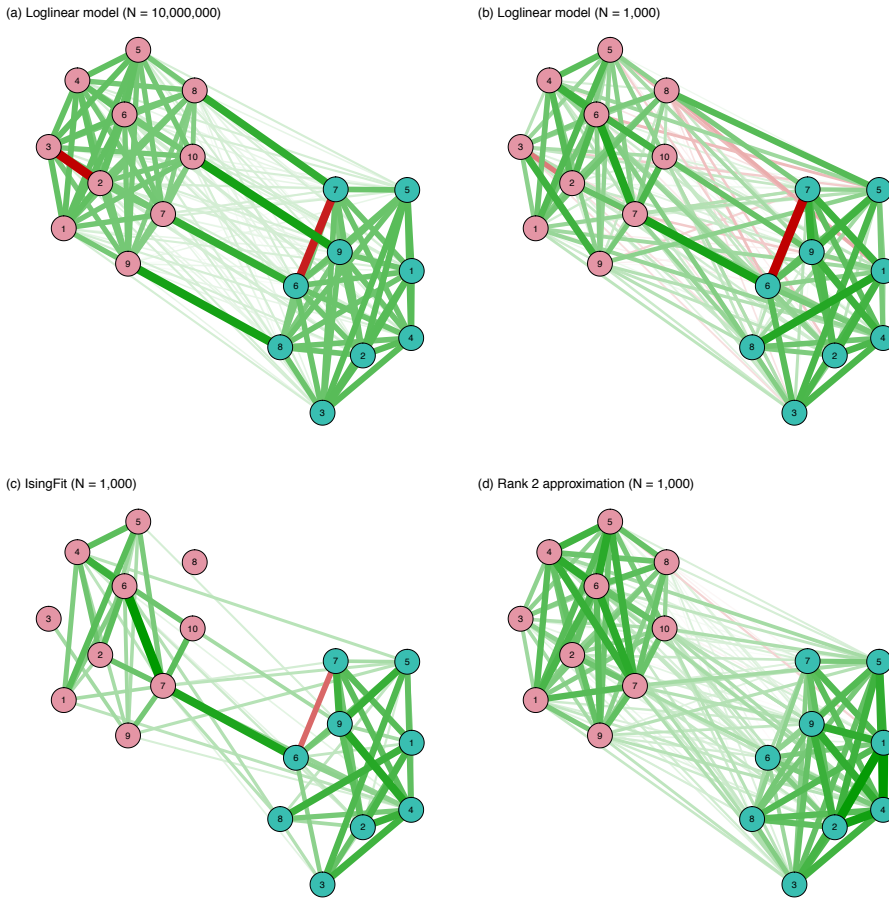
Figure 4.5: Estimated network structures based on data generated by the MIRT model in Figure 4.4.

We focused the argumentation on LASSO regularization and low-rank approximation because these are the main methodologies that have been applied in psychological literature and present two extreme cases of a range of different network structures that can be estimated. Because these methods lie on the extreme ends of sparsity relative to dense networks, they best exemplify the main point of this chapter: In small sample sizes, some assumptions of the true model must be made (e.g., the true model is sparse), and these assumptions influence the resulting network structure (e.g., the obtained network is sparse). This does not mean that LASSO and low-rank approximation are the only methods available. An alternative, for example, is to use *elastic-net* estimation, which mixes LASSO regularization with ridge regression (penalizing the sum of squared coefficients). The *elasticIsing* package (Epskamp, 2016) can be used to accomplish this; it uses

cross-validation in selecting the tuning parameters. The online supplementary materials show an example of elastic-net applied to the data analyzed in Figure 4.5. It is noteworthy that the elastic-net procedure selected a dense network (i.e., ridge regression) over LASSO regularization, indicating that data-driven evidence can be garnered to argue whether or not LASSO regularization should be used. The obtained network, like the unregularized network in Figure 4.5 (Panel B), also shows many connections which were falsely estimated to be negative; this raises the question of whether its result should or should not be preferred over LASSO regularized estimation. The online supplementary materials also contain examples of LASSO regularization using different tuning arguments (e.g., BIC selection instead of EBIC selection), which improves sensitivity (i.e., more edges are detected) in this particular case. Doing so, however, will result in less specificity when the true model is sparse (van Borkulo et al., 2014).

4.6 Conclusion

Network estimation has grown increasingly popular in psychopathological research. The estimation of network structures, such as the Ising model, is a complicated problem due to the fast growing number of parameters to be estimated. As a result, the sample size typically used in psychological science may be insufficient to capture the true underlying model. Although a large sample size network estimation typically goes well regardless of the estimation method used, Figures 4.2, 4.3, and 4.5 show that estimating an Ising model with sample sizes commonly used in psychological research results in poor estimates without the use of some form of constraint on the parameter space. Two such constraints involve limiting the size and number of nonzero parameters (LASSO) or reducing the rank of a network (low-rank approximation). It is important to realize that using such estimation methods makes an assumption on the underlying true model structure: The LASSO assumes a sparse structure whereas low-rank approximation assumes a dense but low-rank structure. Investigating the results of the estimation methods cannot validate these assumptions. The LASSO always yields a sparse structure, which does not mean that the true underlying structure could not have been dense. On the other hand, low-rank approximations rarely produce sparse structures, but that does not mean that the true underlying structure could not have been sparse.

Figure 4.2 illustrates this point by showing that LASSO estimation when the true network structure is a Curie-Weiss model still results in a sparse structure. This means that observing any of the sparse structures shown in Figure 4.2 does not mean that the nodes in the network could not represent interchangeable indicators of a single latent trait. Figure 4.5 illustrates this point again in a plausible scenario in psychopathology and also shows that when the true network structure is complicated and neither sparse nor low rank, as is the case here, all regularization methods partly fail even when using a relatively large sample size. As such, interpreting the sparsity of such a structure is questionable; the LASSO resulting in a sparse model gives us little evidence for the true model being sparse because a low-rank approximation returning a dense model seems to indicate that the true model is dense. Those characteristics from the networks we obtain are a

consequence of the method used to estimate a network structure (specifically the assumptions made by the employed method about the data-generating network structure) and often pollute the resulting estimated model (Kruis & Maris, 2015).

Recently it has been demonstrated that three, statistically indistinguishable, representations of the Ising model exist that explain observed associations between binary variables either through a common cause (latent variable), through the reciprocal effect between variables (network), or through the conditioning on a common effect (collider variable; Epskamp et al., in press; Marsman et al., 2015; Kruis & Maris, 2016). Consequently, when a model from one of these frameworks can sufficiently describe the associative structure of the measured variables, there exists an alternative representation for other frameworks that can also accurately represent the structure of the data. For example, Boschloo et al.'s spare network structure (Boschloo et al., 2015), resulting from the LASSO being applied to the data, can also be described by a multidimensional latent variable model (with a single latent variable for each clique in the network) and residual correlations. As such, obtaining sufficient fit for a statistical network model cannot be regarded as evidence for the theoretical model, where a network structure acts as the causal mechanism from which associations between variables emerge. We therefore advise, in general, to tread carefully when drawing inferences about the theoretical causal mechanisms that generate the data from statistical model fit.

Network models show great promise in mapping out and visualizing relationships present in the data and are useful to comprehend high dimensional multivariate relationships. In addition, network models can be powerful tools to estimate the backbones of potential causal relationships—if those relationships are assumed to exist. Using the LASSO to estimate such network structures is a powerful tool in performing fast high-dimensional model selection that results in fewer false positives, and interpreting network structures obtained from the LASSO can illuminate the strong relationships present in the dataset. Important to realize is that using LASSO estimation will result in a sparse structure, and similarly, using a low-rank approximation will result in a dense low-rank result. Our aim here is not to argue against using the LASSO or to argue that estimating network structures is wrong. Our aim is to clarify that choosing the estimation method is not trivial and can greatly impact both the estimated structure as well as any conclusions drawn from that structure.

Chapter 5

Personalized Network Modeling in Psychopathology

Abstract

Two methodological trends have become prominent in the fields of psychopathology and psychiatry: (1) technological developments in collecting time-intensive, repeated, intra-individual measurements in order to capture symptom fluctuations and other time varying factors (e.g., emotions/affect) in daily life (i.e., time-series), and (2) an increasing number of statistical tools for for estimating assocations between these measurements (i.e., network structures) based on these time-series data. Combining these two trends allows for the estimation of intra-individual network structures. Using vector-autoregression (VAR), two networks can be obtained: a *temporal network*, in which one investigates if symptoms (or other relevant variables) predict one another over time, and a *contemporaneous network*, in which one investigates if symptoms predict one another in the same window of measurement. The network literature using these models has so far mostly focused on the temporal network. Here we argue that temporal relations between psychopathological variables might typically unfold within shorter time intervals (e.g., minutes) than the time intervals commonly and feasibly used in current time-series studies (e.g., hours). As a result, such temporal relations will be captured in the contemporaneous network, rather than in the temporal network. Both temporal and contemporaneous networks may highlight *potential* causal pathways—they are not definitive proof of causality but may lead to meaningful insights. As such, both types of networks function as *hypothesis generators*. We conclude the chapter with empirical examples of such analyses on symptom time-series data from clinical cases.

5.1 Introduction

Recent years have witnessed an emergence of two distinct trends in the study of psychopathology. First, technological advances have permitted the gathering of intensive repeated measurements of patients and healthy controls with the Experience Sampling Method (ESM; Aan het Rot, Hogenelst, & Schoevers, 2012; Myin-Germeys et al., 2009; Wichers, Lothmann, Simons, Nicolson, & Peeters, 2012). With ESM, participants are measured repeatedly within short time intervals during daily life. For example, someone is queried five times a day during a period of two weeks on his or her level of insomnia, depressed mood, and fatigue since the previous measurement. We will term the time frame on which one reports the *window of measurement.* The resulting time-series data allow for the investigation of intra-individual processes (Hamaker, 2012). The second trend is the network perspective on psychopathology, in which mental disorders are interpreted as the consequence of a dynamical interplay between symptoms and other variables (Borsboom & Cramer, 2013; Cramer et al., 2010; Cramer & Borsboom, 2015). This literature uses network models in an attempt to understand and predict the dynamics of psychopathology. From this perspective, symptoms are not seen as passive indicators of a mental disorder but rather play an active role, making symptoms prime candidates for interventions (Borsboom, in press; Fried, Epskamp, et al., 2016).

   Time-series data of a single individual offer a promising gateway into understanding the dynamical processes that may occur within that individual over time (e.g., Bringmann et al., 2013, 2015; Pe et al., 2015; Wigman et al., 2015). Such personalized network structures are typically estimated using a statistical technique called *vector-autoregression* (VAR; van der Krieke et al., 2015). Predominantly, VAR analyses have focused on the estimation of *temporal relationships* (relationships that occur between different windows of measurement). However, as will be outlined in Chapter 6, the residuals of the VAR model can be further used to estimate *contemporaneous relationships* (relationships that occur in the same window of measurement), which are not yet commonly used in the field. In this chapter, we argue that both network structures generate valuable hypothesis-generating information directly applicable to the study of psychopathology as well as to clinical practice. We focus the majority of the discussion on explaining contemporaneous partial correlation networks, as these are not yet often utilized in the literature of intra-individual analysis. We exemplify this by analyzing two ESM datasets obtained from patients.

5.2 Temporal and Contemporaneous Networks

In time-series data analysis with an average time-interval of a few hours, a typical default statistical assumption is violated: consecutive responses are not likely to be independent (e.g., someone who is tired between 9:00 and 11:00 is likely to still be tired between 11:00 and 13:00). The minimal method of coping with this violation of independence is the lag-1 VAR model (van der Krieke et al., 2015). In this model, a variable in a certain window of measurement is predicted by the same

variable in the previous window of measurement (autoregressive effects) and all other variables in the previous window of measurement (cross-lagged effects; Selig & Little, 2012)[1]. This model does not assume auto-correlations between larger differences in time (e.g., lag-2) are zero, but merely that such relationships can be fully explained by the lag-1 model. These autoregressive and cross-lagged effects can be estimated and visualized in a *network* (Bringmann et al., 2013). In this network, measured variables (such as symptoms) are represented by *nodes*. When one variable *predicts* another in the next window of measurement, we draw a *link* with an arrowhead pointing from one node to the other. We term this network the *temporal network*.

The predictive effects shown in the temporal network satisfy the assumption that in a causal relationship the cause must precede the effect. Therefore, these are often interpreted to be indicative of causal relationships. Only interpreting temporal coefficients, however, does not utilize VAR to its full potential. The residuals of the temporal VAR model are correlated; correlations in the *same* window of measurement remain that cannot be explained by the temporal effects. These correlations can be used to compute a network of *partial correlations* (Wild et al., 2010). In such a network, each variable is again represented by a node. Links (without arrowhead) between two nodes indicate the partial correlation obtained after controlling for both temporal effects and all other variables in the same window of measurement. We term this network the *contemporaneous network*[2].

Figure 5.1 shows an example of the two network structures obtained from a VAR analysis. These networks are estimated using ESM data of a clinical patient, and are further described and interpreted in Section 5.6. The temporal and contemporaneous networks were estimated at the same time, using the methodology outlined by Abegaz and Wit (2013). The temporal network (a) shows autoregressions (an arrow of a node pointing at itself) on three variables: 'tired', 'bodily discomfort' and 'concentration'. Thus, when this patient is tired she is likely still tired during the next window of measurement. There are cross-lagged relationships between several variables. For example, this patient being tired predicts her to ruminate more during the next window of measurement.. The contemporaneous network (b) shows, among other relationships, a relationship between 'relaxed' and 'nervous': when this patient was tired she was also more likely to relax poorer, *as reported during the same window of measurement*. This can be seen as the direct consequence of a plausible causal relationship: being nervous might lead you to feel less relaxed (or vice-versa). There is no reason why such a causal relationship should take a few hours to occur, which brings us to the main point of this chapter.

---

[1]VAR can be seen as an ordinary regression where the predictors are lagged variables.

[2]The contemporaneous network should not be confused with a network of lag-0 (partial) correlations. Such a network would (1) not take into account that responses are not independent, and (2) present a mixture of temporal and contemporaneous effects. Thus, we obtain the contemporaneous network from the residuals of the VAR model, since only then relationships *between* windows of measurement and relationships *within* windows of measurement are separated.
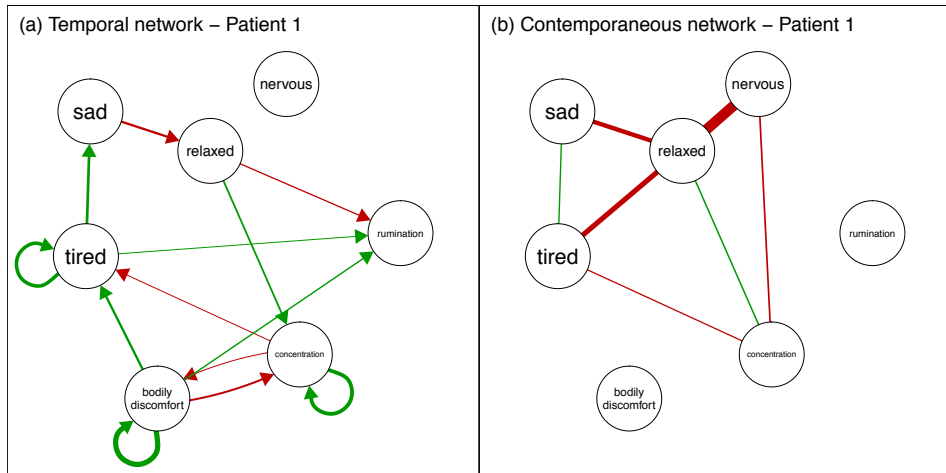
Figure 5.1: Two network structures that can be estimated with time-series data analysis, based on data of a clinical patient ($n = 52$) measured over a period of two weeks. The model was estimated using the *graphicalVAR* package for R. Circles (nodes) represent variables, such as symptoms, and connections (links, both undirected drawn as simple lines or directed drawn as an arrow) indicate predictive relationships. Green links indicate positive relationships, red links indicate negative relationships, and the width and saturation of a link indicates the strength of the relationship. The network on the left shows a *temporal network*, in which a link denotes that one variable predicts another variable in the next window of measurement. The network on the right shows a *contemporaneous network*, in which links indicate partial correlations between variables in the same window of measurement, after controlling for all other variables in the same window of measurement and all variables of the previous window of measurement.

5.3 Causation at the Contemporaneous Level

In a typical ESM study, the time between consecutive measurements is a few hours[3]. As such, the temporal network will only contain predictive effects of measured variables on other measured variables about a few hours later. However, it is likely that many causal relationships occur *much faster* than a timeframe of a few hours. Take for example a classical causal model:

$$\text{Turn on sprinklers} \rightarrow \text{Grass is wet.}$$

Turning on the sprinklers causes the grass to become wet. This causal effect occurs very fast: after turning on the sprinklers it takes perhaps a few seconds for the grass to become wet. If we take measures of sprinklers ("on" or "off") and the wetness of the grass every two hours, it would be rather improbable to capture the

---

[3]Notable exceptions are sampling designs in which individuals are asked to fill-out questionnaires once a day or week.

case in which the sprinklers were turned on *just* before the grass became wet. As such, the temporal network would *not* contain a connection between turning on the sprinklers and the grass being wet, and likely *would* only contain a temporal auto-regression of the grass being wet (because it takes time for grass to dry). However, after controlling for this auto-regression, we most likely would find a connection between these variables in the contemporaneous network: in windows of measurement where the sprinklers were on we would likely also find that the grass was wet.

We can think in a similar vein about psychopathological causal relationships. For example, a patient suffering from panic disorder might anticipate a panic attack by experiencing somatic arousal (e.g., sweating, increased heart rate):

$$\text{Somatic arousal} \rightarrow \text{anticipation of panic attack}$$

In this structure, an arrow indicates that whatever is on the left causes whatever is on the right. This patient anticipates a panic attack, because the patient is experiencing somatic arousal. This causal effect would likely occur within minutes, not hours. Someone who experiences somatic arousal between 13:00 and 15:00 might still experience somatic arousal between 15:00 and 17:00. Thus, we can expect to find auto-regressions. However, between somatic arousal and anticipation of panic attack we would likely only find a contemporaneous connection.

In sum, relations between symptoms and other variables can plausibly unfold faster than the time-window of measurement; such relationships will be captured in the contemporaneous network. Figure 5.1 showed, however, that the contemporaneous network has no direction (links have no arrow-heads). To understand *how* such undirected networks can still highlight potential causal pathways, we need to delve into the literature on estimation of networks in psychopathology.

5.4 Partial Correlation Networks

As outlined above, the contemporaneous relationships can be interpreted and drawn as a network of partial correlations. In this section, we describe how such partial correlation networks can be interpreted and how links in such a network can be seen as indicative of causal relationships. Partial correlation networks are part of a more general class of undirected (i.e., no arrows) networks (formally called Markov Random Fields; Lauritzen, 1996) that have been introduced to psychopathology in response to the call for conceptualizing psychopathological behavior (such as symptoms) as complex networks (Borsboom et al., 2011; Cramer et al., 2010). After the initial introduction of partial correlation networks to the psychopathological literature (Borsboom & Cramer, 2013), the use of undirected networks in psychopathology gained considerable traction following the introduction of easy-to-use estimation methods and publicly available software packages for both estimation and visualization (Epskamp et al., 2012; van Borkulo et al., 2014). Ever since, such network structures have been extensively applied to research in the fields of psychopathology and psychiatry, such as comorbidity (Boschloo et al., 2015), autism (Ruzzano, Borsboom, & Geurts, 2015), post-traumatic stress disorder (McNally et al., 2015), psychotic disorders (Isvoranu, van Borkulo, et al.,

2016; Isvoranu, Borsboom, et al., 2016), major depression (Fried et al., 2015; van Borkulo et al., 2015), and clinical care (Kroeze et al., 2016).

Partial correlation networks have become so prominent because they present a relatively easy method to estimate and visualize potential causal pathways, while taking into account that observational data (i.e., no experimental interventions) only contains limited information on such causal relationships. In observational data, causality is reflected only in the conditional independence structure (Pearl, 2000). Conditional independence means that two variables are no longer correlated at fixed levels of a third variable. A partial correlation network shows conditional independence, because when the partial correlation between two variables after conditioning on all others equals zero, then that means two variables are conditionally independent. Therefore, two nodes that are not directly connected via a link are conditionally independent.

Taking again the patient described above suffering from a panic disorder, suppose we expand the causal structure to include this patient's pathway related to avoiding feared situations:

Somatic arousal → anticipation of panic attack → avoidance of feared situations.

Anticipating a panic attack might cause this patient to avoid feared situations, such as malls or busy shopping streets[4]. The causal structure indicates that we would expect to be able to predict this patent avoiding feard situations given that he or she is experiencing somatic arousal. However, if we already know this person is anticipating a panic attack, we already predict that this person will avoid feared sitatations. Then, observing somatic arousal on top of anticipating the panic attack does *not* improve this prediction. Thus, we would expect non-zero partial correlations between somatic arousal and anticipation of panic attack, and between anticipation of panic attack and avoidance. We would furthermore expect a partial correlation of *zero* between somatic arousal and avoidance behavior; somatic arousal and avoidance behavior are conditionally independent given the anticipation of a panic attack. Consequently, we would expect the following partial correlation network:

Somatic arousal — anticipation of panic attack — avoidance behavior.

Finding such a partial correlation network often does not allow one to find the true direction of causation. This is due to two technical arguments: (1) equivalent models explain the same conditional independencies and (2) these models only work when we can assume the causal structure is acyclic (i.e., contains no feedback loops). Concerning the first argument, we can summarize the above causal structure as $A \rightarrow B \rightarrow C$, in which A and C are conditionally independent given B. This conditional independence, however, also holds for two other models: $A \leftarrow B \leftarrow C$ and $A \leftarrow B \rightarrow C$ (Pearl, 2000). In general, we cannot distinguish between these three models using only observational data. Adding more variables only increases this problem of potentially equivalent models, making it difficult to construct such a network only from observational data. Even

---

[4]These relationships should be taken as an example. The direction of such effects is still at topic of debate, and likely differs from patient to patient (Frijda, 1988).

when such a network can be constructed, we need to assume that the structure is not self-enforcing. That is, a variable cannot cause itself via some chain (e.g., $A \rightarrow B \rightarrow C \rightarrow A$). In psychopathology, however, this assumption likely does not hold (in our example above: anticipating a panic attack might cause more somatic arousal). As a result of these problems with directed structures when temporal or experimental information is lacking, undirected networks have been more successful in this emergent research field. In an undirected network, the observation that A and C are conditionally independent given B is represented by only one model: $A - B - C$ (Lauritzen, 1996).

To summarize, the contemporaneous network is an undirected network without arrowheads. This network shows a link when two variables are not conditionally independent given both all responses in the previous window of measurement and responses of all other variables in the current window of measurement. When two variables are conditionally independent, no link is drawn. If a causal relationship were present, we would expect such a link, and if a causal relationship were not present, we would not expect such a link. Therefore, the links in the contemporaneous network can be indicative of causal relationships. However, as finding the direction of such relationships is hard, we do not attempt to do so and keep the links in the contemporaneous network without direction.

5.5   Generating Causal Hypotheses

The connections in both the temporal and contemporaneous network cannot be interpreted as true causal relationships except under strong assumptions. The pathways shown can only be *indicative* of potential causal relationships. Such a pathway is a necessary condition for causality (we would expect such relationships when there is a true causal effect), but not sufficient (the relationship can also be spurious and due to, e.g., unobserved causes; Pearl, 2000). Therefore, these networks can be seen as *hypothesis generating*. To test for causality one needs to investigate what happens after experimentally changing one variable. If fatigue causes concentration problems, we would expect concentration levels to change after experimentally making someone fatigued. Experimentally changing concentration levels should, on the other hand, not influence fatigue. Such causal testing can only be done experimentally; it is hard to infer causality from observational data, no matter how often and intensive someone is measured and how intensive the sampling rate is.

In addition to generating hypotheses on causal links, both networks also generate hypotheses on which *nodes* are important. The importance of nodes in a network can be quantified with descriptive measures called *centrality measures* (Costantini, Epskamp, et al., 2015; Newman, 2010; Opsahl et al., 2010). A node with a high centrality is said to be 'central', indicating the node is well connected in the network. Such a central node may be a prime candidate for intervention, as targeting this node will influence the rest of the system. This is not only the case for central nodes in the temporal network, but also for central nodes in the contemporaneous network. Even when a node has no temporal connections, it can still carry a lot of information on subsequent measurements, purely by being

central in the contemporaneous network. For example, if A predicts B and C in the same window of measurement (contemporaneous links), and B and C both predict themselves in the next window of measurement (autoregressions), then as a result A is able to predict B and C in the next window of measurement, even though no cross-lagged relationships might be found in the temporal network.

While experimental intervention is needed to test causal hypotheses, such hypotheses on causal relationships and central nodes might be hard to verify in practice. For example, one cannot wait with forming treatment plans until after lengthy experimental designs have been tested on a clinical patient. In addition, in intensive treatments for example, multiple nodes are likely to be targeted simultaneously; the causal effect of one particular node is hard to test. Furhermore, it might not be known how certain symptoms can be treated at all (e.g., feelings of derealisation when the patient is suffering from a comorbid depersonalisation disorder, a disorder that is often concurrent with a panic disorder). Still, the obtained insights are useful: the personalized networks can be discussed with the patient and, when the patient recognizes the discovered relationships, help to generate hypotheses and choose interventions that target these nodes (Kroeze et al., 2016).

5.6   Clinical Example

To exemplify how the described symptom networks can be utilized in clinical practice, we analyzed ESM data obtained from two patients treated in a tertiary outpatient clinic in Groningen. Patient 1 was a female patient, aged 23, who received cognitive behavioral therapy (CBT) for a severe panic disorder and a depressive disorder secondary to the panic disorder. Her response rate was 74%. Patient 2 was a female patient, aged 53 suffering from major depressive disorder, in early partial remission after having received electroconvulsive therapy (ECT). Her response rate was 93%, and data collection started one day after her last ECT session.

Methods

The patients received an extensive briefing plus written user instructions for the ESM measurements. Direct support was available 24/7. Patient data were gathered during normal daily life with an ESM tool developed for an ongoing epidemiological study. With our secured server system (RoQua; roqua.nl; Sytema & Van der Krieke, 2013), text messages with links to online questionnaires were sent to the patient's smartphone. All items could be answered on a 7-point Likert scale varying from '1=not at all' to '7=very much'. Measurement occasions were scheduled five times a day every three hours for two weeks (maximal number of possible measurement is 70), and took three to five minutes to complete. The timing of the measurements was adjusted to their individual daily rhythm with the last measurement timed 30 minutes before going to bed. Patients were instructed to fill-out the questionnaires as soon as possible after receiving the text message. The patient received a reminder after 30 minutes, and after 60 minutes the link
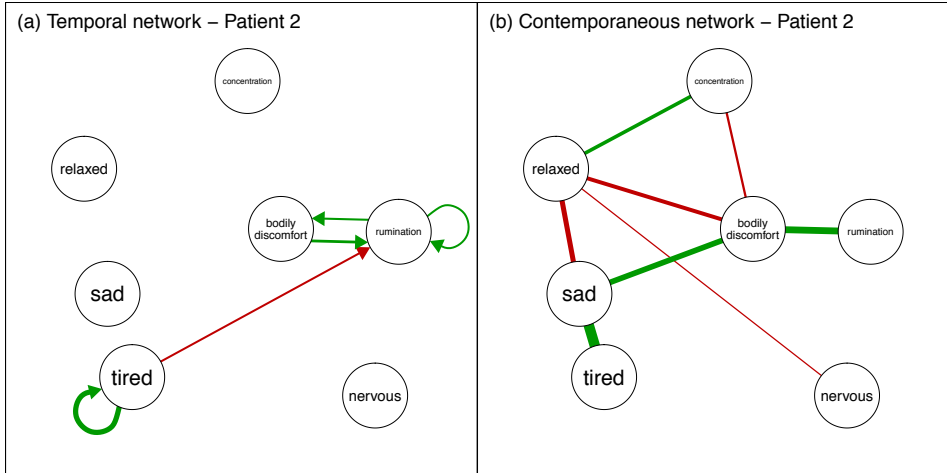
Figure 5.2: Temporal (panel a) and contemporaneous (panel b) network based on data of a clinical patient (n = 65) measured over a period of two weeks. The model was estimated using the graphicalVAR package for R.

was closed. The protocol used was submitted to the ethical review board of the University Medical Center Groningen, who confirmed that formal assessment was not required. Prior to participation, patients were fully informed about the study after which they gave written informed consent.

To estimate simple network structures with not many nodes, we selected seven of the administered variables that usually should interact with each other: feeling sad, being tired, ruminating, experiencing bodily discomfort, feeling nervous, feeling relaxed and being able to concentrate. Network structures were standardized as described by Wild et al. (2010) to avoid misleading parameter estimates in the network structure (Bulteel, Tuerlinckx, Brose, & Ceulemans, 2016). The networks were estimated using the *graphicalVAR* package for R (Epskamp, 2015), which uses penalized maximum likelihood estimation to estimate model parameters (strength of connections) while simultaneously controlling for parsimony (which links are removed; Abegaz & Wit, 2013; Rothman et al., 2010). The graphicalVAR package estimates 2,500 different models, varying 50 levels of parsimony in the temporal network and 50 levels of parsimony in the contemporaneous network. Bayesian Information Criterion (BIC) model selection was used to select the best fitting model. A more detailed description of the estimation procedure is beyond the scope of this chapter. An introduction to model selection of regularized networks is provided in Chapter 2, and a methodological introduction to this modeling framework is provided in Chapter 6. We refer the reader to (Abegaz & Wit, 2013) for the estimation procedure used.

Results

Figure 5.1 shows the two network structures of Patient 1. The temporal network in Panel (a) shows several connections involving bodily discomfort: whenever she experienced bodily discomfort, she ruminated more, felt more tired and was less able to concentrate three hours later. The contemporaneous network in Panel (b) shows that feeling relaxed plays a central role in the network. Whenever she was relaxed she experienced less sadness, tiredness, nervousness and was better able to concentrate (and vise versa, e.g., whenever she experienced less sadness she was more relaxed). In the case of Patient 1, therapy sessions revealed that intensively cleaning her house was her way of coping with stress. This lead to bodily discomfort and eventually rumination about her inability to do the things in the way she used to do things, resulting in a sad mood. Teaching her other ways to cope with stress broke this negative pattern.

Figure 5.2 shows the two network structures of Patient 2. The contemporaneous network in Panel (b) features more connections than the temporal network in Panel (a). In the contemporaneous network, the node bodily discomfort has a central role. Whenever Patient 2 experienced bodily discomfort (in her case, palpitations), she felt sadder, less relaxed, ruminated more and was less able to concentrate within the same window of measurement. This fits the pathology of a panic disorder where bodily sensations are interpreted catastrophically. The temporal network shows the effects over time and highlights a potential feedback loop, where bodily discomfort rumination (in her case, catastrophic interpretations of the bodily sensations) leads to more attention to bodily discomfort, causing more rumination. Feeling tired seems also to lead to more rumination in time.

5.7   Conclusion

In this chapter we argued that when analyzing intra-individual time-series data in clinical settings, researchers should focus on both temporal and contemporaneous relationships. While temporal networks are commonly estimated and interpreted in the network approach to psychopathology (e.g., Bringmann et al., 2013), contemporaneous networks, especially when drawn as a partial correlation network, are not commonly used. We have argued that both contemporaneous and temporal networks can highlight meaningful relationships, interpretable and useable by patients and clinicians in treatment, as well as present researchers with hypothesis generating exploratory results on potential causal relationships. Such personalized knowledge can be used for intervention selection (e.g., choosing which symptoms to treat), as well as generate testable hypotheses pertaining to the individual patient that can be used to perform experiments. In addition to temporal relationship, contemporaneous relationships are also important in discovering psychological dynamics, as such relationships can also occur at a much faster time scale than the typical lag interval used in ESM studies.

A main limitation of the VAR method is that, even when contemporaneous networks are estimated, the results depend on the lag-interval used. If the lag interval is too long, meaningful relationships might not be retrieved (e.g., some dynamics might occur between days or weeks rather than hours). Conversely, if

the relationship is too fast, and dissipates fast, it might also not be retrieved (e.g., if the effect of a relation dissipates after minutes, it might not be captured in a design that measures persons hourly or slower). The optimal lag-interval is often unknown, and can even differ between individuals and different variables. The lag-interval used is typically chosen in part due to practical reasons; it is not feasible for a patient to fill out a questionnaire often during a day (e.g., 20 times a day). The data gathering can also not take too long (e.g., more than two weeks), as the VAR model typically assumes people do not structurally change (Haslbeck & Waldorp, 2016a; Wichers et al., 2016). While effects that are slower than the lag-interval could be captured in a second temporal network (e.g., a network between days in addition to a network between measurements; de Haan-Rietdijk, Kuppens, & Hamaker, 2016), such methods require more observations.

The aim of this chapter is not to argue against interpreting temporal coefficients; both temporal and contemporaneous effects contain meaningful information on how the observed variables relate to one-another. Regardless, we strongly argue that the temporal and contemporaneous relationships should not be over-interpreted, as these merely highlight potential causal pathways. So what is the use then of contemporaneous and temporal networks if they do not allow for causal interpretation? We argue that, for an individual patient, it is hardly relevant if relationships in his/her data are causal or not. What matters is that both types of networks give the clinician as well as the patient a personalized, and visualized window into a patient's daily life. Moreover, this personalized window comes with a host of opportunities to arrive at tailor-made intervention strategies (e.g., treating central symptom of patient), and to monitor progress (e.g., will "deactivating" central symptom result in the deactivation of other symptoms?). Additionally, discussing the idiographic intricacies of networks with the patient offers ample opportunity for the patient to gain insight into his/her strengths and pitfalls and for reinforcing a sense of participation in one's own care. Personalized care is on everybody's agenda, and rightly so; given its benefits, so should network modeling of psychopathology data at the level of the individual be.

Part II

Technical Advances in Network Psychometrics

Chapter 6

Discovering Psychological Dynamics

Abstract

This chapter outlines statistical network models in cross-sectional and time-series data, that attempt to highlight potential causal relationships between observed variables. The chapter describes three kinds of datasets. In cross-sectional data (1), one can estimate a Gaussian graphical model (GGM; a network of partial correlation coefficients). In single-subject time-series analysis (2), networks are typically constructed through the use of (multilevel) vector autoregression (VAR). VAR estimates a directed network that encodes temporal predictive effects—the temporal network. We show that GGM and VAR models are closely related: VAR generalizes the GGM by taking violations of independence between consecutive cases into account. VAR analyses can also return a GGM that encodes relationships within the same window of measurement—the contemporaneous network. When multiple subjects are measured (3), multilevel VAR estimates fixed and random temporal networks. We show that between-subject effects can also be obtained in a GGM network—the between-subjects network. We propose a novel two-step multilevel estimation procedure to obtain fixed and random effects for contemporaneous network structures. This procedure is implemented in the R package mlVAR. The chapter presents a simulation study to show the performance of *mlVAR* and showcases the method in an empirical example on personality inventory items and physical exercise.

6.1 Introduction

Network modeling of psychological data has increased in recent years. This is consistent with a general call to conceptualize observed psychological processes that are not merely indicative of latent common causes but rather reflect the emergent

---

behavior of complex, dynamical systems in which psychological, biological, and sociological components directly interact with each other (Borsboom et al., 2011; Cramer, Sluis, et al., 2012; Cramer et al., 2010; Schmittmann et al., 2013; Van Der Maas et al., 2006). Such relationships are typically not known, and probabilistic network models (Koller & Friedman, 2009) are used to explore potential causal relationships between observables (Epskamp et al., in press; van Borkulo et al., 2014)—the dynamics of psychology. This chapter provides a methodological overview of statistical network models in cross-sectional and time-series data. Furthermore, this chapter shows that the common network models for cross-sectional and time-series data are closely related. In time-series modeling, this relationship allows researchers to extend the modeling framework to incorporate contemporaneous and between-subject effects. We propose a novel estimation procedure to do so, which we implemented in the free software package *mlVAR*.[1]

We can distinguish two lines of research in which networks are utilized on psychological datasets: modeling of cross-sectional data and modeling of intensive repeated measures in relatively short time frames (e.g., several times per day during several weeks). In cross-sectional modeling, a model is applied to a dataset in which multiple persons are measured only once. The most popular method is to estimate undirected network models, indicating pairwise interactions—so-called pairwise Markov random fields (Lauritzen, 1996; Murphy, 2012). When the data are continuous, the Gaussian graphical model (GGM; Lauritzen, 1996) can be estimated. The GGM estimates a network of *partial correlation coefficients*—the correlation between two variables after conditioning on all other variables in the dataset. This model is applied extensively to psychological data (e.g., Cramer, Sluis, et al., 2012; Fried, Epskamp, et al., 2016; Isvoranu, van Borkulo, et al., 2016; Kossakowski et al., 2016; McNally et al., 2015; van Borkulo et al., 2015).

Time-series data can be obtained by using the experience sampling method (ESM; Myin-Germeys et al., 2009), in which subjects are asked several times per day to fill in a short questionnaire through a device or smartphone app. Often in ESM data, repeated measures of one or multiple participants are modeled through the use of (multilevel) vector autoregressive (VAR) models, which estimate how well each variable predicts the measured variables at the next time point (Borsboom & Cramer, 2013). These models are growing increasingly popular in assessing intraindividual dynamical structures (e.g., Bringmann et al., 2013, 2015; Wigman et al., 2015). As will be shown below, the VAR model can be seen as a generalization of the GGM that takes violations of independence between consecutive cases into account. Thus, the lines of research on cross-sectional and time-series data can naturally be combined. The GGM is, however, not yet commonly used in time-series analysis.

In this chapter we present an overview of out-of-the-box methods, applicable to normally distributed data, that aim to map out the dynamics present in psychological data. We will do so in two distinct settings: cross-sectional data, in which observations are plausibly independent, and intensive repeated measures in a relatively short time span obtained through ESM data. In multivariate normal data,

---

[1]CRAN link: `http://cran.r-project.org/package=mlVAR`
Github link (developmental): `http://www.github.com/SachaEpskamp/mlVAR`

all relationships between variables are contained within the variance–covariance matrix. Thus, characterizing the covariances in estimable ways provides a method to characterize all relationships that are present. This information can then be represented in networks. In cross-sectional data, one network can be obtained—an undirected network of partial correlation coefficients. In time-series data, up to three networks can be obtained—the *temporal network*, a directed network indicating within-person relationships across time; the *contemporaneous network*, an undirected partial correlation network within the same measurement; and the *between-subjects network*, an undirected partial correlation network between the means of the subject's scores within the time span of measurement. We describe how all three networks can highlight potential causal pathways and thereby act as hypothesis-generating structures.

The chapter is structured in the following manner. We first characterize the joint likelihood of full ESM data in three steps: (a) when cases are deemed to be plausibly independent, (b) when the time-series data of a single subject contain plausibly nonindependent observations, and (c) when we have time-series data of several subjects that combine both independent and nonindependent observations. In each of these situations, we outline estimation procedures including a description of open-source software packages. We also implement the novel methods of this chapter, the methods for analyzing ESM data of multiple subjects, in the software package mlVAR. Furthermore, we show the functionality of this package in an empirical example by reanalyzing personality inventory items measured in an ESM design (Mõttus, Epskamp, & Francis, 2016). Finally, we assess the performance of these methods in a large-scale simulation study.

6.2 Characterizing Multivariate Gaussian Data

In this chapter we will model measurements of N subjects ($p \in 1, 2, \ldots, N$), in which subject p is measured T_p times ($t \in 1, 2, \ldots, T_p$) on I variables ($i \in 1, 2, \ldots, I$). Let $\boldsymbol{Y}$ represent the set of random variables measured in each subject:

$$\boldsymbol{Y} = \left\{ \boldsymbol{Y}^{(1)}, \boldsymbol{Y}^{(2)}, \ldots, \boldsymbol{Y}^{(N)} \right\}.$$

Element $\boldsymbol{Y}^{(p)}$ contains the random responses of a subject on all T_p measurements:

$$\boldsymbol{Y}^{(p)} = \begin{bmatrix} \boldsymbol{Y}_1^{(p)} \\ \boldsymbol{Y}_2^{(p)} \\ \vdots \\ \boldsymbol{Y}_{T_p}^{(p)} \end{bmatrix},$$

in which $\boldsymbol{Y}_t^{(p)}$ contains the row vector with random responses of subject p on time-point t, which we assume to be a *multivariate Gaussian* distribution with some mean vector $\boldsymbol{\mu}^{(p)}$ and some variance–covariance matrix $\boldsymbol{\Sigma}^{(p)}$:

$$\boldsymbol{Y}_t^{(p)} = \begin{bmatrix} Y_{t1}^{(p)} & Y_{t2}^{(p)} & \ldots & Y_{tI}^{(p)} \end{bmatrix} \sim N\left(\boldsymbol{\mu}^{(p)}, \boldsymbol{\Sigma}^{(p)} \right).$$

We will denote realizations of the above described random variables by lowercase letters (e.g., $\boldsymbol{y}$ and $y_{ti}^{(p)}$). It is important to note that measurements are nested in persons and not the other way around (two subjects can have a different number of measurements) and that the measurement t of one subject does not correspond to measurement t of another subject.[2] In typical data representation used in statistical software, $\boldsymbol{Y}_t^{(p)}$ will correspond to a row of responses from a given subject on all items. We will term these the *cases*: A case is the set of responses of a subject on a time point on all items.

Given a set of parameters $\boldsymbol{\xi}$, let $\mathcal{L}$ denote the likelihood function

$$\mathcal{L}\left(\boldsymbol{\xi};\boldsymbol{y}\right) = f_{\boldsymbol{Y}}\left(\boldsymbol{y}\mid\boldsymbol{\xi}\right),$$

in which $f_{\boldsymbol{Y}}\left(\boldsymbol{y}\mid\boldsymbol{\xi}\right)$ denotes a probability density function, which we will shorten to $f\left(\boldsymbol{y}\mid\boldsymbol{\xi}\right)$ for the remainder of this chapter. The likelihood function is crucial in estimating the set of parameters $\boldsymbol{\xi}$, either by maximum likelihood estimation (MLE) or by playing a crucial role in the formation of the posterior distribution of $\boldsymbol{\xi}$ in Bayes' rule. An important assumption in computing such a likelihood function is the *assumption of independence*. Given a set of estimated parameters $\boldsymbol{\xi}$, and assuming the model is correct, we can reasonably assume that scores of subjects are independent, allowing us to write the joint likelihood as a product of marginal likelihoods:

$$f\left(\boldsymbol{y}\mid\boldsymbol{\xi}\right) = \prod_{p=1}^{N} f\left(\boldsymbol{y}^{(p)}\mid\boldsymbol{\xi}\right).$$

Suppose we measured subjects only once and on one variable—say, their IQ level. This assumption of independence indicates, for example, that knowing Peter has an IQ of 90 does not help us predict Sarah's IQ level, given that IQ has a mean of 100 and a standard deviation of 15.

In order to fully characterize the likelihood of all observations, we further need to characterize $f(\boldsymbol{y}^{(p)}\mid\boldsymbol{\xi})$, the joint likelihood of all cases of a single subject. This is easily done in the cross-sectional example described above because every subject has only one observation. When multiple cases of a subject can be assumed to be independent, this likelihood similarly can be expressed as a product of all the likelihoods of the cases. However, as we will detail below, often the assumption of independent cases is not valid. The remainder of this section will first describe graphical models based on cross-sectional data, in which cases can be assumed to be independent, followed by a description of dependent cases for a single subject ($N = 1$) as well as for multiple subjects ($N > 1$).

6.3   The Gaussian Graphical Model

In cross-sectional data, every subject is only measured once on a set of response items. In this case, as described above, we can reasonably assume that cases are independent and thus characterize the likelihood as factorized over subjects.

---

[2]Data cannot be represented as a box (Cattell, 1988), as would be the case if subjects were all measured at fixed measurement occasions (e.g., at baseline, one week after treatment, etc.).

Because only one observation per subject is available, however, we cannot expect to estimate subject-specific mean and variance–covariance structures. It is typically assumed that the cases all share the same distribution. That is,

$$\boldsymbol{Y}^{(p)} \sim N\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \forall_p,$$

in which $\forall_p$ should be read as "for all subjects." Now, the full likelihood can be readily obtained and the mean vector $\boldsymbol{\mu}$ and variance–covariance matrix $\boldsymbol{\Sigma}$ can reliably be estimated using MLE, least squares estimation, or Bayesian estimation.

Our focus point is on $\boldsymbol{\Sigma}$. Because we assume multivariate normality, $\boldsymbol{\Sigma}$ encodes all the information necessary to determine how the observed measures relate to one another. It is to this end that great effort has been made to further model the structure of $\boldsymbol{\Sigma}$. Elements of this variance–covariance matrix can be standardized to *correlation coefficients*, allowing researchers to investigate marginal pairwise associations. This matrix, however, encodes more than just marginal associations. The *Schur complement* (Ouellette, 1981) shows that all conditional distributions of a set of variables, given another set of variables, can be obtained from blocks of $\boldsymbol{\Sigma}$. Therefore, in order to discover dynamics in psychological data, investigating the structure of $\boldsymbol{\Sigma}$ is of great importance.

However, we will not focus on $\boldsymbol{\Sigma}$ in this chapter but rather on its inverse—the *precision matrix* $\boldsymbol{K}$,

$$\boldsymbol{K} = \boldsymbol{\Sigma}^{-1},$$

also know as the GGM (Lauritzen, 1996). Of particular importance is that the standardized elements of the precision matrix encode partial correlation coefficients of two variables given all other variables:

$$\text{Cor}\left(Y_i, Y_j \mid \boldsymbol{Y}_{-(i,j)}\right) = -\frac{\kappa_{ij}}{\sqrt{\kappa_{ii}}\sqrt{\kappa_{jj}}},$$

in which κ_{ij} denotes an element of $\boldsymbol{K}$, and $\boldsymbol{Y}_{-(i,j)}$ denotes the set of variables without i and j (we dropped the person superscript for notational clarity). These partial correlations can be used as *edge weights* in a weighted network. Each variable Y_i is represented as a node, and connections (edges) between these nodes represent the partial correlation between two variables. When drawing such a network, positive partial correlations are typically visualized with green edges and negative partial correlations with red edges, and the absolute strength of a partial correlation is represented by the width and saturation of an edge (see Chapter 9). When a partial correlation is zero, we draw no edge between two nodes. As such, the GGM can be seen as a network model of conditional associations; no edge indicates that two variables are independent after conditioning on other variables in the dataset.

Figure 6.1 shows a hypothetical example of such a GGM in psychology. Three nodes represent if someone is able to concentrate well, if someone is fatigued, or if someone is suffering from insomnia. This figure shows that someone who is tired is also more likely to suffer from concentration problems and insomnia. Furthermore, this network shows that concentration problems and insomnia are conditionally independent given the level of fatigue. The GGM shown in Figure 6.1
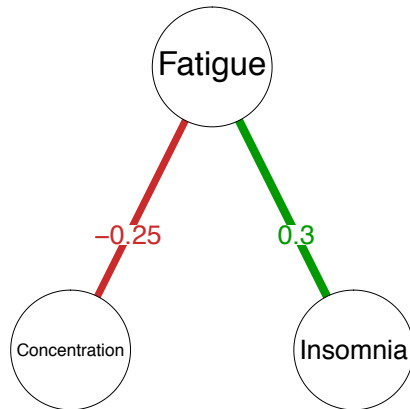
Figure 6.1: A hypothetical example of a GGM on psychological variables. Nodes represent someone's ability to concentrate, someone's level of fatigue, and someone's level of insomnia. Connections between the nodes, termed *edges*, represent partial correlation coefficients between two variables after conditioning on the third. Green edges indicate positive partial correlations, red edges indicate negative partial correlations, and the width and saturation of an edge corresponds to the absolute value of the partial correlation.

can be interpreted in three different ways: (a) potential causal relationships, (b) predictive effects and predictive mediation, and (c) genuine mutual interactions.

First, a GGM can be taken to show potential causal relationships because the structure can be equivalent to three causal structures (Pearl, 2000):

1. Concentration $\rightarrow$ Fatigue $\rightarrow$ Insomnia

2. Concentration $\leftarrow$ Fatigue $\rightarrow$ Insomnia

3. Concentration $\leftarrow$ Fatigue $\leftarrow$ Insomnia

In these structures, $\rightarrow$ denotes that what is on the left side of the arrow causes what is on the right side of the arrow. In observational data without temporal information, distinguishing between these models beyond only identifying the conditional independency is not possible. Thus, we may not know exactly why conditioning on fatigue leads to insomnia and concentration being independent—this finding may represent the smoke of a figurative causal fire. With more variables, the number of potential equivalent causal models can increase drastically (MacCallum et al., 1993).

The GGM is useful for generating hypotheses that can, at least in principle, later be experimentally tested. Specifically, the causal structures above hypothe-

size what happens when you intervene on the nodes (Pearl, 2000). For example, if we observe that someone is less able to concentrate after being fatigued in an experiment, the relationship fatigue $\rightarrow$ concentration appears more plausible. If we also observe that the person does not become more fatigued after we have experimentally impaired his or her ability to concentrate, the reverse causal relationship becomes less plausible.

Second, an edge in a GGM indicates that one node predicts a connected node after controlling for all other nodes in the network. This can be shown in the relationship between coefficients obtained from least-squares prediction and the inverse variance–covariance matrix. Let $\boldsymbol{\Gamma}$ represent an $I \times I$ matrix with zeros on the diagonal. Each row of $\boldsymbol{\Gamma}$, without the diagonal element $\boldsymbol{\gamma}_{i.-(i)}$, contains the regression coefficients obtained in

$$y_i = \tau + \boldsymbol{\gamma}_{i.-(i)} \boldsymbol{y}_{-(i)} + \varepsilon_i. \tag{6.1}$$

As such, γ_{ij} encodes how well the jth variable predicts the ith variable. This predictive effect is naturally symmetric; if knowing someone's level of insomnia predicts his or her level of fatigue, then conversely knowing someone's level of fatigue allows us to predict his or her level of insomnia. As a result, γ_{ij} is proportional to γ_{ji}. There is a direct relationship between these regression coefficients and the inverse variance–covariance matrix (Meinshausen & Bühlmann, 2006). Let $\boldsymbol{D}$ denote a diagonal matrix on which the ith diagonal element is the inverse of the ith residual variance: $d_{ii} = 1/\text{Var}(\varepsilon_i)$. Then, it can be shown (Pourahmadi, 2011) that

$$\boldsymbol{K} = \boldsymbol{D}\left(\boldsymbol{I} - \boldsymbol{\Gamma}\right). \tag{6.2}$$

Thus, γ_{ij} is proportional to κ_{ij}. A zero in the inverse variance–covariance matrix indicates that one variable does not predict another variable. Consequently, the network tells us something about the extent to which variables predict each other. In the case of Figure 6.1, the network demonstrates that both insomnia and fatigue as well as fatigue and concentration predict each other. This does not mean that knowing someone's level of fatigue does not say anything about that person's concentration problems—because these nodes are connected via an indirect path, they may correlate with each other—but merely that fatigue mediates this predictive effect. When someone's level of fatigue is known, also knowing that person's level of insomnia does not add any predictive value to that person's ability to concentrate.

Third, the network structure found in a GGM can be interpreted as showing genuine mutual causation between two nodes of the network—manipulating one node can affect the other and vise versa. Mathematically, the GGM can be shown to have the same form as the Ising model (Ising, 1925) from statistical physics (see Chapter 8), except that the Ising model only models binary data and therefore has a different normalizing constant. This is because both models are part of a class of models, called pairwise Markov random fields (Lauritzen, 1996; Murphy, 2012), which have been extensively used to model complex behavior in physical systems. For example, the Ising model represents particles with nodes and the distance between particles with edges. Particles, in essence very small magnets, are then modeled to have their north pole face up or down. Particles tend to

oriented themselves randomly at normal temperatures but align at low temperatures. Because particles tend to align at low temperatures, one particle being aligned somehow causes an adjacent particle to align in that same way and vice versa. These relationships are naturally symmetric (see Chapter 8). Applying the analogy of the Ising model for particles to the GGM shown in Figure 6.1, we could say that these symptoms tend to be in the same state (of alignment) if there is a positive connection between them and if they tend to be in different states if there is a negative connection. In this system, a person suffering from fatigue could also suffer from insomnia as well as concentration problems (van Borkulo et al., 2014).

Estimation

The maximum likelihood solution of K can readily be obtained by standardizing the inverse sample variance–covariance matrix and by multiplying all off-diagonal elements by -1. An interesting problem pertains to elements of K which are close to, but not exactly, zero. In the interest of parsimony, researchers may want to remove these edges and obtain conditional independence with fewer parameters in the model. One way to obtain this is to use the sampling distribution of the partial correlation coefficients represented in K to test if edges are significantly different from zero. The network can then be thresholded by removing the nonsignificant edges (by fixing them at zero). Alternatively, lengthy model search algorithms can be applied to iteratively add and remove edges. In recent literature, it has become increasingly popular to use regularization techniques, such as penalized MLE, to jointly estimate model structure and parameter values (van Borkulo et al., 2014; see also Chapter 2). In particular, the *least absolute shrinkage and selection operator* (LASSO; Tibshirani, 1996) has been shown to perform well in quickly estimating model structure and parameter estimates of a sparse GGM (Friedman et al., 2008; Meinshausen & Bühlmann, 2006; Yuan & Lin, 2007). A particularly fast variant of LASSO is the *graphical LASSO* (glasso; Friedman et al., 2008), which directly penalizes elements of the inverse variance–covariance matrix (Witten, Friedman, & Simon, 2011; Yuan & Lin, 2007). In addition, glasso utilizes a tuning parameter that controls the sparsity of the network: A sparse network is one with few edges (i.e., K contains mostly zeros). The tuning parameter can be chosen in a way that optimizes cross-validated prediction accuracy or that minimizes information criteria such as the extended Bayesian information criterion (EBIC; Chen & Chen, 2008). Estimating a GGM with the glasso algorithm in combination with EBIC model selection has been shown to work well in retrieving the true network structure (Foygel & Drton, 2010; see also Chapter 2) and is currently the dominant method for estimating the GGM in psychological data (see also Chapter 2 for an introduction to this methodology aimed at empirical researchers).

Figure 6.2 shows an example of a GGM estimated using glasso in combination with EBIC model selection. This network was estimated on the `bfi` dataset from the *psych* R package (Revelle, 2010). This dataset contains the responses of 2,800 people on 25 items designed to measure the Big Five personality traits (McCrae & Costa, 1997). The network shows many meaningful connections, such as "make friends easily" being linked to "make people feel at ease," "don't talk

Agreeableness
- A1: Am indifferent to the feelings of others.
- A2: Inquire about others' well–being.
- A3: Know how to comfort others.
- A4: Love children.
- A5: Make people feel at ease.

Conscientiousness
- C1: Am exacting in my work.
- C2: Continue until everything is perfect.
- C3: Do things according to a plan.
- C4: Do things in a half–way manner.
- C5: Waste my time.

Extraversion
- E1: Don't talk a lot.
- E2: Find it difficult to approach others.
- E3: Know how to captivate people.
- E4: Make friends easily.
- E5: Take charge.

Neuroticism
- N1: Get angry easily.
- N2: Get irritated easily.
- N3: Have frequent mood swings.
- N4: Often feel blue.
- N5: Panic easily.

Openness
- O1: Am full of ideas.
- O2: Avoid difficult reading material.
- O3: Carry the conversation to a higher level.
- O4: Spend time reflecting on things.
- O5: Will not probe deeply into a subject.

Figure 6.2: An example of a network model estimated on the BFI personality dataset from the psych package in R (cross-sectional data, $N = 2,800$). Nodes represent variables (in this case personality inventory items) and edges between the nodes represent partial correlation coefficients. The network was estimated using the glasso in combination with EBIC model selection, using the `EBICglasso` function in the *qgraph* package.

a lot" being linked to "find it difficult to approach others," and "carry the conversation to a higher level" being linked to "know how to captivate people." For a detailed discussion on the interpretation of such models in personality research (see Chapter 10).

The GGM can be estimated by inverting and standardizing the sample variance–covariance matrix, which can be done in the open-source statistical programming language R (R Core Team, 2016) by using the *corpcor* (Schäfer et al., 2015) or *qgraph* (Epskamp et al., 2012) R package. The *qgraph* package also supports thresholding via significance testing or false discovery rates. The glasso algorithm is implemented in the packages glasso (Friedman et al., 2014) and *huge* (Zhao et al., 2015). The huge package also allows for selection of the tuning parameter using cross-validation or EBIC. The EBIC-based tuning parameter selection with the glasso package, using only a variance–covariance matrix as input, has been implemented in the *qgraph* package. The *parcor* package (Krämer et al., 2009) implements other LASSO variants for estimating the GGM. Finally, fitting an estimated GGM to data can be done in the R packages *ggm* (Marchetti, Drton, & Sadeghi, 2015) and *lvnet* (see Chapter 7).

6.4 When Cases Are Not Independent: $n = 1$

Different kinds of data in psychology occur when only one subject is measured several times. The likelihood is fully characterized by the likelihood of the single subject:

$$f(\boldsymbol{y} \mid \boldsymbol{\xi}) = f(\boldsymbol{y}^{(1)} \mid \boldsymbol{\xi}).$$

We need to characterize $f(\boldsymbol{y}^{(p)} \mid \boldsymbol{\xi})$. In repeated measures of psychological constructs—assuming a reasonably short interval between consecutive measurements is typical in ESM studies—we cannot reasonably assume that cases are independent. For example, suppose we measured Peter multiple times on his level of fatigue, measured on a scale from 1 (*not at all fatigued*) to 10 (*extremely fatigued*). Suppose we know Peter has an average fatigue level of 5 with a standard deviation of 1. Knowing that Peter scored a 2 at some time point, we can make a better prediction regarding the level of Peter's fatigue at the next time point (it is probably still low a few hours later) than if we only knew his mean and standard deviation, which would predict this level most likely to be somewhere between 3 and 7. This is because someone's fatigue, like most psychological and physiological states, is likely to show some stability over a time interval of several hours.

It is important to note that $f(\boldsymbol{y} \mid \boldsymbol{\xi})$ cannot be computed by multiplying the marginal likelihoods of every response. Instead, we now need to express the full joint likelihood. When we drop superscript $^{(1)}$ denoting the single subject, this becomes

$$f(\boldsymbol{y} \mid \boldsymbol{\xi}) = f\left(\boldsymbol{y}_T \mid \boldsymbol{y}_1, \ldots, \boldsymbol{y}_{T_p-1}, \boldsymbol{\xi}\right) \cdots f(\boldsymbol{y}_3 \mid \boldsymbol{y}_1, \boldsymbol{y}_2, \boldsymbol{\xi}) f(\boldsymbol{y}_2 \mid \boldsymbol{y}_1, \boldsymbol{\xi}) f(\boldsymbol{y}_1 \mid \boldsymbol{\xi}).$$

The model above, although fully characterizing the joint likelihood, is not estimable without stringent assumptions, so we make the following assumptions.

1. The joint probability distribution can be factorized according to a graph.

2. The conditional probability distributions are stable and independent of t.

3. The first measurements are treated as exogenous and not modeled.

4. The conditional distributions are multivariate normal.

The first assumption is that the time series follows some graph structure such that the factorization of the joint probability distribution can be made easier. Figure 6.3 shows three such potential graph structures. The first panel shows the *Lag 0* factorization, in which each observation is assumed to be independent of others. As described above, although this is a sparse representation, the Lag 0 model is not plausible in most time-series psychological datasets. As such, we could use the graph factorization of the second panel of Figure 6.3 instead, denoting the *Lag 1* factorization

$$f(\boldsymbol{y} \mid \boldsymbol{\xi}) = f\left(\boldsymbol{y}_T \mid \boldsymbol{y}_{T_p-1}, \boldsymbol{\xi}\right) \cdots f(\boldsymbol{y}_2 \mid \boldsymbol{y}_1, \boldsymbol{\xi}) f(\boldsymbol{y}_1 \mid \boldsymbol{\xi}).$$

This is a powerful factorization because it does not assume that measurement are independent of one another. For example, the Lag 1 factorization does not
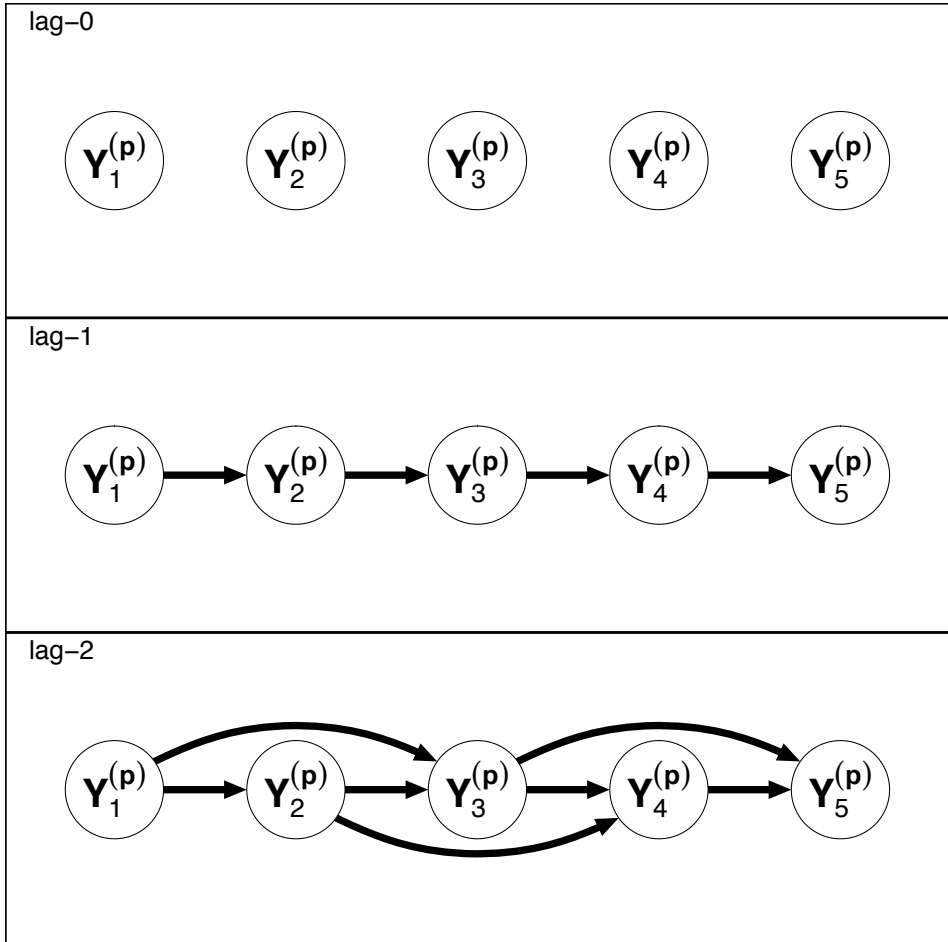
Figure 6.3: Here are three possible graph factorizations of the within-subject likelihood of subject p. Each node represents a vector of measurements at some time point. The top panel shows the Lag 0 factorization, indicating that cases are independent. Because this is usually not a plausible assumption in psychology, we can instead use another factorization. The middle panel shows the Lag 1 factorization, indicating that cases are independent given only the previous case. The bottom panel shows the Lag 2 factorization, indicating that cases are independent given the past two cases.

assume there are no *Lag 2* correlations—correlations between $\boldsymbol{y}_{t-2}$ and $\boldsymbol{y}_t$ for any t—but instead that these Lag 2 correlations can fully be explained by the Lag 1 interactions, $\boldsymbol{y}_{t-2} \perp\!\!\!\perp \boldsymbol{y}_t \mid \boldsymbol{y}_{t-1}$ for all t. However, more flexible models can be specified as well, such as the Lag 2 model shown in the last panel of Figure 6.3. Because the number of observations in ESM studies is often relatively low (people cannot be expected to fill out tens of questionnaires daily over the course of several days), adding complexity to the model requires more observations to reliably estimate parameter values. Therefore, we will only describe the Lag 1 factorization in the remainder of this chapter because this factorization is the simplest that also controls for the most obvious violations of independence between consecutive cases.

The second assumption is that the conditional probability distributions do not depend on t and are thus stable over time. This is called the assumption of *stationarity*. Using this assumption, the time series of a single subject now features multiple observations of the same relationship (e.g., the Lag 1 relationship), making the model estimable. Combining this with the third assumption of first measurements being treated as exogenous, and thereby not modeled, renders the probability distribution simple and straightforward. For example, the Lag 1 factorization then becomes

$$f\left(\boldsymbol{y} \mid \boldsymbol{y}_1, \boldsymbol{\xi}\right)_{\text{lag}-1} = \prod_t f\left(\boldsymbol{y}_t \mid \boldsymbol{y}_{t-1}, \boldsymbol{\xi}\right).$$

The assumption of stationarity is not trivial because people can develop over time. In a typical ESM study, data are gathered in a relatively short time span (e.g., a few weeks). Assuming a person stays relatively stable in such a short interval is much more plausible. It is therefore important to note that the assumption of stability does not assume a person never changes, merely that the person's scores are distributed similarly in a relatively short time span (Fleeson, 2001).

Finally, we assume that these conditional distributions are multivariate normal. Using the Schur complement, these distributions can be shown to be equivalent to a linear regression model with correlated multivariate normal residuals. We can, without loss of information, center the lagged predictors such that we obtain

$$\boldsymbol{Y}_t \mid \boldsymbol{y}_{t-1} \sim N\left(\boldsymbol{\mu} + \boldsymbol{B}\left(\boldsymbol{y}_{t-1} - \boldsymbol{\mu}\right), \boldsymbol{\Theta}\right),$$

in which $\boldsymbol{B}$ denotes an $I \times I$ matrix of *temporal* effects, $\boldsymbol{\mu}$ denotes the I length vector of stationary means, and $\boldsymbol{\Theta}$ denotes the $I \times I$ variance–covariance matrix conditional on the previous time point, which we will term *contemporaneous* effects. This model is also known as the VAR because it can be seen as a multivariate multiple regression on the previous time point. VAR has become popular in psychology because $\boldsymbol{B}$ encodes temporal prediction: Element $\beta_{ij}^{(p)}$ being nonzero means that Y_{ti} is predicted by $Y_{t-1,j}$. Such a temporal prediction is termed *Granger causality* in the economic literature (Eichler, 2007; Granger, 1969) because it satisfies at least the temporal requirement for causation (i.e., cause must precede the effect).

This model implies a stationary distribution of cases:

$$\boldsymbol{y}_t \sim N\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right),$$
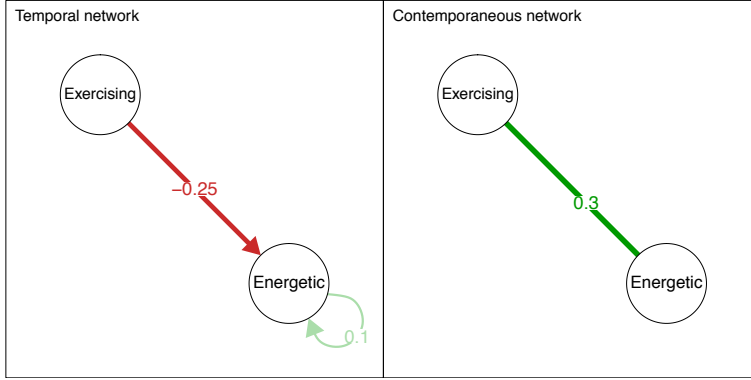
Figure 6.4: A hypothetical example of two network structures obtained from a vector-autoregression model. The network on the left indicates the temporal network, demonstrating that a variable predicts another variable at the next time point. The network on the right indicates the contemporaneous network, demonstrating that two variables predict each other in the same time point.

in which $\boldsymbol{\Sigma}$ can be obtained from $\boldsymbol{B}$ and $\boldsymbol{\Theta}$, making use of the vectorization operator Vec and the Kronecker product $\otimes$:

$$\text{Vec}\left(\boldsymbol{\Sigma}\right) = \left(\boldsymbol{I} - \boldsymbol{B} \otimes \boldsymbol{B}\right)^{-1} \text{Vec}\left(\boldsymbol{\Theta}\right). \tag{6.3}$$

A proof of Equation (6.3) is beyond the scope of this chapter and can be requested from the author. It is important to note that in addition to the estimation of temporal effects, the VAR model can be used to obtain the GGM (the inverse of the variance–covariance matrix described above) for nonindependent cases. We can further note that if cases are independent, $\boldsymbol{B} = \boldsymbol{O}$ and subsequently $\boldsymbol{\Sigma} = \boldsymbol{\Theta}$. Thus, the GGM is a special case of the VAR model. This leads to a strikingly different interpretation of the VAR model; the VAR model can be seen as an inclusion of temporal effects on a GGM.

Contemporaneous Causation

In order to disentangle the temporal and contemporaneous relationship, it is best not to combine them into a single GGM model but rather to investigate them separately. Following Wild et al. (2010), the inverse of $\boldsymbol{\Theta}$ can be standardized to a GGM model encoding residual partial correlation coefficients, which can be drawn in a network model. As a result, the VAR model returns two network models: the *temporal network*, a directed network indicating temporal prediction or Granger causality, and the *contemporaneous network*, a partial-correlation network of effects in the same window of measurement. Both network structures can highlight potential causal pathways. In psychology, there will likely be many causal relationships that occur much faster than the lag interval in a typical ESM study; in which case, these pathways will be captured in the contemporaneous network.

For example, if someone is experiencing bodily discomfort, that will immediately negatively affect that person's ability to enjoy him or herself.

Figure 6.4 shows a hypothetical example of the two network structures obtained in a VAR analysis. The left panel shows the temporal network (a graphical representation of $\boldsymbol{B}$). This network shows that whenever the subject in question felt energetic (or tired) this person also felt more (or less) energetic in the next measurement. The temporal network also shows us that after exercising, this person felt less energetic as well. The contemporaneous network in the right panel (a graphical representation of the GGM based on $\boldsymbol{\Theta}$) shows a plausible reverse relationship: Whenever this person exercised, he or she felt more energetic in the same measurement occasion.

Estimation

We can estimate the VAR model by specifying it as a regression model. Without the loss of information, we can center the variables to have a mean of zero. The corresponding multivariate regression model then becomes

$$\boldsymbol{y}_t = \boldsymbol{B}\boldsymbol{y}_{t-1} + \boldsymbol{\varepsilon}_t$$
$$\boldsymbol{\varepsilon}_t \sim N(\boldsymbol{0}, \boldsymbol{\Theta}).$$

Alternatively, the VAR model can be estimated in steps using separate *univariate* models for every variable:

$$y_{ti} = \boldsymbol{\beta}_i \boldsymbol{y}_{t-1} + \varepsilon_{ti}$$
$$\varepsilon_{ti} \sim N(0, \sqrt{\theta_{ii}}),$$

in which $\boldsymbol{\beta}_i$ denotes the ith row of $\boldsymbol{B}$. Figure 6.5 shows the difference between univariate and multivariate estimation. In univariate estimation, every model contains a different subset of the parameters of interest. In addition, the contemporaneous covariance θ_{12} is not obtained in any of the models and needs to be estimated post hoc by correlating the residuals of both regression models.

Abegaz and Wit (2013) proposed to apply LASSO estimation of $\boldsymbol{B}$ and $\boldsymbol{\Theta}$ using the multivariate regression with the covariance estimation (MRCE) algorithm described by Rothman et al. (2010). MRCE involves iteratively optimizing $\boldsymbol{B}$ using cyclical-coordinate descent and $\boldsymbol{K}$ using the glasso algorithm (Friedman et al., 2008, 2014). EBIC model selection can be used to obtain the best performing model. This methodology has been implemented in two open source R packages: *sparseTSCGM* (Abegaz & Wit, 2015), which aims to estimate the model on repeated multivariate genetic data, and *graphicalVAR* (Epskamp, 2015), which was designed to estimate the model on the psychological data of a single subject. The *graphicalVAR* package also allows for unregularized multivariate estimation.

6.5 When Cases Are Not Independent: $n > 1$

When multiple subjects are measured, we need to characterize the likelihood for every subject. Using the assumptions described above, we can model the time
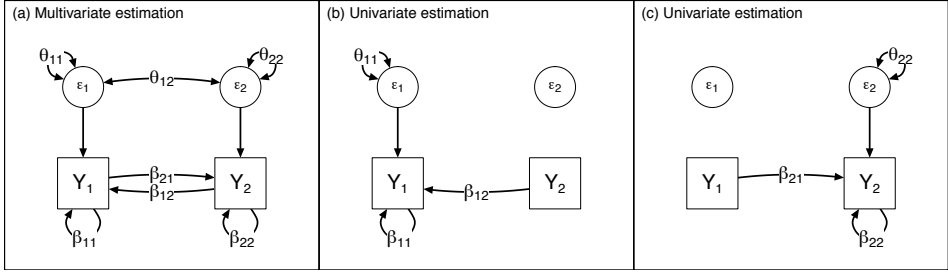
Figure 6.5: A multivariate and univariate estimation of a VAR model with two observed variables. Intercepts are not shown to improve clarity. Panel A shows that in multivariate estimation, the entire model is estimated at once whereas Panels B and C show that in sequential estimation two separate models are estimated.

series of each subject with a subject-specific VAR model:

$$\boldsymbol{y}_t^{(p)} \mid \boldsymbol{y}_{t-1}^{(p)} \sim N\left(\boldsymbol{\mu}^{(p)} + \boldsymbol{B}^{(p)}\left(\boldsymbol{y}_{t-1}^{(p)} - \boldsymbol{\mu}^{(p)}\right), \boldsymbol{\Theta}^{(p)}\right).$$

Often, however, researchers are not interested in the dynamics of a single participant but rather in the generalizability of dynamics over multiple subjects. To this end, researchers may want to estimate the average effects and interindividual differences of such intraindividual dynamics. We can model these by using the language of multilevel modeling (Bringmann et al., 2013). For each parameter, we denote the average effect as the *fixed effects*, $\boldsymbol{f}$, and the person-level deviance from this mean as the *random effects*, $\boldsymbol{R}^{(p)}$, with the realization $\boldsymbol{r}^{(p)}$. Using this notation, the parameter vector of a single subject becomes

$$\begin{bmatrix} \boldsymbol{\mu}^{(p)} \\ \text{Vec}\left(\boldsymbol{B}^{(p)}\right) \\ \text{Vech}\left(\boldsymbol{\Theta}^{(p)}\right) \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_{\boldsymbol{\mu}}^{(p)} \\ \boldsymbol{f}_{\boldsymbol{B}}^{(p)} \\ \boldsymbol{f}_{\boldsymbol{\Theta}}^{(p)} \end{bmatrix} + \begin{bmatrix} \boldsymbol{r}_{\boldsymbol{\mu}}^{(p)} \\ \boldsymbol{r}_{\boldsymbol{B}}^{(p)} \\ \boldsymbol{r}_{\boldsymbol{\Theta}}^{(p)} \end{bmatrix},$$

in which Vec stacks the columns of a matrix, and Vech does the same but only takes the upper-triangular elements including the diagonal. The random effects are centered on zero:

$$\mathcal{E}\left(\begin{bmatrix} \boldsymbol{R}_{\boldsymbol{\mu}} \\ \boldsymbol{R}_{\boldsymbol{B}} \\ \boldsymbol{R}_{\boldsymbol{\Theta}} \end{bmatrix}\right) = \boldsymbol{0},$$

such that the fixed effects reflect the population means of the parameters. The variance of the random effects can be interpreted as the *individual differences*.

The fixed effects and random effect variances and covariances can be estimated by estimating a VAR model for every subject, pooling the parameter estimates, and computing the mean (fixed effects) and variance–covariance matrix (random effects distribution). This estimation, however, is separate for every subject. To combine all observations in a single model, we can assign distributions over the parameters; in which case, we make use of multilevel modeling. Assigning distributions has two main benefits. First, instead of having a single parameter per

subject, we now only need to estimate the parameters of the distribution. For example, when we model observations from 100 subjects, instead of estimating each parameter 100 times, we now only need to estimate its mean and variance. Second, the multilevel structure acts as a prior distribution in Bayesian estimation procedures—in case we wish to obtain person-specific parameter estimates post hoc. In particular, multilevel modeling leads to *shrinkage*; parameter values that are very different from the fixed effects are likely to be estimated closer to the fixed effect in multilevel modeling than when using a separate model for every subject. For example, if we estimate a certain temporal regression in five people and find the values 1.1, 0.9, 0.7, 1.3, and 10, it is likely that the fifth statistic, 10, is an outlier. Ideally, we would estimate this value to be closer to the other values.

Modeling and estimating a random distribution for the contemporaneous variance–covariance matrix is still a topic for future research and not readily implemented in open-source software. This is mainly because these matrices must be positive definite. We cannot simply assign normal distributions to elements of the contemporaneous (partial) variance–covariance matrix because doing so might lead to nonzero probability of matrices that are not positive definite. Therefore, we do not define this distribution here and merely state that there is some population mean for its elements, f_Θ. We assume the means and lagged regression parameters to be normally distributed:

$$\begin{bmatrix} R_\mu \\ R_B \end{bmatrix} \sim N \left(0, \begin{bmatrix} \Omega_\mu & \Omega_{\mu B} \\ \Omega_{B\mu} & \Omega_B \end{bmatrix} \right).$$

To summarize, the multilevel VAR model makes use of the following parameters for all subjects:

- f_B: The average within-person temporal relationships between consecutive time points.

- f_Θ: The average within-person contemporaneous relationships.

- Ω_μ: The between-person relationships between observed variables.

- $\Omega_{\mu B}$ and Ω_B: Individual differences between the temporal relationships and other temporal relationships or the means. Of particular interest is $\sqrt{\text{Diag}(\Omega_B)}$, which shows the individual differences of each temporal relationship (Bringmann et al., 2013).

For any researcher interested in investigating results of particular subjects, the subject-specific structures are also of interest:

- $\mu^{(p)}$: The stationary means of subject p.

- $B^{(p)}$: The within-person temporal relationships of subject p.

- $\Theta^{(p)}$: The within-person contemporaneous relationships of subject p.

A New Look at Cross-Sectional Analysis

Such variance decomposition exposes a major limitation of cross-sectional analyses. In cross-sectional data, each subject is only measured once: $T_1 = T_2 = \ldots = T_P = 1$. This can be seen as a special case of the multilevel VAR model in which the Lag 0 factorization is used to model the single response of a subject. This single response can then be written as the stationary mean of person p and the deviation from that mean:

$$\boldsymbol{y}_1^{(p)} = \boldsymbol{\mu}^{(p)} + \boldsymbol{\varepsilon}_1^{(p)}$$
$$\boldsymbol{\varepsilon}_1^{(p)} \sim N(\boldsymbol{0}, \boldsymbol{\Theta}^{(p)}).$$

It is immediately clear that $\boldsymbol{\Theta}^{(p)}$ cannot be estimated from a single set of responses. Moreover, even if we assume that within-person contemporaneous effects are equal across people and drop the superscript (p), this still leaves us without an estimable model because $\boldsymbol{R_\mu}$ is also assumed to be normally distributed. Therefore, we get

$$\boldsymbol{\varepsilon}_1^{(p)} \sim N(\boldsymbol{0}, \boldsymbol{\Theta})$$
$$\boldsymbol{R_\mu} \sim N(\boldsymbol{0}, \boldsymbol{\Omega_\mu}).$$

In no way do we know if deviations from the grand mean are due to the within-person variance in $\boldsymbol{\Theta}$ or the between-person variance in $\boldsymbol{\Omega_\mu}$. Thus, in cross-sectional analysis, within- and between-subject variances are not distinguishable. We can estimate $\boldsymbol{\Theta}$ by assuming $\boldsymbol{\Omega_\mu} = \boldsymbol{O}$, or we can estimate $\boldsymbol{\Omega_\mu}$ by assuming $\boldsymbol{\Theta} = \boldsymbol{O}$. Both assumptions lead to the exact same estimates. This does not mean that cross-sectional analysis is unusable by default because the obtained structure can highlight potential causal relationships between variables; however, it cannot disentangle between-subject relationships from short-term, within-subject relationships (Hamaker, 2012).

Between-Subjects Causation

The variance–covariance matrix $\boldsymbol{\Omega_\mu}$ encodes how variables relate to one another across subjects and can be modeled using a GGM network of partial correlation coefficients. As such, the multilevel VAR model returns three types of network structures describing relationships between observed variables. In addition to the temporal and contemporaneous network fixed effects (the average temporal and contemporaneous network) and random effects (the personal deviations from these averages), the multilevel VAR model also returns a *between-subjects network*—the network structure between stationary means of subjects based on $\boldsymbol{\Omega_\mu}$.

Hamaker (2012) described an example of how within- and between-person effects can strongly differ from each other. Suppose we let people write several texts, and we measure the number of spelling errors they made and the number of words per minute they typed (typing speed). We would expect the seemingly paradoxical three network structures shown in Figure 6.6. First, we would not expect the temporal network to show any relationships. There is no logical reason to assume that observing someone type a text faster than his or her average has any influence on the number of spelling errors in the next text. Second, we expect
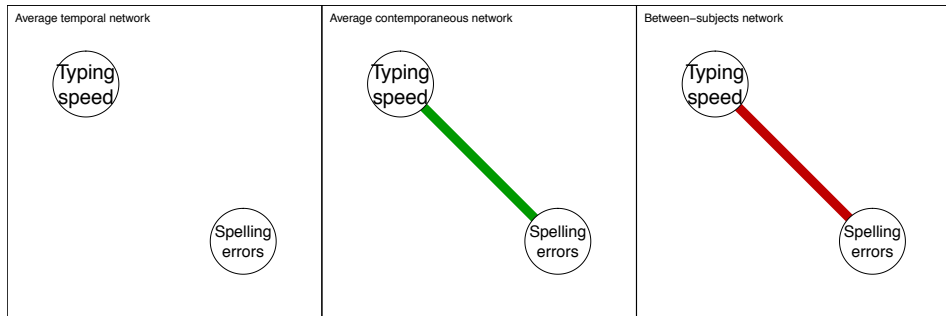
Figure 6.6: A hypothetical example three network structures obtained from a multilevel vector-autoregression model. The network on the left indicates the temporal network, showing that a variable predicts another variable at the next time point. The network in the middle indicates the contemporaneous network, showing that personal deviations from the means predict each other at the same time point. Finally, the network on the right indicates the between-subjects network, showing how the *means* of different subjects relate to one another

a positive relationship in the contemporaneous network. When a person types faster than his or her average typing speed, that person will make more spelling errors. Finally, we expect a negative relationship in the between-person network (e.g., people who type fast, on average, generally make fewer spelling errors). This is because people who type fast, on average, are likely to be more skilled in writing (e.g., a court room stenographer) and therefore are less likely to make a lot of spelling errors, compared to someone who types infrequently.

The different ways of thinking about the effects of manipulations in time-series models can be organized in terms of recently developed interventionist accounts of causation (Woodward, 2005). According to Woodward, causation is fleshed out in terms of interventions: X is a cause of Y if an intervention (natural or experimental) on X would lead to a change in Y. Statistically, the interventionist account is compatible with, for example, Pearl's (2000) semantics in terms of a "do-operator." Here, an intervention on X is represented as $\mathrm{Do}\,(X = x)$, and the causal effect on Y is formally expressed as $\mathbb{E}\,(Y \mid \mathrm{Do}\,(X = x))$. Pearl distinguished this from the classical statistical association, in which no intervention is present, and we get the ordinary regression $\mathbb{E}\,(Y \mid \mathrm{See}\,(X = x))$. This is a useful notation because it immediately raises the important point that there is a difference between doing and seeing, which of course parallels the classic distinction between experimental and correlational research (Cronbach & Meehl, 1955).

Cashing out causal effects in terms of interventions is useful to get a grip on the causal information in the different GGMs defined in this chapter. In a time-series model, interventions on variables can be conceptualized in different ways. In particular, consider the intervention $\mathrm{Do}\,(X = x)$. We can think of this in terms of a random shock to the system, which sets X to value x on a particular time point and evaluates the effect on another variable Y on the next time point (or series of time points as in continuous time models). If we want to gauge this

type of causal relationship, we may look at the within-subjects VAR model. To take Hamaker's (2014) example, say we want to know what would happen to the number of typing errors: If a researcher forced a given person to type very fast, that researcher would need to evaluate the within-person data, which would show a negative association between typing speed and typing errors. Between-subject data would be misleading because individual differences would probably yield a positive correlation between speed and accuracy—faster typists are likely to be more accurate typists.

However, we can also think of a manipulation that sets X to value x in a different way, for instance, by inducing a long-term change in the system that leads it to converge on $X = x$ in expectation. To evaluate the effect of this type of intervention, we should consider the behavior of the system on the changes of the intercept of X. Clearly, in order to evaluate this type of intervention, the within-subject time-series model is useless (as per stationarity). However, the between-subjects model may contain important clues because it contains the relationships between the long-term averages across people. Thus, if we want to gauge the effect of a long-term change (most plausibly conceptualized as a change in intercept), the between-subjects model is a better guide. In terms of Hamaker's (2014) example, if we are interested in the effect of changing someone's typing speed structurally (e.g., by training a person systematically), our preferred source of causal information would likely lie in the between-subjects model because the parameters of the within-subjects model would undoubtedly lead to the wrong conclusion.

Estimation

A straightforward technique is to estimate separate VAR models for each subject. Afterwards, fixed effects (i.e., average effects in the population) can be estimated by pooling the parameters and averaging them. This estimation technique is relatively fast even for large models, but it requires a high number of observations per person. As described above, an alternative is to use multilevel modeling (Hamaker, 2012). The benefit of the latter approach is that instead of estimating the VAR model in each subject, only the fixed effects and variance–covariance of the random effects need to be estimated. This can be done by integrating over the distribution of the random effects or by specifying the model using hierarchical Bayesian Monte-Carlo sampling methods (Gelman & Hill, 2006; Schuurman, Grasman, & Hamaker, 2016). Here, we propose a novel two-step, multilevel estimation procedure that estimates the fixed effects for the temporal, between-subjects, and contemporaneous networks as well as the random effects for the temporal and contemporaneous networks. The contemporaneous networks are estimated in a second step, by analyzing the residuals of the first step.

Temporal network. Although multivariate multilevel estimation is possible in theory, it is computationally expensive in practice. For example, when we want to explore potential dynamics in medium-sized ESM datasets on around 10 to 20 variables, multivariate multilevel estimation becomes very slow in both MLE

and Bayesian estimation. Therefore, we only describe univariate estimation procedures (Bringmann et al., 2013). Because the joint conditional distribution of $\boldsymbol{y}_t^{(p)} \mid \boldsymbol{y}_{t-1}^{(p)}$ is normal, it follows that the marginal distribution of every variable is univariate normal and can be obtained by dropping all other parameters from the distribution:

$$y_{ti}^{(p)} \mid \boldsymbol{y}_{t-1}^{(p)} \sim N\left(\mu_i^{(p)} + \boldsymbol{\beta}_i^{(p)}\left(\boldsymbol{y}_{t-1}^{(p)} - \boldsymbol{\mu}^{(p)}\right), \theta_{ii}^{(p)}\right),$$

in which $\boldsymbol{\beta}_i$ indicates the row vector of the ith row of $\boldsymbol{B}^{(p)}$. When drawn as a temporal network, the edges point to node i. Many software packages do not allow the estimation of $\boldsymbol{\mu}^{(p)}$ as described above. In this case, the sample means of every subject, $\bar{\boldsymbol{y}}^{(p)}$, can be taken as a substitute for $\boldsymbol{\mu}^{(p)}$ (Hamaker & Grasman, 2014). The model then becomes a univariate multilevel regression model with within-subject centered predictors, estimable by functions such as the `lmer` in `lme4` (Bates, Mächler, Bolker, & Walker, 2015). The Level 1 model becomes

$$y_{ti}^{(p)} = \mu_i^{(p)} + \boldsymbol{\beta}_i^{(p)}\left(\boldsymbol{y}_{t-1}^{(p)} - \bar{\boldsymbol{y}}^{(p)}\right) + \varepsilon_{ti}^{(p)} \tag{6.4}$$

$$\varepsilon_{ti}^{(p)} \sim N(0, \theta_{ii}^{(p)}), \tag{6.5}$$

and the Level 2 model becomes

$$\begin{bmatrix} \mu_i^{(p)} \\ \boldsymbol{\beta}_i^{(p)} \end{bmatrix} = \begin{bmatrix} f_{\mu_i} \\ \boldsymbol{f}_{\boldsymbol{\beta}_i} \end{bmatrix} + \begin{bmatrix} r_{\mu_i}^{(p)} \\ \boldsymbol{r}_{\boldsymbol{\beta}_i}^{(p)} \end{bmatrix}$$

$$\begin{bmatrix} R_{\mu_i} \\ \boldsymbol{R}_{\boldsymbol{\beta}_i} \end{bmatrix} \sim N\left(\mathbf{0}, \begin{bmatrix} \omega_{\mu_i} & \boldsymbol{\Omega}_{\boldsymbol{\mu}\boldsymbol{\beta}_i} \\ \boldsymbol{\Omega}_{\boldsymbol{\beta}_i\boldsymbol{\mu}} & \boldsymbol{\Omega}_{\boldsymbol{\beta}_i} \end{bmatrix}\right).$$

Estimation of such univariate models only requires the numeric approximation of an $I + 1$ dimensional integral, which is much easier to compute. Therefore, sequential estimation using univariate models have been used in estimating multilevel VAR models (Bringmann et al., 2013). A downside, however, is that not all parameters are included in the model. In particular, off-diagonal elements of $\boldsymbol{\Theta}^{(p)}$ and $\boldsymbol{\Omega}_{\boldsymbol{\mu}}$ as well as certain elements of $\boldsymbol{\Omega}_{\boldsymbol{\mu}\boldsymbol{B}}$ and $\boldsymbol{\Omega}_{\boldsymbol{B}}$ are not obtained. A second downside is that estimating correlated random effects does not work well for models with many predictors. In particular, `lmer` becomes very slow with approximately more than eight predictors. As such, networks with more than eight nodes are hard to estimate. To estimate larger networks (e.g., 20 nodes), we can choose to estimate uncorrelated random effects, which we term *orthogonal estimation*. The performance of orthogonal estimation, although the random effects are in reality correlated, is assessed in the simulation study below.

Between-subjects network. To obtain estimates of between-subject effects, Hamaker and Grasman (2014) suggest that the sample means of every subject, $\bar{\boldsymbol{y}}^{(p)}$ in Equation (6.4), can be included as predictors at the subject level. With this extension, the Level 2 model for the person-specific mean of the ith variable now becomes

$$\mu_i^{(p)} = f_{\mu_i} + \boldsymbol{\gamma}_{\mu,i}\bar{\boldsymbol{y}}_{-(i)}^{(p)} + r_{\mu_i}^{(p)}, \tag{6.6}$$

in which we use $\boldsymbol{\gamma}_{\mu,i}$ to denote the ith row (without the diagonal element i) of a $I \times I$ matrix $\boldsymbol{\Gamma}_\mu$, and $\bar{\boldsymbol{y}}_{-(i)}^{(p)}$ denotes the vector $\bar{\boldsymbol{y}}^{(p)}$ without the i-th element. Because $\bar{y}_i^{(p)}$ is itself an estimate of $\mu_i^{(p)}$, Equation (6.6) seems to take the form of Equation (6.1). As such, these estimates can be used, as seen in Equation (6.2), to estimate a GGM between the means (Lauritzen, 1996; Meinshausen & Bühlmann, 2006)— the between-subjects network. Due to the estimation in a multilevel framework, the resulting matrix will not be perfectly symmetric and must be made symmetric by averaging lower and upper triangular elements. Thus, each edge (i.e., partial correlation) in the between-subjects network is estimated by standardizing and averaging two regression parameters: the parameter denoting how well mean A predicts mean B and the regression parameter denoting how well mean B predicts mean A.

Contemporaneous network.   An estimate for contemporaneous networks can be obtained in a second step by investigating the residuals of the multilevel model that estimate the temporal and between-subject effects. These residuals can be used to run multilevel models that predict the residuals of one variable from the residuals of other variables at the same time point. Let $\hat{\varepsilon}_{ti}^{(p)}$ denote the estimated residual of variable i at time point t of person p, and let $\hat{\boldsymbol{\varepsilon}}_{t,-(i)}^{(p)}$ denote the vector of residuals of all other variables at this time point. The Level 1 model then becomes

$$\hat{\varepsilon}_{ti}^{(p)} = \tau_i^{(p)} + \boldsymbol{\gamma}_{\varepsilon,i}^{(p)} \hat{\boldsymbol{\varepsilon}}_{t,-(i)}^{(p)} + \zeta_{ti}^{(p)}, \tag{6.7}$$

in which $\boldsymbol{\gamma}_{\varepsilon,i}^{(p)}$ represents the i-th row (without the diagonal element i) of a $I \times I$ matrix, $\boldsymbol{\Gamma}_\mu^{(p)}$, $\tau_i^{(p)}$ represents some intercept, and $\zeta_{ti}^{(p)}$ represents a residual. In the Level 2 model, we again assign a multivariate normal distribution to parameters $\tau_i^{(p)}$ and $\boldsymbol{\gamma}_{\varepsilon,i}^{(p)}$. It can be seen that Equation (6.7) also takes the form of Equation (6.1). Thus, this model can again be seen as the node-wise GGM estimation procedure. Estimates of both the person-specific and fixed-effects contemporaneous networks can be obtained by using Equation (6.2), where again the matrices need to be made symmetric by averaging upper and lower triangle elements. As with the temporal network, orthogonal estimation can be used when the number of variables is large (i.e., larger than approximately eight).

Thresholding.   After estimating network structures, researchers may be interested in removing edges that may be spurious and due to sampling error. By setting edge weights to zero, effectively removing edges from a network, a sparse network is obtained that is more easily interpretable. One method of doing so is by removing all edges that are not significantly different from zero. For fixed effects, multilevel software returns standard errors and p values, allowing this thresholding to be done. For the temporal networks, each edge is represented by one parameter and thus by one p value. The contemporaneous and between-subjects networks, however, are a function of two parameters that are standardized and averaged: a regression parameter for the multiple regression model of the first node and a regression parameter for the multiple regression model of the second node. As

such, for every edge, two p values are obtained. We can choose to retain edges of which at least one of the two p values is significant, termed the OR-rule, or we can choose to retain edges in which both p values are significant, termed the AND-rule (Barber, Drton, & Others, 2015).

Summary. In sum, the above described two-step estimation method proposes to estimate a multilevel model per variable, using within-person centered lagged variables as within-subject predictors and the sample means as between-subject predictors. These models can be used to obtain estimates for the temporal network and between-subjects network. In a second step, the contemporaneous networks can be estimated by estimating a second multilevel on the residuals of the first multilevel model. The *mlVAR* R package implements these methods (Epskamp, Deserno, & Bringmann, 2016). In this package, temporal coefficients can be estimated as being "unique" per subject (unique VAR models per subject), "correlated" (estimating correlations between temporal effects), "orthogonal" (assuming temporal effects are not correlated), or "fixed" (no multilevel structure on temporal effects). The contemporaneous effects can also be estimated as being "unique" (all residuals are used to obtain one GGM), "correlated" (second step multilevel model with correlated random effects), "orthogonal" (second step multilevel model with uncorrelated random effects), or "unique" (residuals are used to obtain a GGM per subject). The *mlVAR* package can also be used to plot the estimated networks, in which significance thresholding is used by default with a significance level of $\alpha = 0.05$.

6.6   Empirical Example

To provide an empirical example of the multilevel VAR methods described above, we reanalyzed the data of Mõttus et al. (2016). This data consists of two independent ESM samples, in which items tapping three of the five Five-Factor Model (McCrae & John, 1992) domains (neuroticism, extraversion, and conscientiousness) were administered as was an additional question that asked participants how much they had exercised since the preceding measurement occasion. Sample 1 consisted of 26 people providing 1,323 observations in total, and Sample 2 consisted of 62 people providing a total of 2,193 observations. Participants in Sample 1 answered questions three times per day whereas participants in Sample 2 answered questions five times per day. In both samples, the minimum time between measurements was 2 hr. For more information about the samples and the specific questions asked, we refer readers to Mõttus et al. (2016).

To obtain an easier and more interpretable example, we first only analyzed questions aimed to measure the extraversion trait and the question measuring exercise. This lead to five variables of interest: questions pertaining to feeling outgoing, energetic, adventurous, or happy and the question measuring participants' exercise habits. We analyzed the data using the *mlVAR* package. Because the number of variables was small, we estimated the model using correlated temporal and contemporaneous random effects. We ran the model separately for both samples and computed the fixed effects for the temporal, contemporaneous, and

between-subjects networks. Correlations of the edge weights indicated that all three networks showed high correspondence between the two samples (temporal network: 0.82, contemporaneous network: 0.94, between-subjects network: 0.70). Owing to the degree of replicability, we combined the two samples and estimated the model on the combined data.

Figure 6.7 shows the estimated fixed effects of the temporal, contemporaneous, and between-subjects network. In these figures, only significant edges ($\alpha = 0.05$) are shown. In the contemporaneous and between-subjects networks, an edge was retained if one of the two regressions on which the partial correlation is based was significant (the so-called OR-rule; van Borkulo et al., 2014). These results are in line with the hypothetical example shown in Figure 6.4: People who exercised were more energetic while exercising and less energetic after exercising. In the between-subjects network, no relationship between exercising and energy was found. The between-subjects network, however, showed a strong relationship between feeling adventurous and exercising: People who, on average, exercised more also felt, on average, more adventurous. This relationship was not present in the temporal network and much weaker in the contemporaneous network. Also noteworthy is that people were less outgoing after exercising. Figure 6.8 shows the standard deviation of the random effects in the temporal and contemporaneous networks. Although not many differences can be detected in the temporal network, the contemporaneous network shows strong differences: People mostly differed in their relationship between exercising and feeling energetic.

In addition to using only the extraversion and exercise items, we also ran the model on all 17 administered items in the dataset. In this instance, we used orthogonal random effects to estimate the model. Figure 6.9 shows the estimated fixed effects of the three network structures. It can be seen that indicators of the three traits tend to cluster together in all three networks. Regarding the node exercise, we found the same relationships between exercise, energetic, and adventurous (also found in the previous example) in the larger networks. Furthermore, we noted that exercising was connected to feeling angry in the between-subjects network but not in the other networks. Finally, there was a between-subjects connection between exercising and feeling self-disciplined: People who, on average, exercised more also felt, on average, more self-disciplined.

6.7 Simulation Study

In this section, we present a simulation study to assess the performance of *ml-VAR* and the above-described methods for estimating network structures on ESM data of multiple subjects. Simulation studies on the described methods for cross-sectional and $n = 1$ studies are available elsewhere (Abegaz & Wit, 2013; Foygel & Drton, 2010; see also Chapter 2). For this study, we simulated datasets of 10 variables, in which the fixed-effect temporal, contemporaneous, and between-subjects networks were simulated to be 50% sparse (i.e., containing only half the possible edges). A more detailed description of how the models were simulated can be read in the Appendix. We varied the number of subjects (50, 100, and 250) and the number of measurements per subject (25, 50, 75, and 100) and replicated
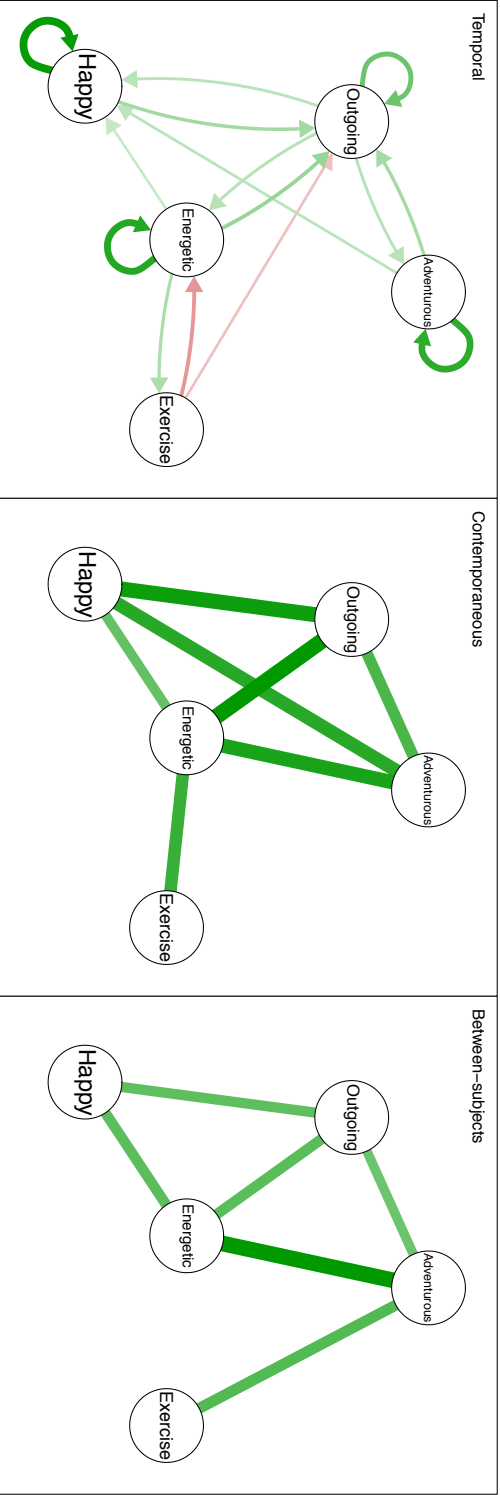
Figure 6.7: The estimated fixed effects of the three network structures obtainable in multilevel VAR. The model is based on ESM data of 88 people providing a total of 3,516 observations. Due to differences in the scale of the networks, the temporal network was drawn with a different maximum value (i.e., the value indicating the strongest edge in the network) than the contemporaneous and between-subjects networks. Edges that were not significantly different from zero were removed from the networks.
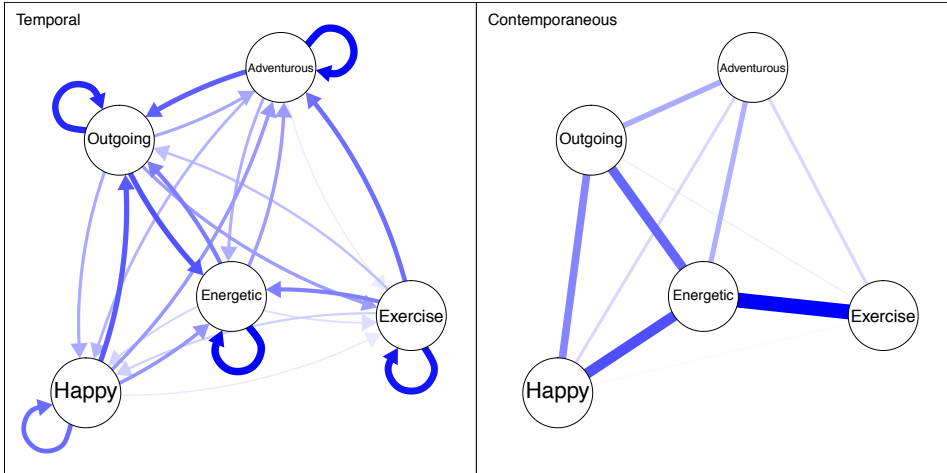
Figure 6.8: The networks showing the standard deviation of random effects in the temporal and contemporaneous networks. Due to scale differences, networks are plotted using different maximum values.

each condition 100 times. This led to a total number of 1,200 simulated datasets. In each dataset, we estimated a multilevel VAR model using orthogonal random effects.

In order to assess how well the estimated networks resemble the true networks, we computed for each dataset the correlations between true and estimated fixed temporal, contemporaneous, and between-subjects networks and the correlations between true and estimated random effects of the temporal and contemporaneous networks—because the between-subjects network does not have random effects. In addition, we assessed the performance of using significance in thresholding the network. We used the OR-rule in thresholding the fixed-effects contemporaneous and between-subjects network and removed in all the networks all edges not significant at $\alpha = 0.05$. In line with other studies on assessing how well a method retrieves the structure of a network (e.g., van Borkulo et al., 2014), we computed the *sensitivity* and *specificity*. The sensitivity (also termed *true positive rate*) is high when the method retains edges that are in the true network, and the specificity (also termed *true negative rate*) is high when the method does not retain edges that are not in the true model (i.e., models without edges that are, in reality, zero).

Figure 6.10 shows the results of the simulation study. It can be seen that performance was generally good. Fixed effects of the temporal and contemporaneous networks were well estimated (high correlations), most edges in the true network were detected (high sensitivity), and few edges were detected to be nonzero that were, in truth, zero (high specificity). Random-effect estimation was poorer but steeply increased with more measurements per subject. The between-subjects network was better estimated with more people. At low sample-sizes, the method lacked power to detect true edges (low sensitivity) but did not estimate false edges
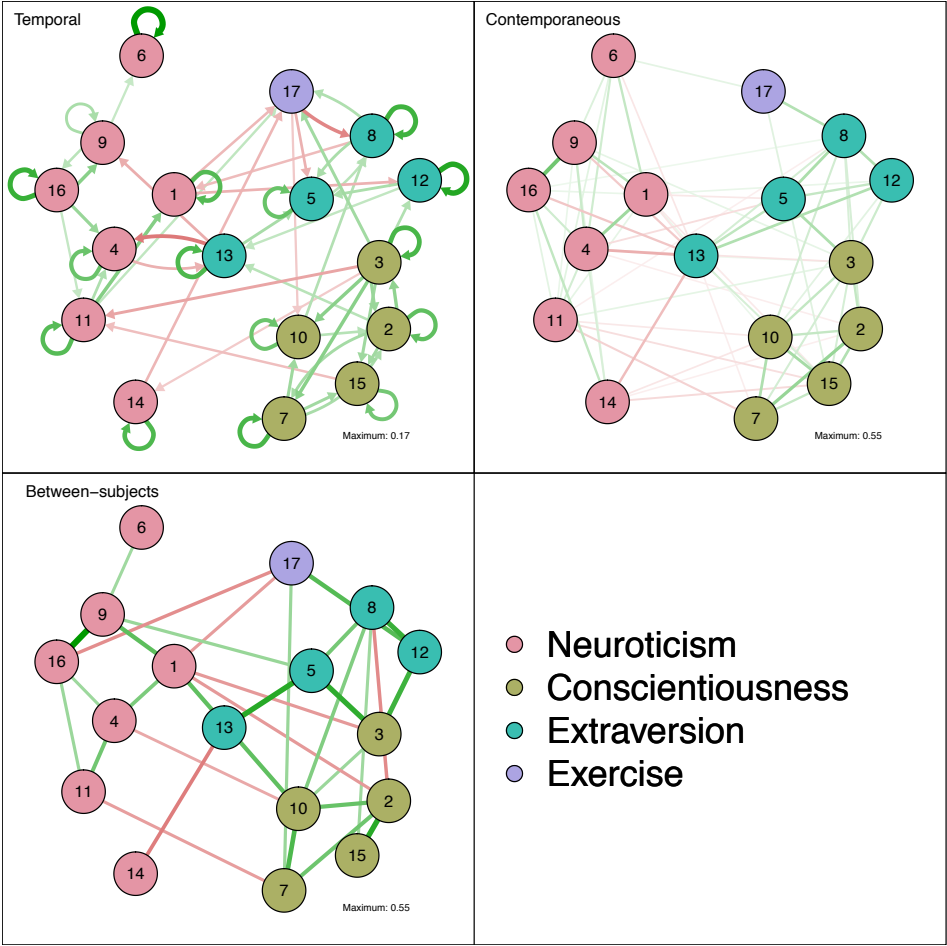
Figure 6.9: The estimated fixed effects of the three network structures based on all 17 variables administered. Only significant edges are shown. Legend: 1 = "Worried"; 2 = "Organized"; 3 = "Ambitious"; 4 = "Depressed"; 5 = "Outgoing"; 6 = "Self-Conscious"; 7 = "Self-Disciplined"; 8 = "Energetic"; 9 = "Frustrated"; 10 = "Focused"; 11 = "Guilty"; 12 = "Adventurous"; 13 = "Happy"; 14 = "Control"; 15 = "Achieved"; 16 = "Angry"; 17 = "Exercise."
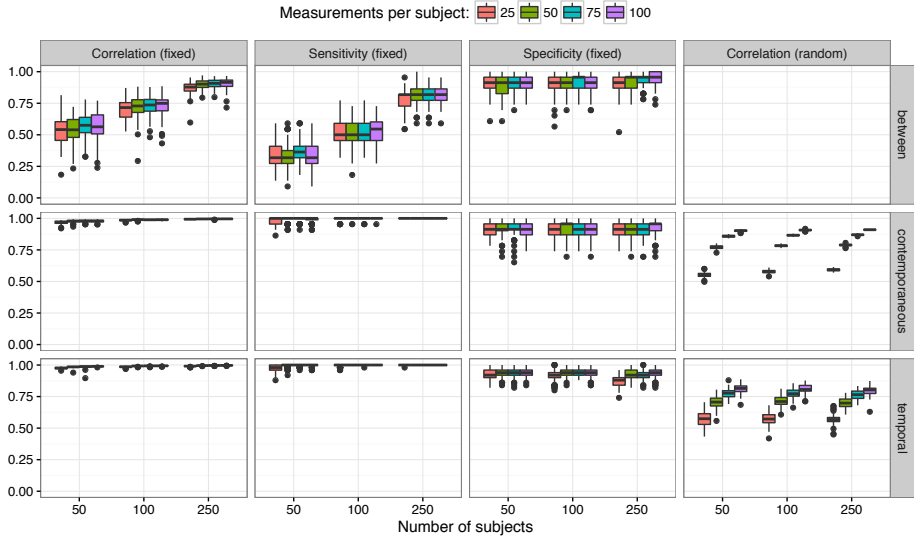
Figure 6.10: Here are the results of the simulation study. Boxplots indicate the distribution of the measures over all the 100 simulated datasets per condition. From left to right is shown: the correlation between true and estimated fixed effects, the sensitivity (i.e., the ability to detect true edges), the specificity (i.e., the ability to remove false edges), and the correlation between true and estimated random effects.

(high specificity).

6.8 Conclusion

In this chapter, we presented an overview of statistical methods that estimate network models—both cross-sectional and time-series—of multivariate Gaussian data. In our cross-sectional data analysis, we described the GGM, which takes the form of a network of partial correlation coefficients. In time-series data, we described that two network structures can be obtained: a temporal network, which is a directed network of regression coefficients between lagged and current variables, and a contemporaneous network, which is a GGM describing the relationships that remain after controlling for temporal effects. We argued that both can generate causal hypotheses. When multiple subjects were measured, the natural combination of cross-sectional and time-series data came by adding a third network structure: the between-subjects network, which is a GGM that describes relationships between the stationary means of people. We argued that this network can also show potential causal relationships but in a different way than the temporal and contemporaneous networks. We proposed a two-step, multilevel estimation procedure to estimate temporal, contemporaneous, and between-subjects networks, which we implemented in the open-source R package, *mlVAR*. We presented a

simulation study showing that *mlVAR* closely estimates the true network structure and presented an empirical example showcasing the three network structures described above.

The outlined methodology in this chapter is not the only possible methodology for obtaining network structures from multivariate Gaussian data. A detailed description of these methods was beyond the scope of this chapter, as this chapter focussed on the GGM and its natural generalizations in time-series data. In particular, much work has been done on the estimation of directed acyclic graphs (DAG; Kalisch & Bühlmann, 2007; Pearl, 2000) which aim to model causal effects. When cases can be assumed to be independent, such DAGs can be fitted in standard structural equation (SEM) modeling software (see Chapter 7). Several software packages exist that aim to find such a DAG (e.g., *pcalg*, Kalisch et al., 2012; *bnlearn*, Scutari, 2010; *BDgraph*, Mohammadi & Wit, 2015). In time-series data, one can use structural VAR (Chen et al., 2011; also termed unified SEM, Gates, Molenaar, Hillary, Ram, & Rovine, 2010) to fit contemporaneous effects in a directed network. Structural VAR can be shown to be equivalent to the VAR model discussed in this chapter, and can under strict assumptions be interpreted as a causal model. A promising estimation procedure to estimate such models over many individuals, while dealing with potential heterogeneity, is 'group iterative multiple model estimation' (GIMME; Gates & Molenaar, 2012), which is implemented in R in the *gimme* package (Lane, Gates, Molenaar, Hallquist, & Pike, 2016).

The presented methods are not without problems and have several limitations. First, multivariate estimation of the multilevel VAR model is not yet feasible for larger datasets. As such, we only focused on combining univariate models. Doing so, however, means that not all parameters are in the same model. It is important to note that univariate models do not readily provide estimates of the contemporaneous networks, which must be estimated in a second step. Second, even when multivariate estimation is possible, it is still challenging to estimate a multilevel model on the contemporaneous networks due to the requirement of positive definite matrices. Third, when more than approximately eight variables are measured, estimating the multilevel models with correlated random effects is no longer feasible in open-source, MLE software. In this case, orthogonal random effects can be used. Although the simulation study showed that the networks are still attainable when using orthogonal random effects (even though random effects were correlated in the true model), using orthogonal estimation enforces parsimony on the model that may not be plausible. Finally, even when orthogonal estimation was used, the analysis ran very slowly in models with more than 20 variables. As such, multilevel VAR analysis of high-dimensional datasets is not yet feasible. LASSO estimation as used in $n = 1$ models can also be used with multiple subjects, but it does not take individual differences into account (Abegaz & Wit, 2013). LASSO estimation methods that combine the strengths of high-dimensional network estimation in $n = 1$ models, with the ability to use information of other subjects, could be promising in this regard, but they have not yet been worked out in detail.

It should further be noted that all network structures only generate hypotheses and are in no way confirmatory of causal relationships. The analyses showcased in

this chapter are therefore exploratory and allow researchers to obtain insights into the predictive relationships present in the data—regardless of theory with respect to the data-generating model. Under the assumptions of multivariate normality, stationarity, and the Lag 1 factorization, the networks show how variables predict each other over time (temporal network), within time (contemporaneous network), and on average (between-subjects network). Furthermore, in the thresholding of edges, no correction for multiple testing was applied by default. We deliberately chose this because our aim was to present exploratory hypothesis-generating structures, and not correcting for multiple testing yields greater sensitivity. This means that care should be taken in substantively interpreting the selected edges of the networks.

One of the main innovations in this chapter comes in the substantial interpretation of between-subjects effects being potentially causal. The function of between-subjects differences in causal models has been argued to be problematic (Borsboom, Mellenbergh, & Van Heerden, 2003; Markus & Borsboom, 2013a). In order to make inferences on causal processes based on how people differ from each other, we must place very strong assumptions on the homogeneity of causal structures across individuals. In essence, we must assume that individuals are independent realizations of the same causal model (see also Hamaker, 2012). It is rarely acknowledged, however, that a similar problem holds for intraindividual data. As in the between-subjects case, the inference from a statistical association in the data to a causal model, operative at the level of the individual, is dependent on the strength of the research design and does not follow from the statistical associations themselves. In addition, if time series do not contain actual manipulations, the generalization in question can be equally problematic as in between-subjects designs.

Suppose we found a robust association between X and Y together with the temporal precedence of X (e.g., as in Granger causality) in a time-series analysis; we still would not know whether interventions on X would actually lead to changes in Y. Associations in within-subjects models can be subject to third-variable issues, such as Simpson's paradox, just as well as between-subjects models can. Correlations remain correlations, whether they come from individual differences or from time series, and rather than categorically preferring one type of data over another, it appears more sensible to let our judgment on optimal inferences depend on the substantive context.

In sum, this chapter provided methodological tools that can be used to gain insight in the potential dynamics present in psychological data. The described software packages and estimation methods present the current state-of-the-art in a field that is growing rapidly. These methods provide new ways to look at data—both literally, through the use of networks to visualize the results, and figuratively, by investigating contemporaneous and between-subjects effects in combination with temporal effects.

6.9 Appendix A: Simulating Multi-level VAR Models and Data

This algorithm generates data for P subjects on T measurement occasions of I items, using a lag-1 factorization. The fixed effect temporal, contemporaneous, and between-subjects networks all have a sparsity of 50%. This algorithm is implemented in the `mlVARsim` function. Given parameters are

- $\text{DF}_{\boldsymbol{\Theta}}$: The degrees of freedom (sampling variation) in sampling contemporaneous covariance matrices (default: $2I$)

- $S_{\boldsymbol{f_\beta}}$: The shrinkage factor of the temporal fixed effects (default: 0.9)

- $S_{\boldsymbol{\sigma_\beta^2}}$: The shrinkage factor of the temporal random effects (default: 0.9)

- $\boldsymbol{V_B}$: Vector of variances of the temporal effects (default uniformly drawn between 0.01 and 1)

1. Generate the following structures:

   - Inverse $I \times I$ variance–covariance matrices $\boldsymbol{\Theta}^{-1}$, $\boldsymbol{\Omega_\mu}^{-1}$, and $I^2 \times I^2$ variance–covariance matrix $\boldsymbol{\Omega_B}^{-1}$. All with 50% sparsity and 50% negative edges, using the methodology described by Yin and Li (2011) with a constant of 1.1 and a parameter range of 0.5 to 1. Standardize these matrices such that the diagonals of $\boldsymbol{\Theta}$ and $\boldsymbol{\Omega_\mu}$ are equal to ones and the diagonal of $\boldsymbol{\Omega_B}$ is equal to $\boldsymbol{V_B}$.
   - I length vector $\boldsymbol{f_\mu} \sim N(\boldsymbol{0}, \boldsymbol{I})$
   - I^2 length vector $\boldsymbol{f_\beta} \sim N(\boldsymbol{0}, \boldsymbol{I})$. Subsequently, set 50% lowest absolute values to zero.

2. Generate P covariance matrices $\boldsymbol{\Theta}^{(p)} \sim \text{Wishart}^{-1}(\boldsymbol{\Theta}/\text{DF}_{\boldsymbol{\Theta}}, \text{DF}_{\boldsymbol{\Theta}})$

3. Generate P parameter sets $\boldsymbol{\mu}^{(p)} \sim N(\boldsymbol{f_\mu}, \boldsymbol{\Omega_\mu})$ and $\text{Vec}(\boldsymbol{B}^{(p)}) \sim N(\boldsymbol{f_\beta}, \boldsymbol{\Omega_B})$

4. Compute eigenvalues of $\boldsymbol{B}^{(p)}$: $\lambda_1^{(1)} \ldots \lambda_I^{(P)}$

5. If $\max\left(\text{Re}(\lambda_i^{(p)})^2 + \text{Im}(\lambda_i^{(p)})^2 > 1\right)$

   a) Set $\boldsymbol{f_\beta} \leftarrow S_{\boldsymbol{f_\beta}} \boldsymbol{f_\beta}$
   b) Scale $\boldsymbol{\Omega_B}$ such that $\text{diag}(\boldsymbol{\Omega_B}) \leftarrow S_{\boldsymbol{\sigma_\beta^2}} \text{diag}(\boldsymbol{\Omega_B})$
   c) Go to 3

6. For each p, set $\boldsymbol{y}^{(p)}_{-100} = \boldsymbol{\mu}^{(p)}$

7. For each p, generate for $t = -99, -98, \ldots, T$ the scores $\boldsymbol{y}_t^{(p)}$

8. Discard all scores with $t < 1$

9. If any $|y_{ti}^{(p)}| > 100$, go to 5a

Generalized Network Psychometrics

Abstract

We introduce the network model as a formal psychometric model, conceptualizing the covariance between psychometric indicators as resulting from pairwise interactions between observable variables in a network structure. This contrasts with standard psychometric models, in which the covariance between test items arises from the influence of one or more common latent variables. Here, we present two generalizations of the network model that encompass latent variable structures, establishing network modeling as parts of the more general framework of Structural Equation Modeling (SEM). In the first generalization, we model the covariance structure of latent variables as a network. We term this framework *Latent Network Modeling* (LNM) and show that, with LNM, a unique structure of conditional independence relationships between latent variables can be obtained in an explorative manner. In the second generalization, the residual variance-covariance structure of indicators is modeled as a network. We term this generalization *Residual Network Modeling* (RNM) and show that, within this framework, identifiable models can be obtained in which local independence is structurally violated. These generalizations allow for a general modeling framework that can be used to fit, and compare, SEM models, network models, and the RNM and LNM generalizations. This methodology has been implemented in the free-to-use software package *lvnet*, which contains confirmatory model testing as well as two exploratory search algorithms: stepwise search algorithms for low-dimensional datasets and penalized maximum likelihood estimation for larger datasets. We show in simulation studies that these search algorithms performs adequately in identifying the structure of the relevant residual or latent networks. We further demonstrate the utility of these generalizations in an empirical example on a personality inventory dataset.

7.1   Introduction

Recent years have seen an emergence of network modeling in psychometrics (Borsboom, 2008; Schmittmann et al., 2013), with applications in clinical psychology (e.g., van Borkulo et al., 2015; McNally et al., 2015; Fried et al., 2015), psychiatry (e.g., Isvoranu, van Borkulo, et al., 2016; Isvoranu, Borsboom, et al., 2016), health sciences (e.g., Kossakowski et al., 2016), social psychology (e.g., Dalege et al., 2016; Cramer, Sluis, et al., 2012), and other fields (see for a review of recent literature Fried & van Borkulo, 2016). This line of literature stems from the network perspective of psychology, which conceptualizes psychological behavior as complex systems in which observed variables interact with one-another (Cramer et al., 2010). As described in previous chapters of this dissertation, network models are used to gain insight into this potentially high-dimensional interplay. In practice, network models can be used as a sparse representation of the joint distribution of observed indicators, and as such these models show great promise in psychometrics by providing a perspective that complements latent variable modeling. Network modeling highlights variance that is unique to pairs of variables, whereas latent variable modeling focuses on variance that is shared across all variables (Costantini, Epskamp, et al., 2015). As a result, network modeling and latent variable modeling can complement—rather than exclude—one-another.

In this chapter, we introduce the reader to this field of *network psychometrics* (Epskamp et al., in press) and formalize the network model for multivariate normal data, the Gaussian Graphical Model (GGM; Lauritzen, 1996), as a formal psychometric model. We contrast the GGM to the Structural Equation Model (SEM; Wright, 1921; Kaplan, 2000) and show that the GGM can be seen as another way to approach modeling covariance structures as is typically done in psychometrics. In particular, rather than modeling the covariance matrix, the GGM models the *inverse* of a covariance matrix. The GGM and SEM are thus very closely related: every GGM model and every SEM model imply a constrained covariance structure. We make use of this relationship to show that, through a reparameterization of the SEM model, the GGM model can be obtained in two different ways: first, as a network structure that relates a number of latent variables to each other, and second, as a network between residuals that remain given a fitted latent variable model. As such, the GGM can be modeled and estimated in SEM, which allows for network modeling of psychometric data to be carried out in a framework familiar to psychometricians and methodologists. In addition, this allows for one to assess the fit of a GGM, compare GGMs to one-another and compare a GGM to a SEM model.

However, the combination of GGM and SEM allows for more than fitting network models. As we will show, the strength of one framework can help overcome shortcomings of the other framework. In particular, SEM falls short in that exploratory estimation is complicated and there is a strong reliance on local independence, whereas the GGM falls short in that it assumes no latent variables. In this chapter, we introduce network models for latent covariances and for residual covariances as two distinct generalized frameworks of both the SEM and GGM. The first framework, Latent Network Modeling (LNM), formulates a network among latent variables. This framework allows researchers to exploratively estimate con-

ditional independence relationships between latent variables through model search algorithms; this estimation is difficult in the SEM framework due to the presence of equivalent models (MacCallum et al., 1993). The second framework, which we denote Residual Network Modeling (RNM), formulates a network structure on the residuals of a SEM model. With this framework, researchers can circumvent critical assumptions of both SEM and the GGM: SEM typically relies on the assumption of local independence, whereas network modeling typically relies on the assumption that the covariance structure among a set of the items is not due to latent variables at all. The RNM framework allows researchers to estimate SEM models without the assumption of local independence (all residuals can be correlated, albeit due to a constrained structure on the inverse residual covariance matrix) as well as to estimate a network structure, while taking into account the fact that the covariance between items may be partly due to latent factors.

While the powerful combination of SEM and GGM allows for confirmative testing of network structures both with and without latent variables, we recognize that few researchers have yet formulated strict confirmatory hypotheses in the relatively new field of network psychometrics. Often, researchers are more interested in exploratively searching a plausible network structure. To this end, we present two exploratory search algorithms. The first is a step-wise model search algorithm that adds and removes edges of a network as long as fit is improved, and the second uses penalized maximum likelihood estimation (Tibshirani, 1996) to estimate a sparse model. We evaluate the performance of these search methods in four simulation studies. Finally, the proposed methods have been implemented in a free-to-use R package, `lvnet`, which we illustrate in an empirical example on personality inventory items (Revelle, 2010).

7.2 Modeling Multivariate Gaussian Data

Let $\boldsymbol{y}$ be the response vector of a random subject on P items[1]. We assume $\boldsymbol{y}$ is centered and follows a multivariate Gaussian density:

$$\boldsymbol{y} \sim N_P\left(\boldsymbol{0}, \boldsymbol{\Sigma}\right),$$

In which $\boldsymbol{\Sigma}$ is a $P \times P$ variance–covariance matrix, estimated by some model-implied $\hat{\boldsymbol{\Sigma}}$. Estimating $\hat{\boldsymbol{\Sigma}}$ is often done through some form of *maximum likelihood estimation*. If we measure N independent samples of $\boldsymbol{y}$ we can formulate the $N \times P$ matrix $\boldsymbol{Y}$ containing realization $\boldsymbol{y}_i^{\top}$ as its ith row. Let $\boldsymbol{S}$ represent the sample variance–covariance matrix of $\boldsymbol{Y}$:

$$\boldsymbol{S} = \frac{1}{N-1}\boldsymbol{Y}^{\top}\boldsymbol{Y}.$$

---

[1]Throughout this chapter, vectors will be represented with lowercase boldfaced letters and matrices will be denoted by capital boldfaced letters. Roman letters will be used to denote observed variables and parameters (such as the number of nodes) and Greek letters will be used to denote latent variables and parameters that need to be estimated. The subscript i will be used to denote the realized response vector of subject i and omission of this subscript will be used to denote the response of a random subject.

In maximum likelihood estimation, we use $\boldsymbol{S}$ to compute and minimize -2 times the log-likelihood function to find $\hat{\boldsymbol{\Sigma}}$ (Lawley, 1940; Jöreskog, 1967; Jacobucci, Grimm, & McArdle, 2016):

$$\min_{\hat{\boldsymbol{\Sigma}}} \left[\log \det \left(\hat{\boldsymbol{\Sigma}} \right) + \operatorname{Trace} \left(\boldsymbol{S} \hat{\boldsymbol{\Sigma}}^{-1} \right) - \log \det \left(\hat{\boldsymbol{S}} \right) - P \right]. \tag{7.1}$$

To optimize this expression, $\hat{\boldsymbol{\Sigma}}$ should be estimated as closely as possible to $\boldsymbol{S}$ and perfect fit is obtained if $\hat{\boldsymbol{\Sigma}} = \boldsymbol{S}$. A properly identified model with the same number of parameters (K) used to form $\hat{\boldsymbol{\Sigma}}$ as there are unique elements in $\boldsymbol{S}$ ($P(P+1)/2$ parameters) will lead to $\hat{\boldsymbol{\Sigma}} = \boldsymbol{S}$ and therefore a saturated model. The goal of modeling multivariate Gaussian data is to obtain some model for $\hat{\boldsymbol{\Sigma}}$ with *positive degrees of freedom*, $K < P(P+1)/2$, in which $\hat{\boldsymbol{\Sigma}}$ resembles $\boldsymbol{S}$ closely.

Structural Equation Modeling

In Confirmatory Factor Analysis (CFA), $\boldsymbol{Y}$ is typically assumed to be a causal linear effect of a set of M centered latent variables, $\boldsymbol{\eta}$, and independent residuals or error, $\boldsymbol{\varepsilon}$:

$$\boldsymbol{y} = \boldsymbol{\Lambda}\boldsymbol{\eta} + \boldsymbol{\varepsilon}.$$

Here, $\boldsymbol{\Lambda}$ represents a $P \times M$ matrix of *factor loadings*. This model implies the following model for $\hat{\boldsymbol{\Sigma}}$:

$$\hat{\boldsymbol{\Sigma}} = \boldsymbol{\Lambda}\boldsymbol{\Psi}\boldsymbol{\Lambda}^{\top} + \boldsymbol{\Theta}, \tag{7.2}$$

in which $\boldsymbol{\Psi} = \operatorname{Var}(\boldsymbol{\eta})$ and $\boldsymbol{\Theta} = \operatorname{Var}(\boldsymbol{\varepsilon})$. In Structural Equation Modeling (SEM), $\operatorname{Var}(\boldsymbol{\eta})$ can further be modeled by adding structural linear relations between the latent variables[2]:

$$\boldsymbol{\eta} = \boldsymbol{B}\boldsymbol{\eta} + \boldsymbol{\zeta},$$

in which $\boldsymbol{\zeta}$ is a vector of residuals and $\boldsymbol{B}$ is an $M \times M$ matrix of regression coefficients. Now, $\hat{\boldsymbol{\Sigma}}$ can be more extensively modeled as:

$$\hat{\boldsymbol{\Sigma}} = \boldsymbol{\Lambda} \left(\boldsymbol{I} - \boldsymbol{B} \right)^{-1} \boldsymbol{\Psi} \left(\boldsymbol{I} - \boldsymbol{B} \right)^{-1\top} \boldsymbol{\Lambda}^{\top} + \boldsymbol{\Theta}, \tag{7.3}$$

in which now $\boldsymbol{\Psi} = \operatorname{Var}(\boldsymbol{\zeta})$. This framework can be used to model direct causal effects between observed variables by setting $\boldsymbol{\Lambda} = \boldsymbol{I}$ and $\boldsymbol{\Theta} = \boldsymbol{O}$, which is often called path analysis (Wright, 1934).

The $\boldsymbol{\Theta}$ matrix is, like $\hat{\boldsymbol{\Sigma}}$ and $\boldsymbol{S}$, a $P \times P$ matrix; if $\boldsymbol{\Theta}$ is fully estimated—contains no restricted elements—then $\boldsymbol{\Theta}$ alone constitutes a saturated model. Therefore, to make either (7.2) or (7.3) identifiable, $\boldsymbol{\Theta}$ must be strongly restricted. Typically, $\boldsymbol{\Theta}$ is set to be diagonal, a restriction often termed *local independence* (Lord, Novick, & Birnbaum, 1968; Holland & Rosenbaum, 1986) because indicators are independent of each other after conditioning on the set of latent variables. To improve fit, select off-diagonal elements of $\boldsymbol{\Theta}$ can be estimated, but systematic violations of local independence—many nonzero elements in $\boldsymbol{\Theta}$—are not possible as that will

---

[2]We make use here of the convenient all-y notation and do not distinguish between exogenous and endogenous latent variables (Hayduk, 1987).

quickly make (7.2) and (7.3) saturated or even over-identified. More precisely, **Θ** can *not* be fully-populated—some elements of **Θ** must be set to equal zero—when latent variables are used. An element of **Θ** being fixed to zero indicates that two variables are *locally independent* after conditioning on the set of latent variables. As such, local independence is a *critical assumption* in both CFA and SEM; if local independence is systematically violated, CFA and SEM will never result in correct models.

The assumption of local independence has led to critiques of the factor model and its usage in psychology; local independence appears to be frequently violated due to direct causal effects, semantic overlap, or reciprocal interactions between putative indicators of a latent variable (Borsboom, 2008; Cramer et al., 2010; Borsboom et al., 2011; Cramer, Sluis, et al., 2012; Schmittmann et al., 2013). In psychopathology research, local independence of symptoms given a person's level of a latent mental disorder has been questioned (Borsboom & Cramer, 2013). For example, three problems associated with depression are "fatigue", "concentration problems" and "rumination". It is plausible that a person who suffers from fatigue will also concentrate more poorly, as a direct result of being fatigued and regardless of his or her level of depression. Similarly, rumination might lead to poor concentration. In another example, Kossakowski et al. (2016) describe the often-used SF-36 questionnaire (Ware Jr & Sherbourne, 1992) designed to measure health related quality of life. The SF-36 contains items such as "can you walk for more than one kilometer" and "can you walk a few hundred meters". Clearly, these items can never be locally independent after conditioning on any latent trait, as one item (the ability to walk a few hundred meters) is a *prerequisite* for the other (walking more than a kilometer). In typical applications, the excessive covariance between items of this type is typically left unmodeled, and treated instead by combining items into a subscale or total score that is subsequently subjected to factor analysis; of course, however, this is tantamount to ignoring the relevant psychometric problem rather than solving it.

Given the many theoretically expected violations of local independence in psychometric applications, many elements of **Θ** in both (7.2) and (7.3) should ordinarily be freely estimated. Especially when violations of local independence are expected to be due to causal effects of partial overlap, residual correlations should not be constrained to zero; in addition, a chain of causal relationships between indicators can lead to all residuals to become correlated. Thus, even when latent factors cause much of the covariation between measured items, fitting a latent variable model that involves local independence may not fully account for correlation structure between measured items. Of course, in practice, many psychometricians are aware of this problem, which is typically addressed by freeing up correlations between residuals to improve model fit. However, this is usually done in an ad-hoc fashion, on the basis of inspection of modification indices and freeing up error covariances one by one, which is post hoc, suboptimal, and involves an uncontrolled journey through the model space. As a result, it is often difficult to impossible to tell how exactly authors arrived at their final reported models. As we will show later in this chapter, this process can be optimized and systematized using network models to connect residuals on top of a latent variable structure.
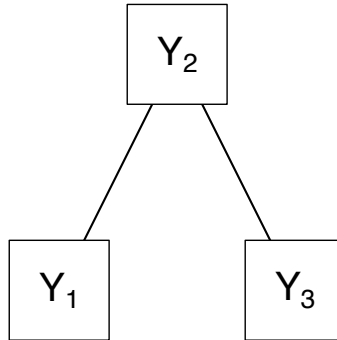
Figure 7.1: Example of a pairwise Markov Random Field model. Edges in this model indicate pairwise interactions, and are drawn using undirected edges to distinguish from (bidirectional) covariances. Rather than a model for marginal associations (such as a network indicating covariances), this is a model for *conditional* associations. The network above encodes that Y_1 and Y_3 are independent after conditioning on Y_2. Such a model allows all three variables to correlate while retaining one degree of freedom (the model only has two parameters)

Network Modeling

Recent authors have suggested that the potential presence of causal relationships between measured variables may allow the explanation of the covariance structure without the need to invoke any latent variables (Borsboom, 2008; Cramer et al., 2010; Borsboom et al., 2011; Schmittmann et al., 2013). The interactions between indicators can instead be modeled as a *network*, in which indicators are represented as nodes that are connected by edges representing *pairwise interactions*. Such interactions indicate the presence of covariances that cannot be explained by any other variable in the model and can represent—possibly reciprocal—causal relationships. Estimating a network structure on psychometric data is termed *network psychometrics* (Epskamp et al., in press). Such a network of interacting components can generate data that fit factor models well, as is commonly the case in psychology. Van Der Maas et al. (2006) showed that the positive manifold of intelligence—which is commonly explained with the general factor for intelligence, g—can emerge from a network of mutually benefiting cognitive abilities. Borsboom et al. (2011) showed that a network of psychopathological symptoms, in which disorders are modeled as clusters of symptoms, could explain comorbidity between disorders. Furthermore, Epskamp et al. (in press) showed that the Ising model for ferromagnetism (Ising, 1925), which models magnetism as a network of particles, is equivalent to multidimensional item response theory (Reckase, 2009; see Chapter 8).

In network psychometrics, psychometric data are modeled through directed or undirected networks. Directed networks are equivalent to path analysis models. For modeling undirected networks, pairwise Markov Random Fields (Lauritzen, 1996; Murphy, 2012) are used. In these models, each variable is represented by a node, and nodes are connected by a set of edges. If two nodes, y_j and y_k, are not connected by an edge, then this means they are independent after conditioning on the set of all other nodes, $\boldsymbol{y}^{-(j,k)}$. Whenever two nodes cannot be rendered independent conditional on the other nodes in the system, they are said to feature in a *pairwise interaction*, which is represented by an *undirected* edge— an edge with no arrows—to contrast such an effect from covariances typically represented in the SEM literature with bidirectional edges. Figure 7.1 represents such a network model, in which nodes y_1 and y_3 are independent after conditioning on node y_2. Such a model can readily arise from direct interactions between the nodes. For example, this conditional independence structure would emerge if y_2 is a common cause of y_1 and y_3, or if y_2 is the mediator in a causal path between y_1 and y_3. In general, it is important to note that pairwise interactions are not mere correlations; two variables may be strongly correlated but unconnected (e.g., when both are caused by another variable in the system) and they may be uncorrelated but strongly connected in the network (e.g., when they have a common effect in the system). For instance, in the present example the model does *not* indicate that y_1 and y_3 are uncorrelated, but merely indicates that any correlation between y_1 and y_3 is due to their mutual interaction with y_2; a network model in which either directly or indirectly connected paths exist between all pairs of nodes typically implies a fully populated (no zero elements) variance–covariance matrix.

In the case of multivariate Gaussian data this model is termed the Gaussian Graphical Model (GGM; Lauritzen, 1996). In the case of multivariate normality, the *partial correlation coefficient* is sufficient to test the degree of conditional independence of two variables after conditioning on all other variables; if the partial correlation coefficient is zero, there is conditional independence and hence no edge in the network. As such, partial correlation coefficients can directly be used in the network as *edge weights*; the strength of connection between two nodes[3]. Such a network is typically encoded in a symmetrical and real valued $p \times p$ *weight matrix*, $\boldsymbol{\Omega}$, in which element ω_{jk} represents the edge weight between node j and node k:

$$\mathrm{Cor}\left(y_j, y_k \mid \boldsymbol{y}^{-(j,k)}\right) = \omega_{jk} = \omega_{kj}.$$

The partial correlation coefficients can be directly obtained from the inverse of variance–covariance matrix $\hat{\boldsymbol{\Sigma}}$, also termed the *precision matrix* $\hat{\boldsymbol{K}}$ (Lauritzen, 1996):

$$\mathrm{Cor}\left(y_j, y_k \mid \boldsymbol{y}^{-(j,k)}\right) = -\frac{\kappa_{jk}}{\sqrt{\kappa_{kk}}\sqrt{\kappa_{jj}}}.$$

Thus, element κ_{jk} of the precision matrix is proportional to to the partial correlation coefficient of variables y_j and y_k after conditioning on all other variables. Since this process simply involves standardizing the precision matrix, we propose

---

[3]A saturated GGM is also called a partial correlation network because it contains the sample partial correlation coefficients as edge weights.

the following model[4]:

$$\hat{\mathbf{\Sigma}} = \hat{\mathbf{K}}^{-1} = \mathbf{\Delta} \left(\mathbf{I} - \mathbf{\Omega}\right)^{-1} \mathbf{\Delta}, \tag{7.4}$$

in which $\mathbf{\Delta}$ is a diagonal matrix with $\delta_{jj} = \kappa_{jj}^{-\frac{1}{2}}$ and $\mathbf{\Omega}$ has zeroes on the diagonal. This model allows for confirmative testing of the GGM structures on psychometric data. Furthermore, the model can be compared to a saturated model (fully populated off-diagonal values of $\mathbf{\Omega}$) and the independence model ($\mathbf{\Omega} = \mathbf{O}$), allowing one to obtain χ^2 fit statistics as well as fit indices such as the RMSEA (Browne & Cudeck, 1992) and CFI (Bentler, 1990). Such methods of assessing model fit have not yet been used in network psychometrics.

Similar to CFA and SEM, the GGM relies on a critical assumption; namely, that covariances between observed variables are *not* caused by any latent or unobserved variable. If we estimate a GGM in a case where, in fact, a latent factor model was the true data generating structure, then generally we would expect the GGM to be saturated—i.e., there would be no missing edges in the GGM (Chandrasekaran, Parrilo, & Willsky, 2012). A missing edge in the GGM indicates the presence of conditional independence between two indicators given all other indicators; we do not expect indicators to become independent given subsets of other indicators (see also Ellis & Junker, 1997; Holland & Rosenbaum, 1986). Again, this critical assumption might not be plausible. While variables such as "Am indifferent to the feelings of others" and "Inquire about others' well-being" quite probably interact with each other, it might be far-fetched to assume that no unobserved variable, such as a personality trait, in part also causes some of the variance in responses on these items.

7.3   Generalizing Factor Analysis and Network Modeling

We propose two generalizations of both SEM and the GGM that both allow the modeling of network structures in SEM. In the first generalization, we adopt the CFA[5] decomposition in (7.2) and model the variance–covariance matrix of latent variables as a GGM:

$$\mathbf{\Psi} = \mathbf{\Delta_{\Psi}} \left(\mathbf{I} - \mathbf{\Omega_{\Psi}}\right)^{-1} \mathbf{\Delta_{\Psi}}.$$

This framework can be seen as modeling conditional independencies between latent variables not by directed effects (as in SEM) but as an undirected network. As such, we term this framework *latent network modeling* (LNM).

In the second generalization, we adopt the SEM decomposition of the variance–covariance matrix in (7.3) and allow the residual variance–covariance matrix $\mathbf{\Theta}$ to be modeled as a GGM:

$$\mathbf{\Theta} = \mathbf{\Delta_{\Theta}} \left(\mathbf{I} - \mathbf{\Omega_{\Theta}}\right)^{-1} \mathbf{\Delta_{\Theta}}.$$

---

[4]To our knowledge, the GGM has not yet been framed in this form. We chose this form because it allows for clear modeling and interpretation of the network parameters.

[5]We use the CFA framework instead of the SEM framework here as the main application of this framework is in exploratively estimating relationships between latent variables.

A. Structural Equation Modeling   B. Network Modeling

C. Latent Network Modeling   D. Residual Network Modeling
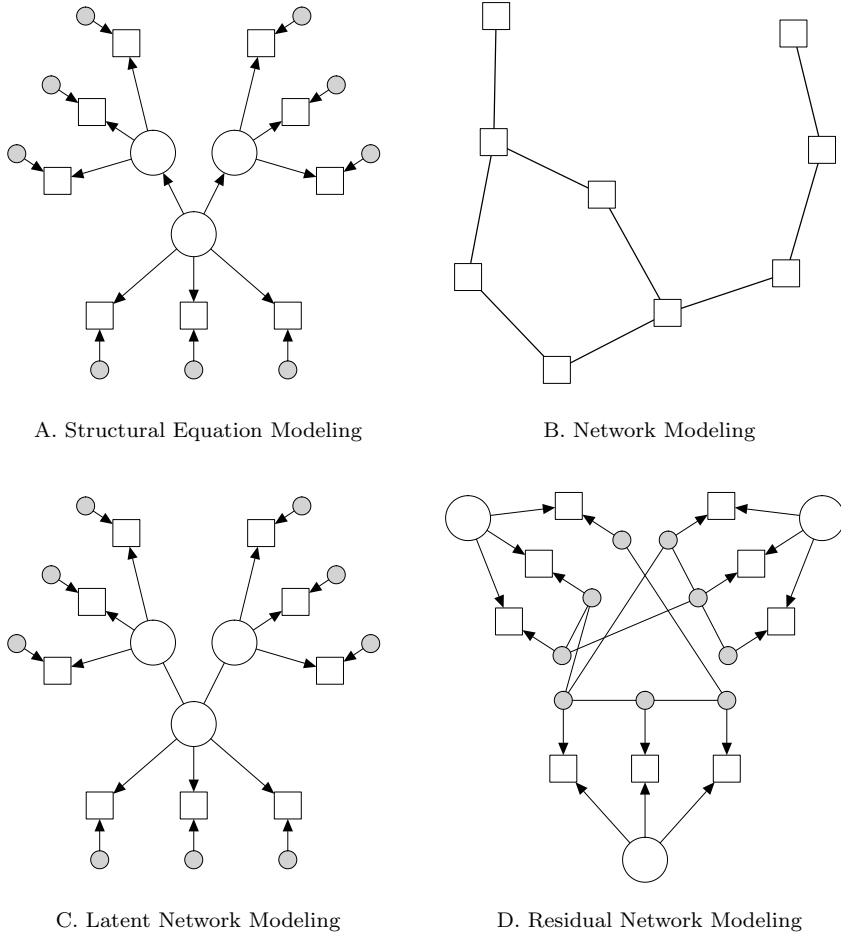
Figure 7.2: Examples of possible models under four different modeling frameworks. Circular nodes indicate latent variables, square nodes indicate manifest variables and gray nodes indicate residuals. Directed edges indicate factor loadings or regression parameters and undirected edges indicate pairwise interactions. Note that such undirected edges do *not* indicate covariances, which are typically denoted with bidirectional edges. Replacing covariances with interactions is where the network models differ from typical SEM.

Because this framework conceptualizes associations between residuals as pairwise *interactions*, rather than correlations, we term this framework *Residual Network Modeling* (RNM). Using this framework allows—as will be described below—for a powerful way of fitting a confirmatory factor structure even though local independence is systematically violated and all residuals are correlated.

Figure 7.2 shows four different examples of possible models that are attainable

under the SEM, LNM and RNM frameworks. Panel A shows a typical SEM model in which one latent variable functions as a common cause of two others. Panel B shows a network model which can be estimated using both the RNM and the LNM frameworks. Panel C shows a completely equivalent LNM model to the SEM model of Panel A in which the direction of effect between latent variables is not modeled. Finally, panel D shows a model in which three exogenous latent variables underlie a set of indicators of which the residuals form a network. The remainder of this section will describe RNM and LNM in more detail and will outline the class of situations in which using these models is advantageous over CFA or SEM.

Latent Network Modeling

The LNM framework models the latent variance–covariance matrix of a CFA model as a GGM:

$$\hat{\boldsymbol{\Sigma}} = \boldsymbol{\Lambda}\boldsymbol{\Delta}_{\boldsymbol{\Psi}} \left(\boldsymbol{I} - \boldsymbol{\Omega}_{\boldsymbol{\Psi}}\right)^{-1} \boldsymbol{\Delta}_{\boldsymbol{\Psi}}\boldsymbol{\Lambda}^{\top} + \boldsymbol{\Theta}. \tag{7.5}$$

This allows researchers to model conditional independence relationships between latent variables without making the implicit assumptions of directionality or acyclicness. In SEM, $\boldsymbol{B}$ is typically modeled as a directed acyclic graph (DAG), meaning that elements of $\boldsymbol{B}$ can be represented by directed edges, and following along the path of these edges it is not possible to return to any node (latent variable). The edges in such a DAG can be interpreted as causal, and in general they imply a specific set of conditional independence relationships between the nodes (Pearl, 2000).

While modeling conditional independence relationships between latent variables as a DAG is a powerful tool for testing strictly confirmatory hypotheses, it is less useful for more exploratory estimation. Though there have been recent advances in exploratory estimation of DAGs within an SEM framework (e.g., Gates & Molenaar, 2012; Rosa, Friston, & Penny, 2012), many equivalent DAGs can imply the same conditional independence relationships, and thus fit the data equally well even though their causal interpretation can be strikingly different (MacCallum et al., 1993). Furthermore, the assumption that the generating model is acyclic—which, in practice, often is made on purely pragmatic grounds to identify a model—is problematic in that much psychological behavior can be assumed to have at least some cyclic and complex behavior and feedback (Schmittmann et al., 2013). Thus, the true conditional independence relationships in a dataset can lead to many equivalent compositions of $\boldsymbol{B}$, and possibly none of them are the true model.

In psychometrics and SEM the GGM representation has not been very prominent, even though it has some manifest benefits over the attempt to identify DAGs directly. For example, by modeling conditional independence relationships between latent variables as a GGM, many relationships can be modeled in a simpler way as compared to a DAG. In addition, in the GGM each set of conditional independence relations only corresponds to one model: there are no equivalent GGMs with the same nodes. Figure 7.3 shows a comparison of several conditional independence relations that can be modeled equivalently or not by using a GGM
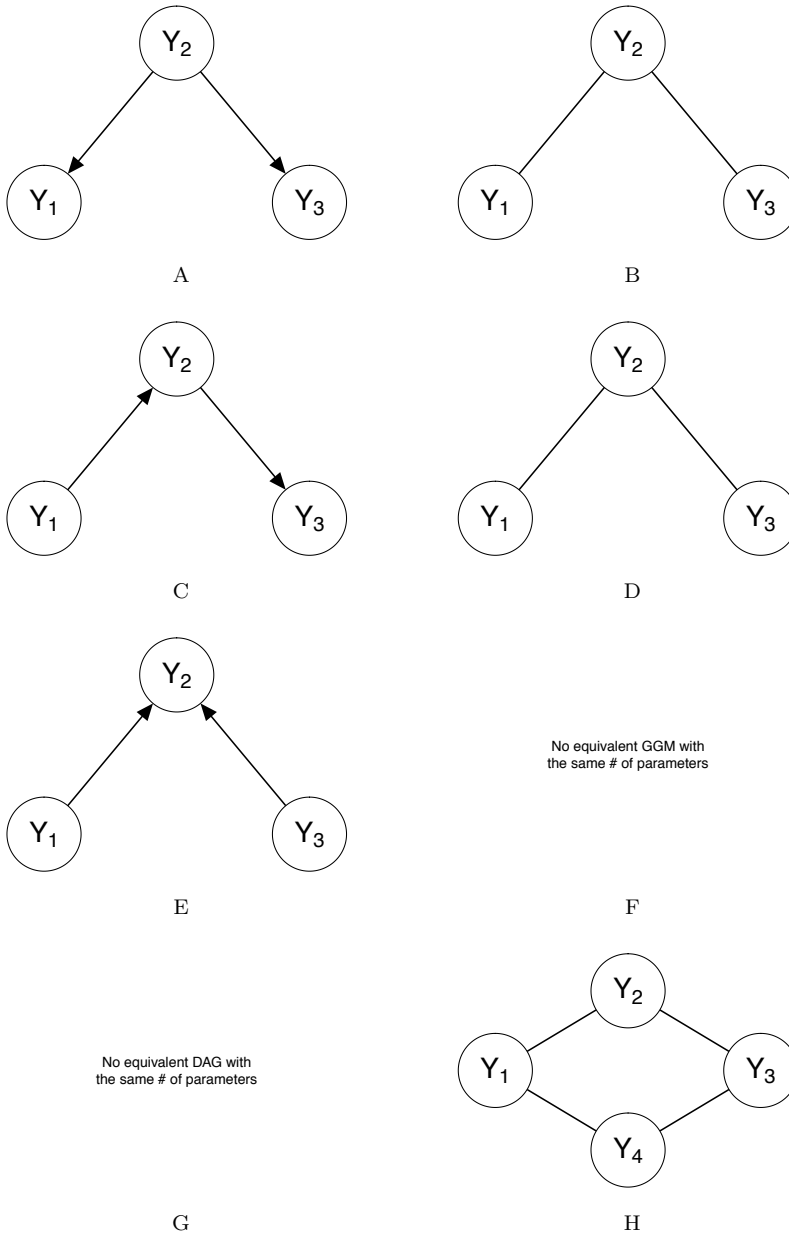
Figure 7.3: Equivalent models between directed acyclic graphs (DAG; left) and Gaussian graphical models (GGM; right). Each row of graphs show two models that are equivalent.

or by using a DAG. Panel A and Panel C show two DAGs that represent the same conditional independence relations, $y_1 \perp\!\!\!\perp y_3 \mid y_2$, which can both be represented by the same GGM shown in Panel B and Panel D. There are some conditional independence relations that a GGM cannot represent in the same number of parameters as a DAG; Panel E shows a collider structure that cannot be exactly represented by a GGM (the best fitting GGM would feature three edges instead of two). On the other hand, there are also conditional independence relationships that a GGM can represent and a DAG cannot; the cycle of Panel H cannot be represented by a DAG. Further equivalences and differences between GGMs and DAGs are beyond the scope of this chapter, but haven been well described in the literature (e.g., chapter 3 of Lauritzen, 1996; Koller & Friedman, 2009; Kolaczyk, 2009). In sum, the GGM offers a natural middle ground between zero-order correlations and DAGs: every set of zero-order correlations implies exactly one GGM, and every DAG implies exactly one GGM. In a sense, the road from correlations to DAGs (including hierarchical factor models) thus always must pass through the realm of GGMs, which acts as a bridge between the correlational and causal worlds.

Because there are no equivalent undirected models possible, LNM offers a powerful tool for exploratory estimation of relationships between latent variables. For example, suppose one encounters data generated by the SEM model in Figure 7.2, Panel A. Without prior theory on the relations between latent variables, exploratory estimation on this dataset would lead to three completely equivalent models: the one shown in Figure 7.2, Panel C and two models in which the common cause instead is the middle node in a causal chain. As the number of latent variables increases, the potential number of equivalent models that encode the same conditional independence relationships grows without bound. The LNM model in Panel C of Figure 7.2 portrays the same conditional independence relationship as the SEM model in Panel A of Figure 7.2, while having no equivalent model. Exploratory estimation could easily find this model, and portrays the retrieved relationship in a clear and unambiguous way.

A final benefit of using LNM models is that they allow network analysts to construct a network while taking measurement error into account. So far, networks have been constructed based on single indicators only and no attempt has been made to remediate measurement error. By forming a network on graspable small concepts measured by a few indicators, the LNM framework can be used to control for measurement error.

Residual Network Modeling

In the RNM framework the residual structure of SEM is modeled as a GGM:

$$\hat{\boldsymbol{\Sigma}} = \boldsymbol{\Lambda}\left(\boldsymbol{I} - \boldsymbol{B}\right)^{-1}\boldsymbol{\Psi}\left(\boldsymbol{I} - \boldsymbol{B}\right)^{-1\top}\boldsymbol{\Lambda}^{\top} + \boldsymbol{\Delta_\Theta}\left(\boldsymbol{I} - \boldsymbol{\Omega_\Theta}\right)^{-1}\boldsymbol{\Delta_\Theta}. \tag{7.6}$$

This modeling framework conceptualizes latent variable and network modeling as two sides of the same coin, and offers immediate benefits to both. In latent variable modeling, RNM allows for the estimation of a factor structure (possibly including structural relationships between the latent variables), while having no uncorrelated

errors and thus no local independence. The error-correlations, however, are still highly structured due to the residual network structure. This can be seen as a compromise between the ideas of network analysis and factor modeling; while we agree that local independence is plausibly violated in many psychometric tests, we think the assumption of no underlying latent traits and therefore a sparse GGM may often be too strict. For network modeling, RNM allows a researcher to estimate a sparse network structure while taking into account that some of the covariation between items was caused by a set of latent variables. Not taking this into account would lead to a saturated model (Chandrasekaran et al., 2012), whereas the residual network structure can be sparse.

To avoid confusion between residual correlations, we will denote edges in $\mathbf{\Omega_{\Theta}}$ *residual interactions*. Residual interactions can be understood as pairwise linear effects, possibly due to some causal influence or partial overlap between indicators that is left after controlling for the latent structure. Consider again the indicators for agreeableness "Am indifferent to the feelings of others" and "Inquire about others' well-being". It seems clear that we would not expect these indicators to be locally independent after conditioning on agreeableness; being indifferent to the feelings of others will cause one to not inquire about other's well-being. Thus, we could expect these indicators to feature a residual interaction; some degree of correlation between these indicators is expected to remain, even after conditioning on the latent variable and all other indicators in the model.

The RNM framework in particular offers a new way of improving the fit of confirmatory factor models. In contrast to increasingly popular methods such as exploratory SEM (ESEM; Marsh, Morin, Parker, & Kaur, 2014) or LASSO regularized SEM models (Jacobucci et al., 2016), the RNM framework improves the fit by adding residual interactions rather than allowing for more cross-loadings. The factor structure is kept exactly intact as specified in the confirmatory model. Importantly, therefore, the interpretation of the latent factor does not change. This can be highly valuable in the presence of a strong theory on the latent variables structure underlying a dataset even in the presence of violations of local independence.

7.4 Exploratory Network Estimation

Both the LNM and RNM modeling frameworks allow for confirmative testing of network structures. Confirmatory estimation is straightforward and similar to estimating SEM models, with the exception that instead of modeling $\mathbf{\Psi}$ or $\mathbf{\Theta}$ now the latent network $\mathbf{\Omega_{\Psi}}$ or $\mathbf{\Omega_{\Theta}}$ is modeled. Furthermore, both modeling frameworks allow for the confirmatory fit of a network model. In LNM, a confirmatory network structure can be tested by setting $\mathbf{\Lambda} = \boldsymbol{I}$ and $\mathbf{\Theta} = \boldsymbol{O}$; in RNM, a confirmatory network model can be tested by omitting any latent variables. We have developed the R package `lvnet`[6], which utilizes `OpenMx` (Neale et al., 2016) for confirmative testing of RNM and LNM models (as well as a combination of the two). The `lvnet` function can be used for this purpose by specifying the fixed and the free elements

---

[6]github.com/sachaepskamp/lvnet

of model matrices. The package returns model fit indices (e.g., the RMSEA, CFI and χ^2 value), parameter estimates, and allows for model comparison tests.

Often the network structure, either at the residual or the latent level, is unknown and needs to be estimated. To this end, the package includes two exploratory search algorithms described below: step-wise model search and penalized maximum likelihood estimation. For both model frameworks and both search algorithms, we present simulation studies to investigate the performance of these procedures. As is typical in simulation studies investigating the performance of network estimation techniques, we investigated the *sensitivity* and *specificity* (van Borkulo et al., 2014). These measures investigate the estimated edges versus the edges in the true model, with a 'positive' indicating an estimated edge and a 'negative' indicating an edge that is estimated to be zero. Sensitivity, also termed the true positive rate, gives the ratio of the number of true edges that were detected in the estimation versus the total number of edges in the true model:

$$\text{sensitivity} = \frac{\#\text{ true positives}}{\#\text{ true positives} + \#\text{ of false negatives}}$$

Specificity, also termed the true negative rate, gives the ratio of true missing edges detected in the estimation versus the total number of absent edges in the true model:

$$\text{specificity} = \frac{\#\text{ true negatives}}{\#\text{ true negatives} + \#\text{ false positives}}$$

The specificity can be seen as a function of the number of false positives: a high specificity indicates that there were not many edges detected to be nonzero that are zero in the true model. To favor degrees of freedom, model sparsity and interpretability, specificity should be high all-around—estimation techniques should not result in many false positives—whereas sensitivity should increase as a function of the sample size.

Simulating Gaussian Graphical models

In all simulation studies reported here, networks were constructed in the same way as done by Yin and Li (2011) in order to obtain a positive definite inverse-covariance matrix $\boldsymbol{K}$. First, a network structure was generated without weights. Next, weights were drawn randomly from a uniform distribution between 0.5 and 1, and made negative with 50% probability. The diagonal elements of $\boldsymbol{K}$ were then set to 1.5 times the sum of all absolute values in the corresponding row, or 1 if this sum was zero. Next, all values in each row were divided by the diagonal value, ensuring that the diagonal values become 1. Finally, the matrix was made symmetric by averaging the lower and upper triangular elements. In the chain graphs used in the following simulations, this algorithm created networks in which the non-zero partial correlations had a mean of 0.33 and a standard deviation of 0.04.

Stepwise Model Search

In exploratory search, we are interested in recovering the network structure of either $\mathbf{\Omega_\Psi}$ in LNM or $\mathbf{\Omega_\Theta}$ in RNM. This can be done through a step-wise model search, either based on χ^2 difference tests (Algorithm 1) or on minimization of some information criterion (Algorithm 2) such as the Akaike information criterion (AIC), Bayesian information criterion (BIC) or the extended Bayesian information criterion (EBIC; Chen & Chen, 2008) which is now often used in network estimation (van Borkulo et al., 2014; Foygel & Drton, 2010). In LNM, removing edges from $\mathbf{\Omega_\Psi}$ cannot improve the fit beyond that of an already fitting CFA model. Hence, model search for $\mathbf{\Omega_\Psi}$ should start at a fully populated initial setup for $\mathbf{\Omega_\Psi}$. In RNM, on the other hand, a densely populated $\mathbf{\Omega_\Theta}$ would lead to an over-identified model, and hence the step-wise model search should start at an empty network $\mathbf{\Omega_\Theta} = \mathbf{O}$. The function `lvnetSearch` in the `lvnet` package can be used for both search algorithms.

Algorithm 1 Stepwise network estimation by χ^2 difference testing.

Start with initial setup for $\mathbf{\Omega}$
repeat
  for all Unique elements of $\mathbf{\Omega}$ **do**
    Remove edge if present or add edge if absent
    Fit model with changed edge
  end for
  if Adding an edge significantly improves fit ($\alpha = 0.05$) **then**
    Add edge that improves fit the most
  else if Removing an edge does not significantly worsen fit ($\alpha = 0.05$) **then**
    Remove edge that worsens fit the least
  end if
until No added edge significantly improves fit and removing any edge significantly worsens fit

---

Algorithm 2 Stepwise network estimation by AIC, BIC or EBIC optimization.

Start with initial setup for $\mathbf{\Omega}$
repeat
  for all Unique elements of $\mathbf{\Omega}$ **do**
    Remove edge if present or add edge if absent
    Fit model with changed edge
  end for
  if Any changed edge improved AIC, BIC or EBIC **then**
    Change edge that improved AIC, BIC or EBIC the most
  end if
until No changed edge improves AIC, BIC or EBIC

Figure 7.4: Model used in simulation study 1: step-wise model search in latent network modeling. Four latent variables were each measured by three items. Latent variables covary due to the structure of a latent Gaussian graphical model in which edges indicate partial correlation coefficients. This model has the form of a chain graph, which cannot be represented in a structural equation model. Factor loadings, residual variances and latent variances were set to 1 and the latent partial correlations had an average of 0.33 with a standard deviation of 0.04.

Simulation Study 1: Latent Network Modeling

We performed a simulation study to assess the performance of the above mentioned step-wise search algorithms in LNM models. Figure 7.4 shows the LNM model under which we simulated data. In this model, four latent factors with three indicators each were connected in a latent network. The latent network was a chain network, leading all latent variables to be correlated according to a structure that cannot be represented in SEM. Factor loadings and residual variances were set to 1, and the network weights were simulated as described in the section "Simulating Gaussian Graphical models". The simulation study followed a 5×4 design: the sample size was varied between 50, 100, 250, 500 and 1 000 to represent typical sample sizes in psychological research, and the stepwise evaluation criterion was either χ^2 difference testing, AIC, BIC or EBIC (using a tuning parameter of 0.5). Each condition was simulated 1 000 times, resulting in 20 000 total simulated datasets.

Figure 7.5 shows the results of the simulation study. Data is represented in standard boxplots (McGill, Tukey, & Larsen, 1978): the box shows the 25th, 50th (median) 75th quantiles, the whiskers range from the largest values in 1.5 times the inter-quantile range (75th - 25th quantile) and points indicate outliers outside that range. In each condition, we investigated the sensitivity and specificity. The top panel shows that sensitivity improves with sample size, with AIC performing best and EBIC worst. From sample sizes of 500 and higher all estimation criterion

Figure 7.5: Simulation results of simulation study 1: step-wise model search in latent network modeling. Each condition was replicated 1 000 times, leading to 20 000 total simulated datasets. High sensitivity indicates that the method is able to detect edges in the true model, and high specificity indicates that the method does not detect edges that are zero in the true model.
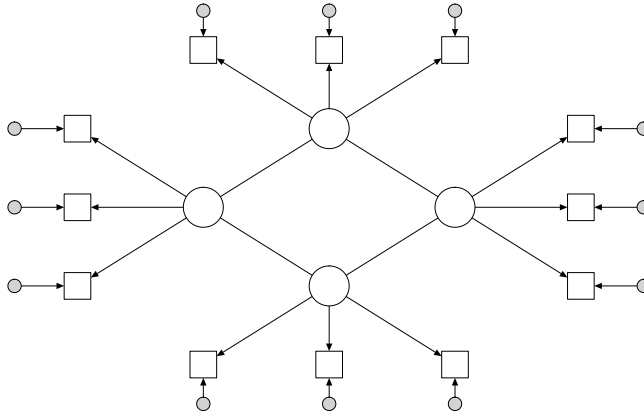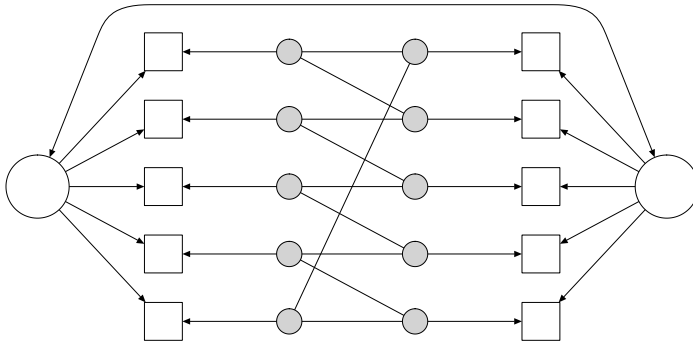
Figure 7.6: Model used in simulation study 2: step-wise model search in residual network modeling. Two latent variables were each measured by five items; a Gaussian graphical model, in which edges indicate partial correlation coefficients, leads to all residuals to be correlated due to a chain graph between residuals, which cannot be represented in a structural equation model. Factor loadings, residual variances and latent variances were set to 1, the factor covariance was set to 0.25 and the latent partial correlations had an average of 0.33 with a standard deviation of 0.04.

performed well in retrieving the edges. The bottom panel shows that specificity is generally very high, with EBIC performing best and AIC worst. These results indicate that the step-wise procedure is conservative and prefers simpler models to more complex models; missing edges are adequately detected but present edges in the true model might go unnoticed except in larger samples. With sample sizes over 500, all four estimation methods show both a high sensitivity and specificity.

Simulation Study 2: Residual Network Modeling

We conducted a second simulation study to assess the performance of step-wise model selection in RNM models. Figure 7.7 shows the model under which data were simulated: two latent variables with 5 indicators each. The residual network was constructed to be a chain graph linking a residual of an indicator of one latent variable to two indicators of the other latent variable. This structure cannot be represented by a DAG and causes all residuals to be connected, so that $\boldsymbol{\Theta}$ is fully populated. Factor loadings and residual variances were set to 1, the factor covariance was set to 0.25, and the network weights were simulated as described in the section "Simulating Gaussian Graphical models".

The simulation study followed a 5×4 design; sample size was again varied between 50, 100, 250, 500 and 1 000, and models were estimated using either χ^2 significance testing, AIC, BIC or EBIC. Factor loadings and factor variances were set to 1 and the factor correlation was set to 0.25. The weights in $\boldsymbol{\Omega_\Theta}$ were chosen as described in the section "Simulating Gaussian Graphical models". Each condition was simulated 1, 000 times, leading to 20 000 total datasets.
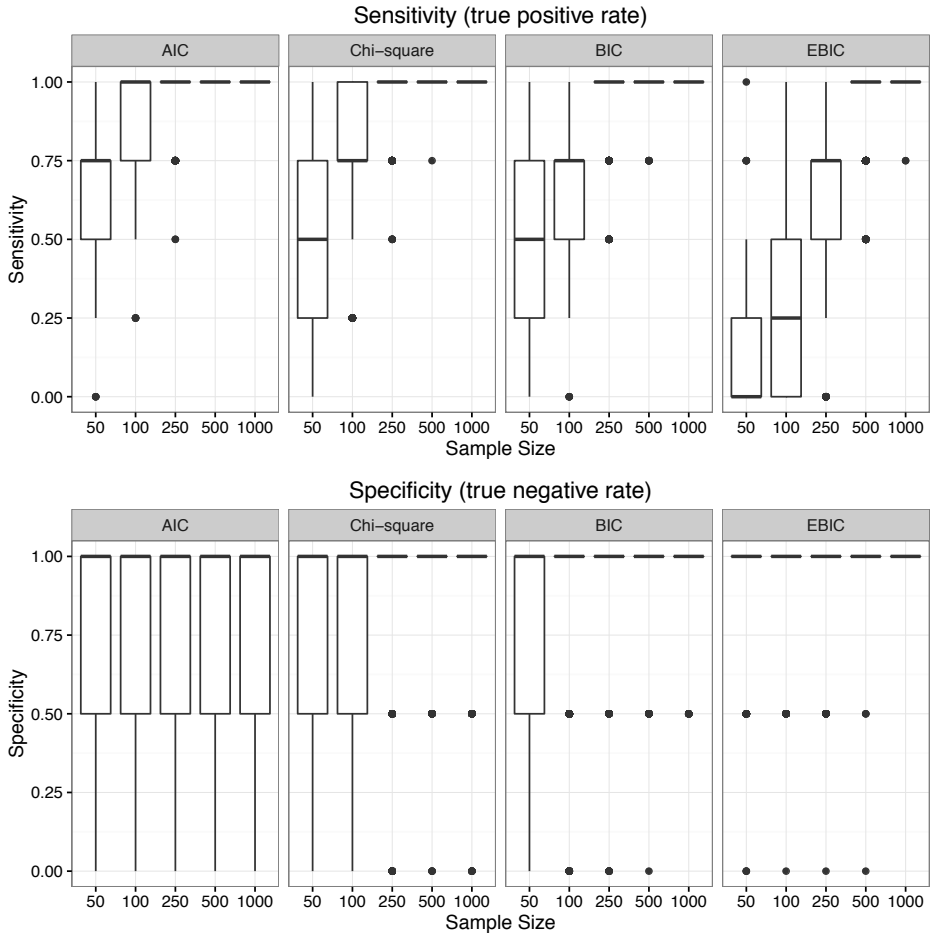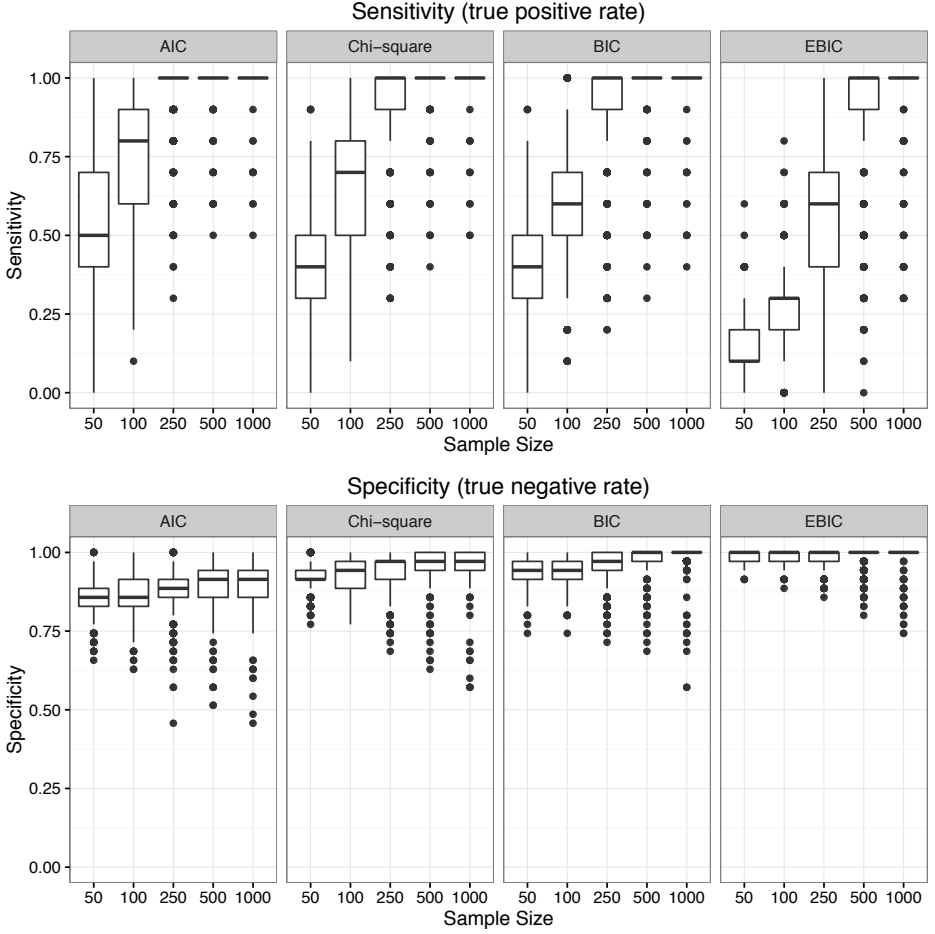
Figure 7.7: Simulation results of simulation study 2: step-wise model search in residual network modeling. Each condition was replicated 1 000 times, leading to 20 000 total simulated datasets. High sensitivity indicates that the method is able to detect edges in the true model, and high specificity indicates that the method does not detect edges that are zero in the true model.

Figure 7.7 shows the results of the simulation study. The top panel shows that sensitivity increases with sample size and performs best when using AIC as the criterion. BIC performed comparably in sensitivity to χ^2 testing and EBIC performed the worst. The bottom panel shows that specificity was very high for all sample sizes and all criteria, with EBIC performing best and AIC worst. These results indicate that the number of false positives is very low and that the method is on average well capable of discovering true edges for sample size larger than 250. In sum, all four criteria perform well with EBIC erring on the side of caution and AIC erring on the side of discovery.

LASSO Regularization

While the step-wise model selection algorithms perform well in retrieving the correct network structure, they are very slow when the number of nodes in the network increases (e.g., more than 10 nodes). This is particularly important in the context of RNM, in which the number of indicators can be larger than 10 even in small models. A popular method for fast estimation of high-dimensional network structures is by applying the least absolute shrinkage and selection operator (LASSO; Tibshirani, 1996). LASSO regularization has also recently been introduced in the SEM literature (Jacobucci et al., 2016) as a method for obtaining sparser structures of $\mathbf{\Lambda}$ and $\mathbf{B}$. In the LASSO, instead of optimizing the likelihood function as described in (7.1), a penalized likelihood is optimized (Jacobucci et al., 2016):

$$\min_{\hat{\mathbf{\Sigma}}} \left[\log \det \left(\hat{\mathbf{\Sigma}} \right) + \text{Trace} \left(\mathbf{S} \hat{\mathbf{\Sigma}}^{-1} \right) - \log \det \left(\hat{\mathbf{S}} \right) - P + \nu \text{Penalty} \right], \qquad (7.7)$$

in which ν denotes a tuning parameter controlling the level of penalization. The penalty here is taken to be the sum of absolute parameters:

$$\text{Penalty} = \sum_{<i,j>} |\omega_{ij}|,$$

in which ω_{ij} denotes an element from either $\mathbf{\Omega_\Psi}$ or $\mathbf{\Omega_\Theta}$. Other penalty functions may be used as well—such as summing the squares of parameter estimates (ridge regression; Hoerl & Kennard, 1970) or combining both absolute and squared values (elastic net; Zou & Hastie, 2005)—but these are not currently implemented in `lvnet`. The benefit of the LASSO is that it returns models that perform better in cross-validation. In addition, the LASSO yields sparse models in which many relationships are estimated to be zero.

Algorithm 3 LASSO estimation for exploratory network search.

    for all Sequence of tuning parameters $\nu_1, \nu_2, \ldots$ **do**
        Estimate LASSO regularized model using given tuning parameter
        Count the number of parameters for which the absolute estimate is larger
        than ϵ
        Determine information criterion AIC or BIC given fit and number of param-
        eters
    end for
    Select model with best AIC, BIC or EBIC
    Refit this model without LASSO in which absolute parameters smaller than ϵ
    are fixed to zero

The `lvnet` function allows for LASSO regularization for a given model matrix ($\mathbf{\Omega_\Theta}$, $\mathbf{\Omega_\Psi}$, $\mathbf{\Theta}$, $\mathbf{\Psi}$, $\mathbf{\Lambda}$ or $\mathbf{B}$) and a given value for the tuning parameter ν. The optimizer used in `lvnet` does not return exact zeroes. To circumvent this issue, any absolute parameter below some small value ϵ (by default $\epsilon = 0.0001$) is treated as zero in counting the number of parameters and degrees of freedom (Zou, Hastie, Tibshirani, et al., 2007). The `lvnetLasso` function implements the search algorithm described in Algorithm 3 to automatically choose an appropriate tuning parameter, use that for model selection and rerun the model to obtain a comparable fit to non-regularized models. In this algorithm, a sequence of tuning parameters is tested, which is set by default to a logarithmically spaced sequence of 20 values between 0.01 and 1.

Simulation Study 3: Latent Network Modeling

We studied the performance of LASSO penalization in estimating the latent network structure in a similar simulation study to the study of the step-wise procedure described above. Data were simulated under a similar model to the one shown in Figure 7.4, except that now 8 latent variables were used leading to a total of 24 observed variables. All parameter values were the same as in simulation study 1. The simulation followed a 5×3 design. Sample size was varied between 100, 250, 500, 1 000 and 2 500, and for each sample size 1 000 datasets were simulated leading to a total of 5 000 generated datasets. On these datasets the best model was selected using either AIC, BIC or EBIC, leading to 15 000 total replications. In each replication, sensitivity and specificity were computed. Figure 7.8 shows that AIC had a relatively poor specificity all-around, but a high sensitivity. EBIC performed well with sample sizes of 500 and higher.

Simulation Study 4: Residual Network Modeling

To assess the performance of LASSO in estimating the residual network structure we simulated data as in Figure 7.6, except that in this case four latent variables were used, each with 5 indicators, the residuals of which were linked via a chain graph. All parameter values were the same as in simulation study 2. The design was the same as in simulation study 3, leading to 5 000 generated datasets on which AIC, BIC or EBIC were used to select the best model. While Figure 7.9
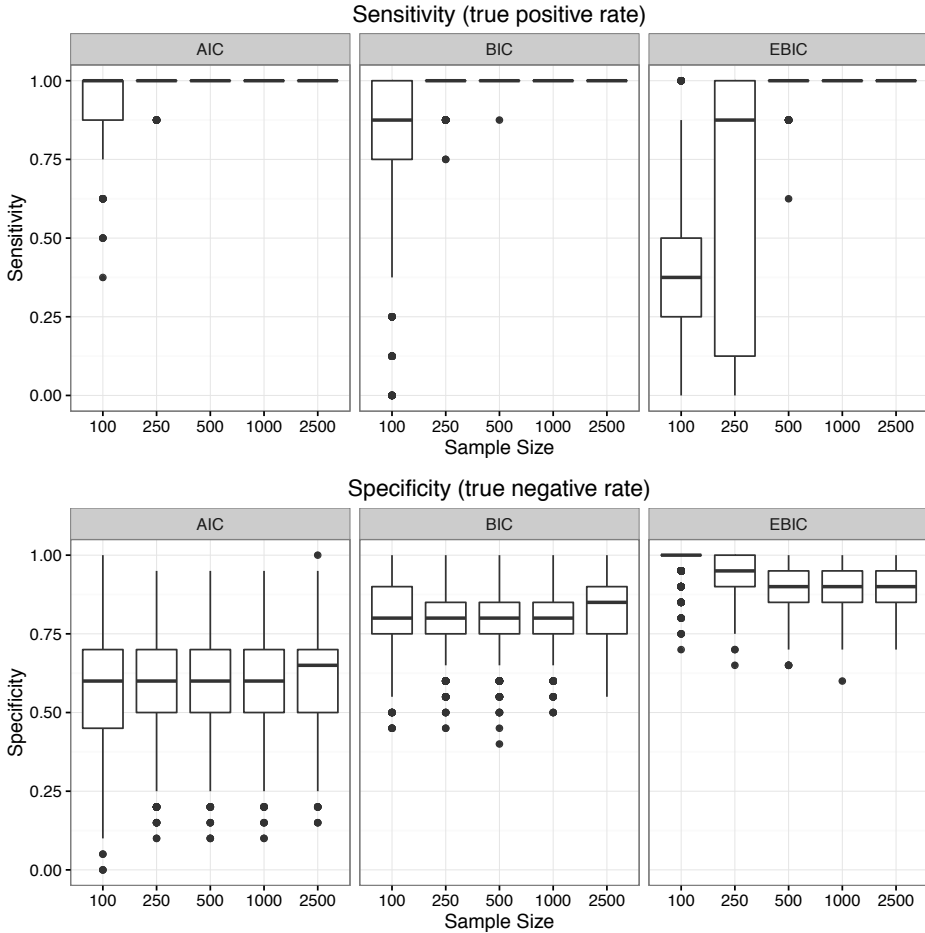
Figure 7.8: Simulation results of simulation study 3: model selection via penalized maximum likelihood estimation in latent network modeling. The same model as in Figure 7.4 was used except now with 4 latent variables leading to 24 observed variables. For each sample size 1 000 datasets were generated, leading to 5 000 total simulated datasets on which AIC, BIC or EBIC was used to select the best model. High sensitivity indicates that the method is able to detect edges in the true model, and high specificity indicates that the method does not detect edges that are zero in the true model.
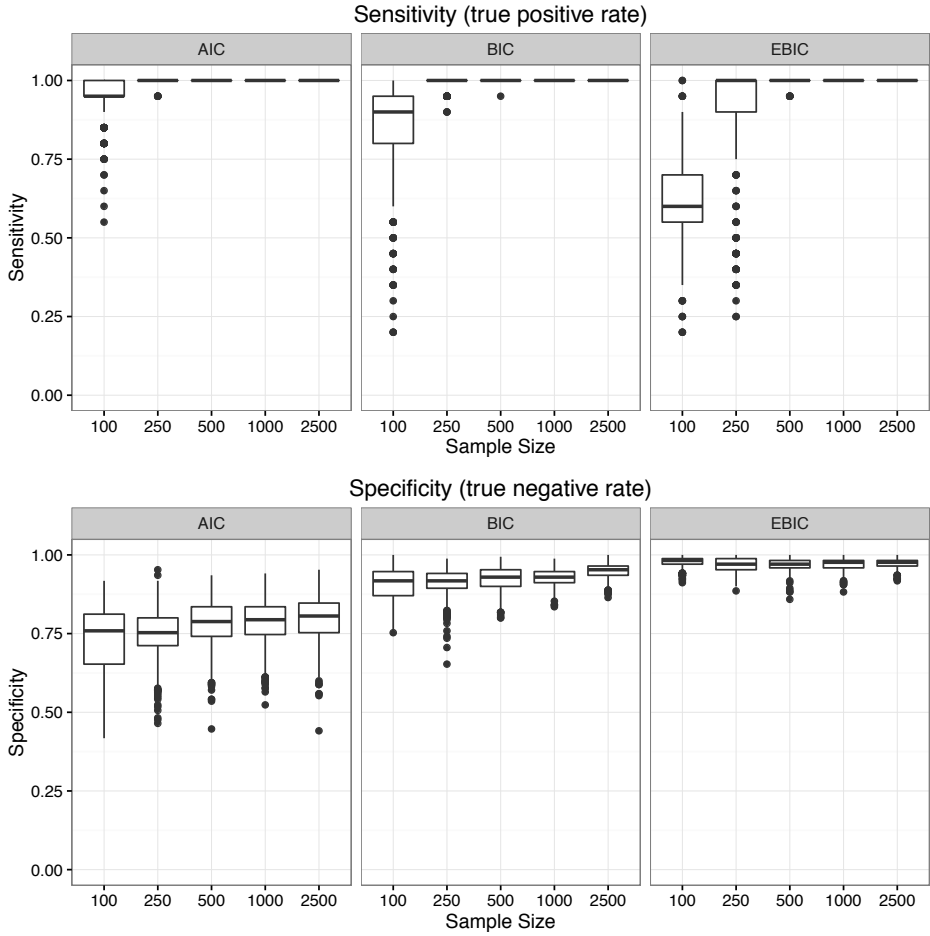
Figure 7.9: Simulation results of simulation study 4: model selection via penalized maximum likelihood estimation in residual network modeling. The same model as in Figure 7.6 was used except now with 4 latent variables leading to 20 observed variables. For each sample size 1 000 datasets were generated, leading to 5 000 total simulated datasets on which AIC, BIC or EBIC was used to select the best model. High sensitivity indicates that the method is able to detect edges in the true model, and high specificity indicates that the method does not detect edges that are zero in the true model.

shows good performance of the LASSO in retrieving the residual network structure and similar results as before: AIC performs the worst in specificity and EBIC the best.

7.5   Empirical Example: Personality Inventory

In this section, we demonstrate LNM and RNM models by confirmative testing a model and exploratively searching a residual and a latent network structure. We use the `lvnet` package, which can be installed directly from Github using the devtools package in R:

```
> library("devtools")
> install_github("sachaepskamp/lvnet")
> library("lvnet")
```

To exemplify the method, we will use a dataset from the psych package (Revelle, 2010) on the Big 5 personality traits (Benet-Martinez & John, 1998; Digman, 1989; Goldberg, 1990a, 1993; McCrae & Costa, 1997). This dataset consists of 2800 observations of 25 items designed to measure the 5 central personality traits with 5 items per trait. We estimated the CFA model on the BFI dataset. Next, we used LASSO estimation to the RNM model using 100 different tuning parameters and using EBIC as criterion to maximize specificity and search for a sparse residual network. The fully correlated covariance matrix of latent variables is equivalent to a fully connected latent network structure. Thus, after fitting a RNM model, we can again apply LASSO to estimate a latent network in the resulting model, which we abbreviate here to an RLNM model. The R code used for this analysis can be found in the supplementary materials.

	df	χ^2	AIC	BIC	EBIC	RMSEA	TLI	CFI
CFA	265	4713.94	183233.7	183589.9	184542.4	0.08	0.75	0.78
RNM	172	806.63	179511.0	180419.4	182848.2	0.04	0.94	0.97
RLNM	176	843.18	179539.5	180424.2	182789.5	0.04	0.94	0.97

Table 7.1: Fit measures for three models estimated on the BFI dataset in the psych R package. CFA is the correlated five-factor model. RNM is the same model as the CFA model with a residual network. RLNM denotes the same model as the RNM model in which edges of the latent network have been removed.

Table 7.1 shows the fit of the three models. The CFA model fits poorly. The RNM model has substantively improved fit and resulted in good fit indices overall. The estimated RLNM model showed that 5 edges could be removed from the latent network after taking residual interactions into account. Figure 7.10 shows the factor structure and residual network of the final RLNM model. It can be seen that Agreeableness is now only connected to extraversion: after taking into account someone's level of extraversion agreeableness is independent of the other three personality traits. Extraversion is the most central node in this network and the only trait that is directly linked to all other traits. The residual network shows many meaningful connections. While seemingly densely connected, this network only has 30% of all possible edges in a network of that size, leading the model to have 176 degrees of freedom. The corresponding residual covariance structure is fully populated with no zero elements.

It should be noted that the procedures used in this example are highly explorative. While Table 7.1 shows that the RNM fits better than the CFA model,

Agreeableness
- A1: Am indifferent to the feelings of others.
- A2: Inquire about others' well–being.
- A3: Know how to comfort others.
- A4: Love children.
- A5: Make people feel at ease.

Conscientiousness
- C1: Am exacting in my work.
- C2: Continue until everything is perfect.
- C3: Do things according to a plan.
- C4: Do things in a half–way manner.
- C5: Waste my time.

Extraversion
- E1: Don't talk a lot.
- E2: Find it difficult to approach others.
- E3: Know how to captivate people.
- E4: Make friends easily.
- E5: Take charge.

Neuroticism
- N1: Get angry easily.
- N2: Get irritated easily.
- N3: Have frequent mood swings.
- N4: Often feel blue.
- N5: Panic easily.

Openness
- O1: Am full of ideas.
- O2: Avoid difficult reading material.
- O3: Carry the conversation to a higher level.
- O4: Spend time reflecting on things.
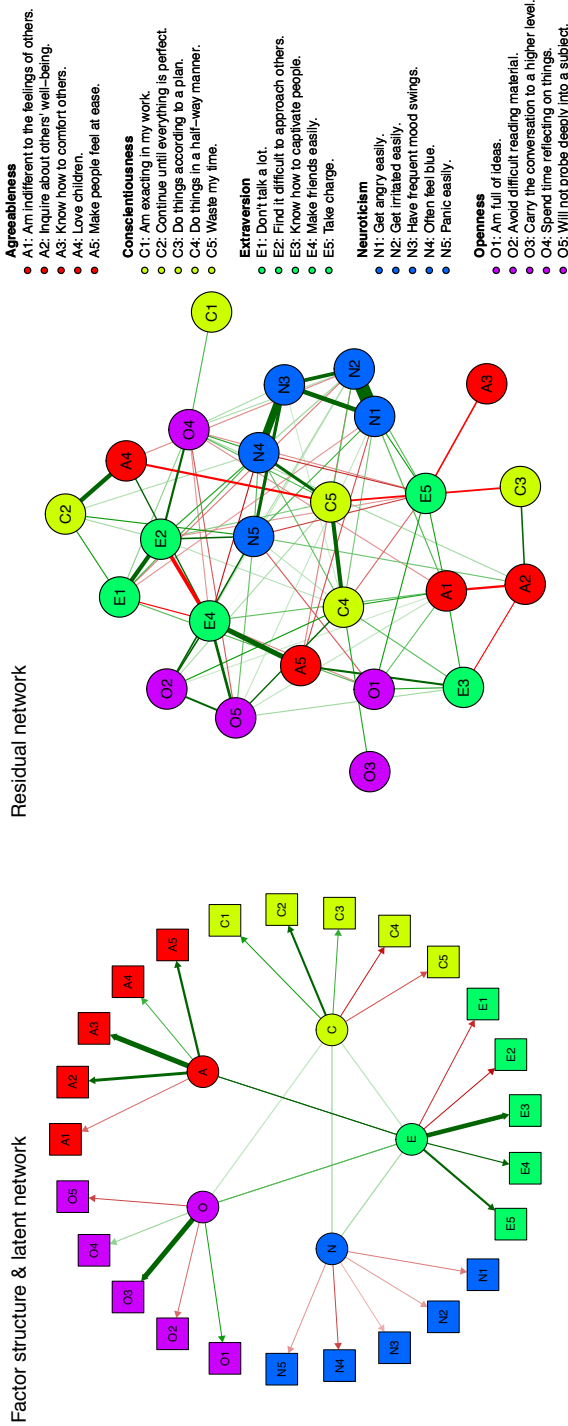- O5: Will not probe deeply into a subject.

Figure 7.10: Visualization of the factor structure and latent network (left) and the residual network (right) of the BFI personality dataset from the psych package in R. LASSO estimation with 100 different tuning parameters in combination with EBIC model selection was used to first estimate the residual network structure and following the latent network structure.

139

the CFA model is solely based on theory whereas the RNM model was found through high-dimensional model search. As a result, to substantially interpret the structure found in Figure 7.10 it should first be replicated in independent samples.

7.6 Conclusion

In this chapter we introduced a formal psychometric model for network modeling of multivariate normal data. We contrasted this model with latent variable models as commonly used in CFA and SEM. Furthermore, using the CFA and SEM frameworks, we proposed two generalizations of the network model to encompass latent variable structures within the network paradigm. In the first generalization, LNM, we construct a network among the latent variables, whereas in the second generalization, RNM, a network is formed among residuals of indicators. Both frameworks offer powerful benefits over both latent variable and network modeling. From the perspective of latent variable modeling, the LNM framework allows one to exploratively search for conditional independence relationships between latent variables without the need for prior theory, and the RNM framework allows one to model latent common causes without assuming local independence. From the perspective of network modeling, the LNM framework allows one to model network structures while taking measurement error into account, and the RNM framework allows one to estimate a network structure, even when all nodes are in part caused by unobserved or latent variables. In addition, both frameworks allow for network models to be fitted and compared to SEM models. The discussed methodology has been implemented in the freely available R package `lvnet`.

Simulation studies showed that step-wise search and penalized maximum likelihood estimation of the residual or latent network structures resulted in high specificity all around—the methods did not result often in false positives—and rapidly increasing sensitivity as a function of the sample size; the higher the sample size, the more true edges were detected in the algorithm. These numbers are comparable to state-of-the-art network estimation techniques in sample and model sizes that are plausible in psychological settings (van Borkulo et al., 2014). In all four simulation studies, using AIC as the model selection criterion led to the best sensitivity and using EBIC led to the best specificity. However, it is important to note that the choice of a particular information criterion cannot be argued by these numbers alone, and depends on the relative importance one assigns to the side of discovery (optimizing sensitivity) or the side of caution (optimizing specificity; Dziak et al., 2012). Furthermore, it should be noted that these simulation results are specific for the particular model setup and sample sizes used; results might be different for other kinds of models or sample size ranges.

In addition to the LNM and RNM frameworks, other combinations of CFA, SEM and network modeling are possible as well. For example, a framework can be constructed which contains both a latent and a residual network (as shown in our empirical example), or directed regression paths as in the SEM model can be added to the LNM model. While these models are all estimable in the `lvnet` software, in the current chapter we chose to focus on the distinct benefits that modeling a residual or latent network presents. Thus, in this chapter, we only described the

modeling of multivariate normal data. More advanced models are possible, but not yet implemented in the `lvnet` software. In the case of binary variables, the appropriate model to use is the Ising model, which has been shown to be equivalent to multivariate item response models (see Chapter 8). Future research could aim at constructing Ising models among binary latent variables in latent class analysis, or constructing residual networks in models with binary indicators. Finally, the expressions optimized in Equations (7.1) and (7.7) are based on summary statistics and therefore only truly applicable to complete data. With incomplete data, the appropriate estimation method is to use full-information maximum likelihood (FIML; Arbuckle, Marcoulides, & Schumacker, 1996); however, FIML has not yet been implemented in the `lvnet` software.

In our view, the presented modeling framework is a versatile and promising addition to the spectrum of psychometric models. The GGM, which has a central place in this modeling framework, acts as a natural interface between correlation and causality, and we think this representation should receive more attention in psychometrics. From the point of view afforded by the current chapter, the typical attempt to determine directed SEMs from correlation structures in fact appears somewhat haphazard in psychology, a historical accident in a field that has been prematurely directed to hypothesis testing at the expense of systematic exploration. Perhaps, psychometrics as a field should consider taking a step back to focus on the consistent identification of GGMs, instead of wanting to jump to the causal conclusion immediately. In this regard, the fact that GGMs do not have equivalent models would appear to be a major benefit, as they allow us to focus on charting connections between variables systematically, without being forced to adhere to one particular causal interpretation or another. In addition, because the GGM does not specify the nature or direction of interactions between variables, it appears a natural model for research situations where no temporal information or experimental interventions are present, so that associations may arise for a multitude of reasons: the GGM can be consistently interpreted regardless of whether associations arise from direct causal relations, reciprocal causation, latent common causes, semantic overlap between items, or homeostatic couplings of parameters. While this can be seen as a downside of the GGM—the lack of directionality leads to less falsifiable hypotheses—it can also be a major asset in a field like psychology, where strong causal theory is sparse and the identification of DAGs often appears a bridge too far.

Chapter 8

The Ising Model in Psychometrics

Abstract

This chapter provides a general introduction of network modeling in psychometrics. The chapter starts with an introduction to the statistical model formulation of pairwise Markov random fields (PMRF), followed by an introduction of the PMRF suitable for binary data: the *Ising model*. The Ising model is a model used in ferromagnetism to explain phase transitions in a field of particles. Following the description of the Ising model in statistical physics, the chapter continues to show that the Ising model is closely related to models used in psychometrics. The Ising model can be shown to be equivalent to certain kinds of logistic regression models, loglinear models and multi-dimensional item response theory (MIRT) models. The equivalence between the Ising model and the MIRT model puts standard psychometrics in a new light and leads to a strikingly different interpretation of well-known latent variable models. The chapter gives an overview of methods that can be used to estimate the Ising model, and concludes with a discussion on the interpretation of latent variables given the equivalence between the Ising model and MIRT.

8.1 Introduction

In recent years, network models have been proposed as an alternative way of looking at psychometric problems (Van Der Maas et al., 2006; Cramer et al., 2010; Borsboom & Cramer, 2013). In these models, psychometric item responses are conceived of as proxies for variables that directly interact with each other. For example, the symptoms of depression (such as loss of energy, sleep problems, and low self esteem) are traditionally thought of as being determined by a common latent variable (depression, or the liability to become depressed; Aggen, Neale, &

This chapter has been adapted from: Epskamp, S., Maris, G., Waldorp, L.J., and Borsboom, D. (in press). Network Psychometrics. In Irwing, P., Hughes, D., and Booth, T. (Eds.), *Handbook of Psychometrics*. New York: Wiley.

Kendler, 2005). In network models, these symptoms are instead hypothesized to form networks of mutually reinforcing variables (e.g., sleep problems may lead to loss of energy, which may lead to low self esteem, which may cause rumination that in turn may reinforce sleep problems). On the face of it, such network models offer an entirely different conceptualization of why psychometric variables cluster in the way that they do. However, it has also been suggested in the literature that latent variables may somehow correspond to sets of tightly intertwined observables (e.g., see the Appendix of Van Der Maas et al., 2006).

In the current chapter, we aim to make this connection explicit. As we will show, a particular class of latent variable models (namely, multidimensional Item Response Theory models) yields exactly the same probability distribution over the observed variables as a particular class of network models (namely, Ising models). In the current chapter, we exploit the consequences of this equivalence. We will first introduce the general class of models used in network analysis called Markov Random Fields. Specifically, we will discuss the Markov random field for binary data called the *Ising Model*, which originated from statistical physics but has since been used in many fields of science. We will show how the Ising Model relates to psychometrical practice, with a focus on the equivalence between the Ising Model and multidimensional item response theory. We will demonstrate how the Ising model can be estimated and finally, we will discuss the conceptual implications of this equivalence.

Notation

Throughout this chapter we will denote random variables with capital letters and possible realizations with lower case letters; vectors will be represented with bold-faced letters. For parameters, we will use boldfaced capital letters to indicate matrices instead of vectors whereas for random variables we will use boldfaced capital letters to indicate a random vector. Roman letters will be used to denote observable variables and parameters (such as the number of nodes) and Greek letters will be used to denote unobservable variables and parameters that need to be estimated.

In this chapter we will mainly model the random vector $\boldsymbol{X}$:

$$\boldsymbol{X}^{\top} = \begin{bmatrix} X_1 & X_2 & \ldots & X_P \end{bmatrix},$$

containing P binary variables that take the values 1 (e.g., correct, true or yes) and -1 (e.g., incorrect, false or no). We will denote a realization, or *state*, of $\boldsymbol{X}$ with $\boldsymbol{x}^{\top} = \begin{bmatrix} x_1 & x_2 & \ldots & x_p \end{bmatrix}$. Let N be the number of observations and $n(\boldsymbol{x})$ the number of observations that have response pattern $\boldsymbol{x}$. Furthermore, let i denote the subscript of a random variable and j the subscript of a different random variable ($j \neq i$). Thus, X_i is the ith random variable and x_i its realization. The superscript $-(\ldots)$ will indicate that elements are removed from a vector; for example, $\boldsymbol{X}^{-(i)}$ indicates the random vector $\boldsymbol{X}$ without X_i: $\boldsymbol{X}^{-(i)} = \begin{bmatrix} X_1, \ldots, X_{i-1}, X_{i+1}, \ldots X_P \end{bmatrix}$, and $\boldsymbol{x}^{-(i)}$ indicates its realization. Similarly, $\boldsymbol{X}^{-(i,j)}$ indicates $\boldsymbol{X}$ without X_i and X_j and $\boldsymbol{x}^{-(i,j)}$ its realization. An overview of all notations used in this chapter can be seen in Appendix B.
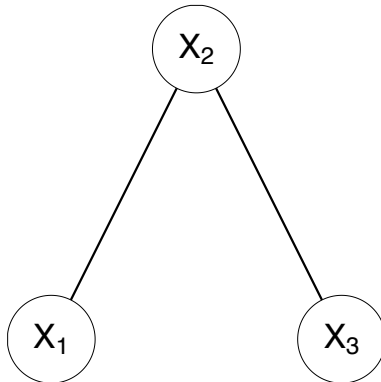
Figure 8.1: Example of a PMRF of three nodes, X_1, X_2 and X_3 , connected by two edges, one between X_1 and X_2 and one between X_2 and X_3.

8.2 Markov Random Fields

A network, also called a graph, can be encoded as a set G consisting of two sets: V, which contains the nodes in the network, and E, which contains the edges that connect these nodes. For example, the graph in Figure 8.1 contains three nodes: $V = \{1, 2, 3\}$, which are connected by two edges: $E = \{(1, 2), (2, 3)\}$. We will use this type of network to represent a *pairwise Markov random field* (PMRF; Lauritzen, 1996; Murphy, 2012), in which nodes represent observed random variables[1] and edges represent (conditional) association between two nodes. More importantly, the absence of an edge represents the Markov property that two nodes are conditionally independent given all other nodes in the network:

$$X_i \perp\!\!\!\perp X_j \mid \boldsymbol{X}^{-(i,j)} = \boldsymbol{x}^{-(i,j)} \iff (i, j) \notin E \qquad (8.1)$$

Thus, a PMRF encodes the independence structure of the system of nodes. In the case of Figure 8.1, X_1 and X_3 are independent given that we know $X_2 = x_2$. This could be due to several reasons; there might be a causal path from X_1 to X_3 or vise versa, X_2 might be the common cause of X_1 and X_3, unobserved variables might cause the dependencies between X_1 and X_2 and X_2 and X_3, or the edges in the network might indicate actual pairwise interactions between X_1 and X_2 and X_2 and X_3.

Of particular interest to psychometrics are models in which the presence of latent common causes induces associations among the observed variables. If such a common cause model holds, we cannot condition on any observed variable to completely remove the association between two nodes (Pearl, 2000). Thus, if an unobserved variable acts as a common cause to some of the observed variables, we should find a fully connected clique in the PMRF that describes the associations

---

[1]Throughout this chapter, nodes in a network designate variables, hence the terms are used interchangeably.

among these nodes. The network in Figure 8.1, for example, cannot represent associations between three nodes that are subject to the influence of a latent common cause; if that were the case, it would be impossible to obtain conditional independence between X_1 and X_3 by conditioning on X_2.

Parameterizing Markov Random Fields

A PMRF can be parameterized as a product of strictly positive potential functions $\phi(x)$ (Murphy, 2012):

$$\Pr\left(\boldsymbol{X} = \boldsymbol{x}\right) = \frac{1}{Z} \prod_i \phi_i\left(x_i\right) \prod_{<ij>} \phi_{ij}\left(x_i, x_j\right),$$ (8.2)

in which $\prod_i$ takes the product over all nodes, $i = 1, 2, \ldots, P$, $\prod_{<ij>}$ takes the product over all distinct pairs of nodes i and j $(j > i)$, and Z is a normalizing constant such that the probability function sums to unity over all possible patterns of observations in the sample space:

$$Z = \sum_{\boldsymbol{x}} \prod_i \phi_i\left(x_i\right) \prod_{<ij>} \phi_{ij}\left(x_i, x_j\right).$$

Here, $\sum_{\boldsymbol{x}}$ takes the sum over all possible realizations of $\boldsymbol{X}$. All $\phi(x)$ functions result in positive real numbers, which encode the *potentials*: the preference for the relevant part of $\boldsymbol{X}$ to be in some state. The $\phi_i(x_i)$ functions encode the node potentials of the network; the preference of node X_i to be in state x_i, regardless of the state of the other nodes in the network. Thus, $\phi_i(x_i)$ maps the potential for X_i to take the value x_i regardless of the rest of the network. If $\phi_i(x_i) = 0$, for instance, then X_i will never take the value x_i, while $\phi_i(x_i) = 1$ indicates that there is no preference for X_i to take any particular value and $\phi_i(x_i) = \infty$ indicates that the system always prefers X_i to take the value x_i. The $\phi_{ij}(x_i, x_j)$ functions encode the pairwise potentials of the network; the preference of nodes X_i and X_j to both be in states x_i and x_j. As $\phi_{ij}(x_i, x_j)$ grows higher we would expect to observe $X_j = x_j$ whenever $X_i = x_i$. Note that the potential functions are not identified; we can multiply both $\phi_i(x_i)$ or $\phi_{ij}(x_i, x_j)$ with some constant for all possible outcomes of x_i, in which case this constant becomes a constant multiplier to (8.2) and is cancelled out in the normalizing constant Z. A typical identification constraint on the potential functions is to set the marginal geometric means of all outcomes equal to 1; over all possible outcomes of each argument, the logarithm of each potential function should sum to 0:

$$\sum_{x_i} \ln \phi_i(x_i) = \sum_{x_i} \ln \phi_{ij}(x_i, x_j) = \sum_{x_j} \ln \phi_{ij}(x_i, x_j) = 0 \quad \forall x_i, x_j$$ (8.3)

in which $\sum_{x_i}$ denotes the sum over all possible realizations for X_i, and $\sum_{x_j}$ denotes the sum over all possible realizations of X_j.

We assume that every node has a potential function $\phi_i(x_i)$ and nodes only have a relevant pairwise potential function $\phi_{ij}(x_i, x_j)$ when they are connected by an edge; thus, two unconnected nodes have a constant pairwise potential function

which, due to identification above, is equal to 1 for all possible realizations of X_i and X_j:

$$\phi_{ij}(x_i, x_j) = 1 \quad \forall x_i, x_j \iff (i,j) \notin E. \tag{8.4}$$

From Equation (8.2) it follows that the distribution of $\boldsymbol{X}$ marginalized over X_k and X_l, that is, the marginal distribution of $\boldsymbol{X}^{-(k,l)}$ (the random vector $\boldsymbol{X}$ without elements X_k and X_l), has the following form:

$$
\begin{aligned}
\Pr\left(\boldsymbol{X}^{-(k,l)} = \boldsymbol{x}^{-(k,l)}\right) &= \sum_{x_k, x_l} \Pr\left(\boldsymbol{X} = \boldsymbol{x}\right) \\
&= \frac{1}{Z} \prod_{i \notin \{k,l\}} \phi_i(x_i) \prod_{<ij \notin \{k,l\}>} \phi_{ij}(x_i, x_j) \\
&\qquad \sum_{x_k, x_l} \left(\phi_k(x_k)\phi_l(x_l)\phi_{kl}(x_k, x_l) \prod_{i \notin \{k,l\}} \phi_{ik}(x_i, x_k)\phi_{il}(x_i, x_l) \right),
\end{aligned} \tag{8.5}
$$

in which $\prod_{i \notin \{k,l\}}$ takes the product over all nodes except node k and l and $\prod_{<ij \notin \{k,l\}>}$ takes the product over all unique pairs of nodes that do not involve k and l. The expression in (8.5) has two important consequences. First, (8.5) does not have the form of (8.2); a PMRF is *not* a PMRF under marginalization. Second, dividing (8.2) by (8.5) an expression can be obtained for the conditional distribution of $\{X_k, X_l\}$ given that we know $\boldsymbol{X}^{-(k,l)} = \boldsymbol{x}^{-(k,l)}$:

$$
\begin{aligned}
\Pr\left(X_k, X_l \mid \boldsymbol{X}^{-(k,l)} = \boldsymbol{x}^{-(k,l)}\right) &= \frac{\Pr\left(\boldsymbol{X} = \boldsymbol{x}\right)}{\Pr\left(\boldsymbol{X}^{-(k,l)} = \boldsymbol{x}^{-(k,l)}\right)} \\
&= \frac{\phi_k^*(x_k)\phi_l^*(x_l)\phi_{kl}(x_k, x_l)}{\sum_{x_k, x_l} \phi_k^*(x_k)\phi_l^*(x_l)\phi_{kl}(x_k, x_l)},
\end{aligned} \tag{8.6}
$$

in which:

$$\phi_k^*(x_k) = \phi_k(x_k) \prod_{i \notin \{k,l\}} \phi_{ik}(x_i, x_k)$$

and:

$$\phi_l^*(x_l) = \phi_l(x_l) \prod_{i \notin \{k,l\}} \phi_{il}(x_i, x_l).$$

Now, (8.6) *does* have the same form as (8.2); a PMRF *is* a PMRF under conditioning. Furthermore, if there is no edge between nodes k and l, $\phi_{kl}(x_k, x_l) = 1$ according to (8.4), in which case (8.6) reduces to a product of two independent functions of x_k and x_l which renders X_k and X_l independent; thus proving the Markov property in (8.1).

The Ising Model

The node potential functions $\phi_i(x_i)$ can map a unique potential for every possible realization of X_i and the pairwise potential functions $\phi_{ij}(x_i, x_j)$ can likewise map unique potentials to every possible pair of outcomes for X_i and X_j. When the data are binary, only two realizations are possible for x_i, while four realizations are possible for the pair x_i and x_j. Under the constraint that the log potential

functions should sum to 0 over all marginals, this means that in the binary case each potential function has one degree of freedom. If we let all X's take the values 1 and -1, there exists a conveniently loglinear model representation for the potential functions:

$$\ln \phi_i(x_i) = \tau_i x_i$$
$$\ln \phi_{ij}(x_i, x_j) = \omega_{ij} x_i x_j.$$

The parameters τ_i and ω_{ij} are real numbers. In the case that $x_i = 1$ and $x_j = 1$, it can be seen that these parameters form an identity link with the logarithm of the potential functions:

$$\tau_i = \ln \phi_i(1)$$
$$\omega_{ij} = \ln \phi_{ij}(1, 1).$$

These parameters are centered on 0 and have intuitive interpretations. The τ_i parameters can be interpreted as *threshold parameters*. If $\tau_i = 0$ the model does not prefer to be in one state or the other, and if τ_i is higher (lower) the model prefers node X_i to be in state 1 (-1). The ω_{ij} parameters are the *network parameters* and denote the pairwise interaction between nodes X_i and X_j; if $\omega_{ij} = 0$ there is no edge between nodes X_i and X_j:

$$\omega_{ij} \begin{cases} = 0 & \text{if } (i,j) \notin E \\ \in \mathbb{R} & \text{if } (i,j) \in E \end{cases}. \tag{8.7}$$

The higher (lower) ω_{ij} becomes, the more nodes X_i and X_j prefer to be in the same (different) state. Implementing these potential functions in (8.2) gives the following distribution for $\boldsymbol{X}$:

$$\Pr(\boldsymbol{X} = \boldsymbol{x}) = \frac{1}{Z} \exp \left(\sum_i \tau_i x_i + \sum_{<ij>} \omega_{ij} x_i x_j \right) \tag{8.8}$$

$$Z = \sum_{\boldsymbol{x}} \exp \left(\sum_i \tau_i x_i + \sum_{<ij>} \omega_{ij} x_i x_j \right),$$

which is known as the Ising model (Ising, 1925).

For example, consider the PMRF in Figure 8.1. In this network there are three nodes (X_1, X_2 and X_3), and two edges (between X_1 and X_2, and between X_2 and X_3). Suppose these three nodes are binary, and take the values 1 and -1. We can then model this PMRF as an Ising model with 3 threshold parameters, τ_1, τ_2 and τ_3 and two network parameters, ω_{12} and ω_{23}. Suppose we set all threshold parameters to $\tau_1 = \tau_2 = \tau_3 = -0.1$, which indicates that all nodes have a general preference to be in the state -1. Furthermore we can set the two network parameters to $\omega_{12} = \omega_{23} = 0.5$. Thus, X_1 and X_2 prefer to be in the same state, and X_2 and X_3 prefer to be in the same state as well. Due to these interactions, X_1 and X_3 become associated; these nodes also prefer to be in

Table 8.1: Probability of all states from the network in Figure 8.1.

x_1	x_2	x_3	Potential	Probability
-1	-1	-1	3.6693	0.3514
1	-1	-1	1.1052	0.1058
-1	1	-1	0.4066	0.0389
1	1	-1	0.9048	0.0866
-1	-1	1	1.1052	0.1058
1	-1	1	0.3329	0.0319
-1	1	1	0.9048	0.0866
1	1	1	2.0138	0.1928
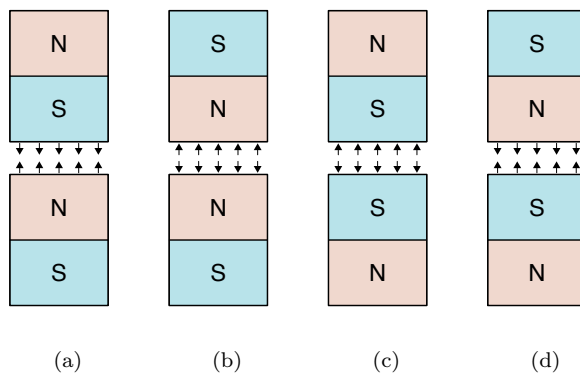
Figure 8.2: Example of the effect of holding two magnets with a north and south pole close to each other. The arrows indicate the direction the magnets want to move; the same poles, as in (b) and (c), repulse each other and opposite poles, as in (a) and (d), attract each other.

the same state, even though they are independent once we condition on X_2. We can then compute the non-normalized potentials $\exp\left(\sum_i \tau_i x_i + \sum_{<ij>} \omega_{ij} x_i x_j\right)$ for all possible outcomes of $\boldsymbol{X}$ and finally divide that value by the sum over all non-normalized potentials to compute the probabilities of each possible outcome. For instance, for the state $X_1 = -1, X_2 = 1$ and $X_3 = -1$, we can compute the potential as $\exp\left(-0.1 + 0.1 + -0.1 + -0.5 + -0.5\right) \approx 0.332$. Computing all these potentials and summing them leads to the normalizing constant of $Z \approx 10.443$, which can then be used to compute the probabilities of each state. These values can be seen in Table 8.1. Not surprisingly, the probability $P(X_1 = -1, X_2 = -1, X_3 = -1)$ is the highest probable state in Table 8.1, due to the threshold parameters being all negative. Furthermore, the probability $P(X_1 = 1, X_2 = 1, X_3 = 1)$ is the second highest probability in Table 8.1; if one node is put into state 1 then all nodes prefer to be in that state due to the network structure.

The Ising model was introduced in statistical physics, to explain the phenomenon of magnetism. To this end, the model was originally defined on a field

(a)

(a)
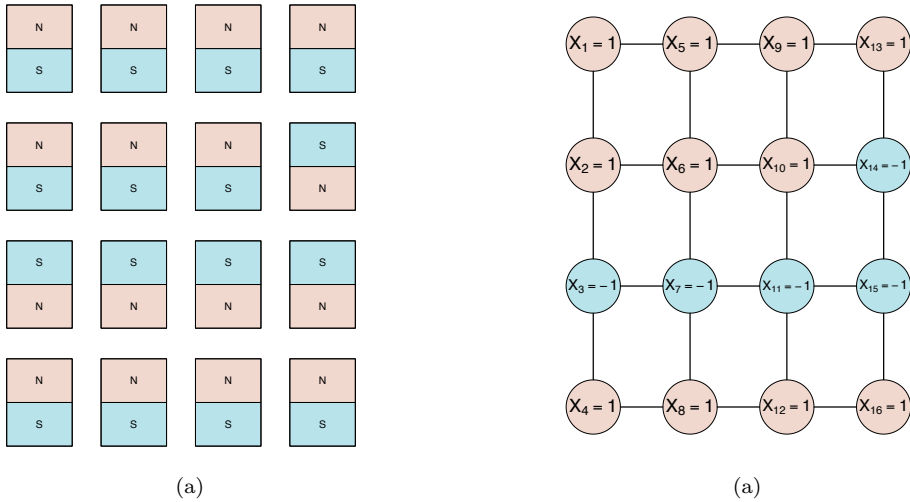
Figure 8.3: A field of particles (a) can be represented by a network shaped as a lattice as in (b). $+1$ indicates that the north pole is alligned upwards and -1 indicates that the south pole is aligned upwards. The lattice in (b) adheres to a PMRF in that the probability of a particle (node) being in some state is only dependent on the state of its direct neighbors.

of particles connected on a lattice. We will give a short introduction on this application in physics because it exemplifies an important aspect of the Ising model; namely, that the interactions between nodes can lead to synchronized behavior of the system as a whole (e.g., spontaneous magnetization). To explain how this works, note that a magnet, such as a common household magnet or the arrow in a compass, has two poles: a north pole and a south pole. Figure 8.2 shows the effect of pushing two such magnets together; the north pole of one magnet attracts to the south pole of another magnet and vise versa, and the same poles on both magnets repulse each other. This is due to the generally tendency of magnets to align, called *ferromagnetism*. Exactly the same process causes the arrow of a compass to align with the magnetic field of the Earth itself, causing it to point north. Any material that is ferromagnetic, such as a plate of iron, consists of particles that behave in the same way as magnets; they have a north and south pole and lie in some direction. Suppose the particles can only lie in two directions: the north pole can be up or the south pole can be up. Figure 8.3 shows a simple 2-dimensional representation of a possible state for a field of 4×4 particles. We can encode each particle as a random variable, X_i, which can take the values -1 (south pole is up) and 1 (north pole is up). Furthermore we can assume that the probability of X_i being in state x_i only depends on the direct neighbors (north, south east and west) of particle i. With this assumption in place, the system in Figure 8.3 can be represented as a PMRF on a lattice, as represented in Figure 8.3.

A certain amount of energy is required for a system of particles to be in some

state, such as in Figure 8.2. For example, in Figure 8.3 the node X_7 is in the state -1 (south pole up). Its neighbors X_3 and X_{11} are both in the same state and thus aligned, which reduces stress on the system and thus reduces the energy function. The other neighbors of X_7, X_6 and X_8, are in the opposite state of X_7, and thus are not aligned, which increasing the stress on the system. The total energy configuration can be summarized in the *Hamiltonian* function:

$$H(\boldsymbol{x}) = -\sum_i \tau_i x_i - \sum_{<i,j>} \omega_{ij} x_i x_j,$$

which is used in the Gibbs distribution (Murphy, 2012) to model the probability of $\boldsymbol{X}$ being in some state $\boldsymbol{x}$:

$$\Pr\left(\boldsymbol{X} = \boldsymbol{x}\right) = \frac{\exp\left(-\beta H(\boldsymbol{x})\right)}{Z}. \tag{8.9}$$

The parameter β indicates the inverse temperature of the system, which is not identifiable since we can multiply β with some constant and divide all τ and ω parameters with that same constant to obtain the same probability. Thus, it can arbitrarily be set to $\beta = 1$. Furthermore, the minus signs in the Gibbs distribution and Hamiltonian cancel out, leading to the Ising model as expressed in (8.8).

The threshold parameters τ_i indicate the natural deposition for particle i to point up or down, which could be due to the influence of an external magnetic field not part of the system of nodes in $\boldsymbol{X}$. For example, suppose we model a single compass, there is only one node thus the Hamiltonian reduces to $-\tau x$. Let $X = 1$ indicate the compass points north and $X = -1$ indicate the compass points south. Then, τ should be positive as the compass has a natural tendency to point north due to the presence of the Earth's magnetic field. As such, the τ parameters are also called external fields. The network parameters ω_{ij} indicate the interaction between two particles. Its sign indicates if particles i and j tend to be in the same state (positive; ferromagnetic) or in different states (negative; anti-ferromagnetic). The absolute value, $|\omega_{ij}|$, indicates the strength of interaction. For any two non-neighboring particles ω_{ij} will be 0 and for neighboring particles the stronger ω_{ij} the stronger the interaction between the two. Because the closer magnets, and thus particles, are moved together the stronger the magnetic force, we can interpret $|\omega_{ij}|$ as a measure for *closeness* between two nodes.

While the inverse temperature β is not identifiable in the sense of parameter estimation, it is an important element in the Ising model; in physics the temperature can be manipulated whereas the ferromagnetic strength or distance between particles cannot. The inverse temperature plays a crucial part in the *entropy* of (8.9) (Wainwright & Jordan, 2008):

$$\begin{aligned} \text{Entropy}\left(\boldsymbol{X}\right) &= \mathbb{E}\left[-\ln \Pr\left(\boldsymbol{X} = \boldsymbol{x}\right)\right] \\ &= -\beta \mathbb{E}\left[-\ln \frac{\exp\left(-H(\boldsymbol{x})\right)}{Z^*}\right], \end{aligned} \tag{8.10}$$

in which Z^* is the rescaled normalizing constant without inverse temperature β. The expectation $\mathbb{E}\left[-\ln \frac{\exp(-H(\boldsymbol{x}))}{Z^*}\right]$ can be recognized as the entropy of the

Ising model as defined in (8.8). Thus, the inverse temperature β directly scales the entropy of the Ising model. As β shrinks to 0, the system is "heated up" and all states become equally likely, causing a high level of entropy. If β is subsequently increased, then the probability function becomes concentrated on a smaller number of states, and the entropy shrinks to eventually only allow the state in which all particles are aligned. The possibility that all particles become aligned is called *spontaneous magnetization* (Lin, 1992; Kac, 1966); when all particles are aligned (all X are either 1 or -1) the entire field of particles becomes magnetized, which is how iron can be turned into a permanent magnet. We take this behavior as a particular important aspect of the Ising model; behavior on microscopic level (interactions between neighboring particles) can cause noticeable behavior on macroscopic level (the creation of a permanent magnet).

In our view, psychological variables may behave in the same way. For example, interactions between components of a system (e.g., symptoms of depression) can cause synchronized effects of the system as a whole (e.g., depression as a disorder). Do note that, in setting up such analogies, we need to interpret the concepts of closeness and neighborhood less literally than in the physical sense. Concepts such as "sleep deprivation" and "fatigue" can be said to be close to each other, in that they mutually influence each other; sleep deprivation can lead to fatigue and in turn fatigue can lead to a disrupted sleeping rhythm. The neighborhood of these symptoms can then be defined as the symptoms that frequently co-occur with sleep deprivation and fatigue, which can be seen in a network as a cluster of connected nodes. As in the Ising model, the state of these nodes will tend to be the same if the connections between these nodes are positive. This leads to the interpretation that a latent trait, such as depression, can be seen as a cluster of connected nodes (Borsboom et al., 2011). In the next section, we will prove that there is a clear relationship between network modeling and latent variable modeling; indeed, clusters in a network can cause data to behave as if they were generated by a latent variable model.

8.3   The Ising Model in Psychometrics

In this section, we show that the Ising model is equivalent or closely related to prominent modeling techniques in psychometrics. We will first discuss the relationship between the Ising model and loglinear analysis and logistic regressions, next show that the Ising model can be equivalent to Item Response Theory (IRT) models that dominate psychometrics. In addition, we highlight relevant earlier work on the relationship between IRT and the Ising model.

To begin, we can gain further insight in the Ising model by looking at the conditional distribution of X_i given that we know the value of the remaining

nodes: $\boldsymbol{X}^{(-i)} = \boldsymbol{x}^{(-i)}$:

$$\Pr\left(X_i \mid \boldsymbol{X}^{(-i)} = \boldsymbol{x}^{(-i)}\right) = \frac{\Pr\left(\boldsymbol{X} = \boldsymbol{x}\right)}{\Pr\left(\boldsymbol{X}^{(-i)} = \boldsymbol{x}^{(-i)}\right)}$$

$$= \frac{\Pr\left(\boldsymbol{X} = \boldsymbol{x}\right)}{\sum_{x_i} \Pr\left(X_i = x_i, \boldsymbol{X}^{(-i)} = \boldsymbol{x}^{(-i)}\right)}$$

$$= \frac{\exp\left(x_i\left(\tau_i + \sum_j \omega_{ij} x_j\right)\right)}{\sum_{x_i} \exp\left(x_i\left(\tau_k + \sum_j \omega_{ij} x_j\right)\right)}, \qquad (8.11)$$

in which $\sum_{x_i}$ takes the sum over both possible outcomes of x_i. We can recognize this expression as a *logistic regression* model (Agresti, 1990). Thus, the Ising model can be seen as the joint distribution of response and predictor variables, where each variable is predicted by all other variables in the network. The Ising model therefore forms a predictive network in which the neighbors of each node, the set of connected nodes, represent the variables that predict the outcome of the node of interest.

Note that the definition of Markov random fields in (8.2) can be extended to include higher order interaction terms:

$$\Pr\left(\boldsymbol{X} = \boldsymbol{x}\right) = \frac{1}{Z} \prod_i \phi_i\left(x_i\right) \prod_{<ij>} \phi_{ij}\left(x_i, x_j\right) \prod_{<ijk>} \phi_{ijk}\left(x_i, x_j, x_k\right) \cdots,$$

all the way up to the P-th order interaction term, in which case the model becomes saturated. Specifying $\nu_{...}(\dots) = \ln \phi_{...}(\dots)$ for all potential functions, we obtain a log-linear model:

$$\Pr\left(\boldsymbol{X} = \boldsymbol{x}\right) = \frac{1}{Z} \exp\left(\sum_i \nu_i\left(x_i\right) + \sum_{<ij>} \nu_{ij}\left(x_i, x_j\right) + \sum_{<ijk>} \nu_{ijk}\left(x_i, x_j, x_k\right) \cdots\right).$$

Let $n(\boldsymbol{x})$ be the number of respondents with response pattern $\boldsymbol{x}$ from a sample of N respondents. Then, we may model the expected frequency $n(\boldsymbol{x})$ as follows:

$$\mathbb{E}\left[n(\boldsymbol{x})\right] = N \Pr\left(\boldsymbol{X} = \boldsymbol{x}\right)$$

$$= \exp\left(\nu + \sum_i \nu_i\left(x_i\right) + \sum_{<ij>} \nu_{ij}\left(x_i, x_j\right) + \sum_{<ijk>} \nu_{ijk}\left(x_i, x_j, x_k\right) \cdots\right), \qquad (8.12)$$

in which $\nu = \ln N - \ln Z$. The model in (8.12) has extensively been used in loglinear analysis (Agresti, 1990; Wickens, 1989)[2]. In loglinear analysis, the same constrains are typically used as in (8.3); all ν functions should sum to 0 over all margins. Thus, if at most second-order interaction terms are included in the loglinear model, it is equivalent to the Ising model and can be represented exactly as in (8.8). The Ising model, when represented as a loglinear model with at most second-order interactions, has been used in various ways. Agresti (1990) and Wickens (1989) call the model the *homogeneous association* model. Because it

---

[2]both Agresti and Wickens used λ rather than ν to denote the log potentials, which we changed in this chapter to avoid confusion with eigenvalues and the LASSO tuning parameter.

does not include three-way or higher order interactions, the association between X_i and X_j—the odds-ratio—is constant for any configuration of $\boldsymbol{X}^{-(i,j)}$. Also, Cox (1972; Cox & Wermuth, 1994) used the same model, but termed it the *quadratic exponential binary distribution*, which has since often been used in biometrics and statistics (e.g., Fitzmaurice, Laird, & Rotnitzky, 1993; Zhao & Prentice, 1990). Interestingly, none of these authors mention the Ising model.

The Relation Between the Ising Model and Item Response Theory

In this section we will show that the Ising model is a closely related modeling framework of Item Response Theory (IRT), which is of central importance to psychometrics. In fact, we will show that the Ising model is equivalent to a special case of the multivariate 2-parameter logistic model (MIRT). However, instead of being hypothesized common causes of the item responses, in our representation the latent variables in the model are *generated* by cliques in the network.

In IRT, the responses on a set of binary variables $\boldsymbol{X}$ are assumed to be determined by a set of M ($M \leq P$) latent variables $\boldsymbol{\Theta}$:

$$\boldsymbol{\Theta}^\top = \begin{bmatrix} \Theta_1 & \Theta_2 & \dots & \Theta_M \end{bmatrix}.$$

These latent variables are often denoted as *abilities*, which betrays the roots of the model in educational testing. In IRT, the probability of obtaining a realization x_i on the variable X_i—often called *items*—is modeled through item response functions, which model the probability of obtaining one of the two possible responses (typically, scored 1 for correct responses and 0 for incorrect responses) as a function of $\boldsymbol{\theta}$. For instance, in the Rasch (1960) model, also called the one parameter logistic model (1PL), only one latent trait is assumed ($M = 1$ and $\boldsymbol{\Theta} = \Theta$) and the conditional probability of a response given the latent trait takes the form of a simple logistic function:

$$\Pr(X_i = x_i \mid \Theta = \theta)_{1\mathrm{PL}} = \frac{\exp\left(x_i \alpha \left(\theta - \delta_i\right)\right)}{\sum_{x_i} \exp\left(x_i \alpha \left(\theta - \delta_i\right)\right)},$$

in which δ_i acts as a *difficulty parameter* and α is a common *discrimination* parameter for all items. A typical generalization of the 1PL is the Birnbaum (1968) model, often called the two-parameter logistic model (2PL), in which the discrimination is allowed to vary between items:

$$\Pr(X_i = x_i \mid \Theta = \theta)_{2\mathrm{PL}} = \frac{\exp\left(x_i \alpha_i \left(\theta - \delta_i\right)\right)}{\sum_{x_i} \exp\left(x_i \alpha_i \left(\theta - \delta_i\right)\right)}.$$

The 2PL reduces to the 1PL if all discrimination parameters are equal: $\alpha_1 = \alpha_2 = \dots = \alpha$. Generalizing the 2PL model to more than 1 latent variable ($M > 1$) leads to the 2PL multidimensional IRT model (MIRT; Reckase, 2009):

$$\Pr(X_i = x_i \mid \boldsymbol{\Theta} = \boldsymbol{\theta})_{\mathrm{MIRT}} = \frac{\exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i\right)\right)}{\sum_{x_i} \exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i\right)\right)}, \tag{8.13}$$

in which $\boldsymbol{\theta}$ is a vector of length M that contains the realization of $\boldsymbol{\Theta}$, while $\boldsymbol{\alpha}_i$ is a vector of length M that contains the discrimination of item i on every latent trait in the multidimensional space. The MIRT model reduces to the 2PL model if $\boldsymbol{\alpha}_i$ equals zero in all but one of its elements.

Because IRT assumes local independence—the items are independent of each other after conditioning on the latent traits—the joint conditional probability of $\boldsymbol{X} = \boldsymbol{x}$ can be written as a product of the conditional probabilities of each item:

$$\Pr(\boldsymbol{X} = \boldsymbol{x} \mid \boldsymbol{\Theta} = \boldsymbol{\theta}) = \prod_i \Pr(X_i = x_i \mid \boldsymbol{\Theta} = \boldsymbol{\theta}). \tag{8.14}$$

The marginal probability, and thus the likelihood, of the 2PL MIRT model can be obtained by integrating over distribution $f(\boldsymbol{\theta})$ of $\boldsymbol{\Theta}$:

$$\Pr(\boldsymbol{X} = \boldsymbol{x}) = \int_{-\infty}^{\infty} f(\boldsymbol{\theta}) \Pr(\boldsymbol{X} = \boldsymbol{x} \mid \boldsymbol{\Theta} = \boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta}, \tag{8.15}$$

in which the integral is over all M latent variables. For typical distributions of $\boldsymbol{\Theta}$, such as a multivariate Gaussian distribution, this likelihood does not have a closed form solution. Furthermore, as M grows it becomes hard to numerically approximate (8.15). However, if the distribution of $\boldsymbol{\Theta}$ is chosen such that it is conditionally Gaussian—the posterior distribution of $\boldsymbol{\Theta}$ given that we observed $\boldsymbol{X} = \boldsymbol{x}$ takes a Gaussian form—we *can* obtain a closed form solution for (8.15). Furthermore, this closed form solution is, in fact, the Ising model as presented in (8.8).

As also shown by Marsman et al. (2015) and in more detail in Appendix A of this chapter, after reparameterizing $\tau_i = -\delta_i$ and $-2\sqrt{\lambda_j/2}q_{ij} = \alpha_{ij}$, in which q_{ij} is the ith element of the jth eigenvector of $\boldsymbol{\Omega}$ (with an arbitrary diagonal chosen such that $\boldsymbol{\Omega}$ is positive definite), the Ising model is equivalent to a MIRT model in which the posterior distribution of the latent traits is equal to the product of univariate normal distributions with equal variance:

$$\Theta_j \mid \boldsymbol{X} = \boldsymbol{x} \sim N\left(\pm\frac{1}{2}\sum_i a_{ij}x_i, \sqrt{\frac{1}{2}} \right).$$

The mean of these univariate posterior distributions for Θ_j is equal to the weighted sumscore $\pm\frac{1}{2}\sum_i a_{ij}x_i$. Finally, since

$$f(\boldsymbol{\theta}) = \sum_{\boldsymbol{x}} f(\boldsymbol{\theta} \mid \boldsymbol{X} = \boldsymbol{x}) \Pr(\boldsymbol{X} = \boldsymbol{x}),$$

we can see that the marginal distribution of $\boldsymbol{\Theta}$ in (8.15) is a *mixture of multivariate Gaussian distributions with homogenous variance–covariance*, with the mixing probability equal to the marginal probability of observing each response pattern.

Whenever $\alpha_{ij} = 0$ for all i and some dimension j—i.e., none of the items discriminate on the latent trait—we can see that the marginal distribution of Θ_j becomes a Gaussian distribution with mean 0 and standard-deviation $\sqrt{1/2}$. This corresponds to complete randomness; all states are equally probable given

the latent trait. When discrimination parameters diverge from 0, the probability function becomes concentrated on particular response patterns. For example, in case X_1 designates the response variable for a very easy item, while X_2 is the response variable for a very hard item, the state in which the first item is answered correctly and the second incorrectly becomes less likely. This corresponds to a decrease in entropy and, as can be seen in (8.10), is related to the *temperature* of the system. The lower the temperature, the more the system prefers to be in states in which all items are answered correctly or incorrectly. When this happens, the distribution of Θ_j diverges from a Gaussian distribution and becomes a bi-modal distribution with two peaks, centered on the weighted sumscores that correspond to situations in which all items are answered correctly or incorrectly. If the entropy is relatively high, $f(\Theta_j)$ can be well approximated by a Gaussian distribution, whereas if the entropy is (extremely) low a mixture of two Gaussian distributions best approximates $f(\Theta_j)$.

For example, consider again the network structure of Figure 8.1. When we parameterized all threshold functions $\tau_1 = \tau_2 = \tau_3 = -0.1$ and all network parameters $\omega_{12} = \omega_{23} = 0.5$ we obtained the probability distribution as specified in Table 8.1. We can form the matrix $\boldsymbol{\Omega}$ first with zeroes on the diagonal:

$$\begin{bmatrix} 0 & 0.5 & 0 \\ 0.5 & 0 & 0.5 \\ 0 & 0.5 & 0 \end{bmatrix},$$

which is not positive semi-definite. Subtracting the lowest eigenvalue, -0.707, from the diagonal gives us a positive semi-definite $\boldsymbol{\Omega}$ matrix:

$$\boldsymbol{\Omega} = \begin{bmatrix} 0.707 & 0.5 & 0 \\ 0.5 & 0.707 & 0.5 \\ 0 & 0.5 & 0.707 \end{bmatrix}.$$

It's eigenvalue decomposition is as follows:

$$\boldsymbol{Q} = \begin{bmatrix} 0.500 & 0.707 & 0.500 \\ 0.707 & 0.000 & -0.707 \\ 0.500 & -0.707 & 0.500 \end{bmatrix}$$

$$\boldsymbol{\lambda} = \begin{bmatrix} 1.414 & 0.707 & 0.000 \end{bmatrix}.$$

Using the transformations $\tau_i = -\delta_i$ and $-2\sqrt{\lambda_j/2}q_{ij} = \alpha_{ij}$ (arbitrarily using the negative root) defined above we can then form the equivalent MIRT model with discrimination parameters $\boldsymbol{A}$ and difficulty parameters $\boldsymbol{\delta}$:

$$\boldsymbol{\delta} = \begin{bmatrix} 0.1 & 0.1 & 0.1 \end{bmatrix}$$

$$\boldsymbol{A} = \begin{bmatrix} 0.841 & 0.841 & 0 \\ 1.189 & 0 & 0 \\ 0.841 & -0.841 & 0 \end{bmatrix}.$$

Thus, the model in Figure 8.1 is equivalent to a model with two latent traits: one defining the general coherence between all three nodes and one defining the
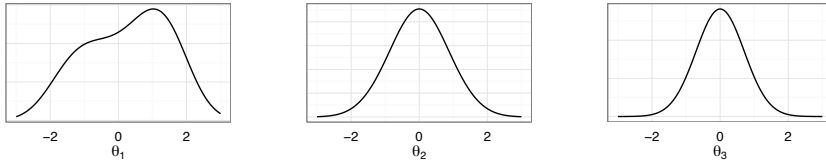
Figure 8.4: The distributions of the three latent traits in the equivalent MIRT model to the Ising model from Figure 8.1

contrast between the first and the third node. The distributions of all three latent traits can be seen in Figure 8.4. In Table 8.1, we see that the probability is the highest for the two states in which all three nodes take the same value. This is reflected in the distribution of the first latent trait in 8.4: because all discrimination parameters relating to this trait are positive, the weighted sumscores of $X_1 = X_2 = X_3 = -1$ and $X_1 = X_2 = X_3 = 1$ are dominant and cause a small bimodality in the distribution. For the second trait, 8.4 shows an approximately normal distribution, because this trait acts as a contrast and cancels out the preference for all variables to be in the same state. Finally, the third latent trait is nonexistent, since all of its discrimination parameters equal 0; 8.4 simply shows a Gaussian distribution with standard deviation $\sqrt{\frac{1}{2}}$.

This proof serves to demonstrate that the Ising model is equivalent to a MIRT model with a posterior Gaussian distribution on the latent traits; the discrimination parameter column vector $\boldsymbol{\alpha_j}$—the item discrimination parameters on the jth dimension—is directly related to the jth eigenvector of the Ising model graph structure $\boldsymbol{\Omega}$, scaled by its jth eigenvector. Thus, the latent dimensions are orthogonal, and the rank of $\boldsymbol{\Omega}$ directly corresponds to the number of latent dimensions. In the case of a Rasch model, the rank of $\boldsymbol{\Omega}$ should be 1 and all ω_{ij} should have exactly the same value, corresponding to the common discrimination parameter; for the uni-dimensional Birnbaum model the rank of $\boldsymbol{\Omega}$ still is 1 but now the ω_{ij} parameters can vary between items, corresponding to differences in item discrimination.

The use of a posterior Gaussian distribution to obtain a closed form solution for (8.15) is itself not new in the psychometric literature, although it has not previously been linked to the Ising model and the literature related to it. Olkin and Tate (1961) already proposed to model binary variables jointly with conditional Gaussian distributed continuous variables. Furthermore, Holland (1990) used the "Dutch identity" to show that a representation equivalent to an Ising model could be used to characterize the marginal distribution of an extended Rasch model (Cressie & Holland, 1983). Based on these results, Anderson and colleagues proposed an IRT modeling framework using log-multiplicative association models and assuming conditional Gaussian latents (Anderson & Vermunt, 2000; Anderson & Yu, 2007); this approach has been implemented in the R package "plRasch" (Anderson, Li, & Vermunt, 2007; Li & Hong, 2014).

With our proof we furthermore show that the clique factorization of the network structure *generated* a latent trait with a functional distribution through a mathematical trick. Thus, the network perspective and common cause perspectives could be interpreted as two different explanations of the same phenomena: cliques of correlated observed variables. In the next section, we show how the Ising model can be estimated.

8.4 Estimating the Ising Model

We can use (8.8) to obtain the log-likelihood function of a realization $\boldsymbol{x}$:

$$\mathcal{L}\left(\boldsymbol{\tau}, \boldsymbol{\Omega}; \boldsymbol{x}\right) = \ln \Pr\left(\boldsymbol{X} = \boldsymbol{x}\right) = \sum_i \tau_i x_i + \sum_{<ij>} \omega_{ij} x_i x_j - \ln Z. \qquad (8.16)$$

Note that the constant Z is only constant with regard to $\boldsymbol{x}$ (as it sums over all possible realizations) and is *not* a constant with regard to the τ and ω parameters; Z is often called the *partition function* because it is a function of the parameters. Thus, while when sampling from the Ising distribution Z does not need to be evaluated, but it *does* need to be evaluated when maximizing the likelihood function. Estimating the Ising model is notoriously hard because the partition function Z is often not tractable to compute (Kolaczyk, 2009). As can be seen in (8.8), Z requires a sum over all possible configurations of $\boldsymbol{x}$; computing Z requires summing over 2^k terms, which quickly becomes intractably large as k grows. Thus, maximum likelihood estimation of the Ising model is only possible for trivially small data sets (e.g., $k < 10$). For larger data sets, different techniques are required to estimate the parameters of the Ising model. Markov samplers can be used to estimate the Ising model by either approximating Z (Sebastiani & Sørbye, 2002; Green & Richardson, 2002; Dryden, Scarr, & Taylor, 2003) or circumventing Z entirely via sampling auxiliary variables (Møller, Pettitt, Reeves, & Berthelsen, 2006; Murray, 2007; Murray, Ghahramani, & MacKay, 2006). Such sampling algorithms can however still be computationally costly.

Because the Ising model is equivalent to the homogeneous association model in log-linear analysis (Agresti, 1990), the methods used in log-linear analysis can also be used to estimate the Ising model. For example, the iterative proportional fitting algorithm (Haberman, 1972), which is implemented in the `loglin` function in the statistical programming language R (R Core Team, 2016), can be used to estimate the parameters of the Ising model. Furthermore, log-linear analysis can be used for model selection in the Ising model by setting certain parameters to zero. Alternatively, while the full likelihood in (8.8) is hard to compute, the conditional likelihood for each node in (8.11) is very easy and does not include an intractable normalizing constant; the conditional likelihood for each node corresponds to a multiple logistic regression (Agresti, 1990):

$$\mathcal{L}_i\left(\boldsymbol{\tau}, \boldsymbol{\Omega}; \boldsymbol{x}\right) = x_i \left(\tau_i + \sum_j \omega_{ij} x_j\right) - \sum_{x_i} \exp\left(x_i \left(\tau_i + \sum_j \omega_{ij} x_j\right)\right).$$

Here, the subscript i indicates that the likelihood function is based on the conditional probability for node i given the other nodes. Instead of optimizing the

full likelihood of (8.8), the pseudolikelihood (PL; Besag, 1975) can be optimized instead. The pseudolikelihood approximates the likelihood with the product of univariate conditional likelihoods in (8.11):

$$\ln \mathrm{PL} = \sum_{i=1}^{k} \mathcal{L}_i \left(\boldsymbol{\tau}, \boldsymbol{\Omega}; \boldsymbol{x} \right)$$

Finally, disjoint pseudolikelihood estimation can be used. In this approach, each conditional likelihood is optimized separately (Liu & Ihler, 2012). This routine corresponds to repeatedly performing a multiple logistic regression in which one node is the response variable and all other nodes are the predictors; by predicting x_i from $\boldsymbol{x}^{(-i)}$ estimates can be obtained for $\boldsymbol{\omega}_i$ and τ_i. After estimating a multiple logistic regression for each node on all remaining nodes, a single estimate is obtained for every τ_i and two estimates are obtained for every ω_{ij}–the latter can be averaged to obtain an estimate of the relevant network parameter. Many statistical programs, such as the R function `glm`, can be used to perform logistic regressions. Estimation of the Ising model via log-linear modeling, maximal pseudolikelihood, and repeated multiple logistic regressions and have been implemented in the `EstimateIsing` function in the R package *IsingSampler* (Epskamp, 2014).

While the above-mentioned methods of estimating the Ising model are tractable, they all require a considerable amount of data to obtain reliable estimates. For example, in log-linear analysis, cells in the 2^P contingency table that are zero—which will occur often if $N < 2^P$—can cause parameter estimates to grow to ∞ (Agresti, 1990), and in logistic regression predictors with low variance (e.g., a very hard item) can substantively increase standard errors (Whittaker, 1990). To estimate the Ising model, P thresholds and $P(P-1)/2$ network parameter have to be estimated, while in standard log linear approaches, rules of thumb suggest that the sample size needs to be three times higher than the number of parameters to obtain reliable estimates. In psychometrics, the number of data points is often far too limited for this requirement to hold. To estimate parameters of graphical models with limited amounts of observations, therefore, regularization methods have been proposed (Meinshausen & Bühlmann, 2006; Friedman et al., 2008).

When regularization is applied, a penalized version of the (pseudo) likelihood is optimized. The most common regularization method is ℓ_1 regularization–commonly known as the least absolute shrinkage and selection operator (LASSO; Tibshirani, 1996)–in which the sum of absolute parameter values is penalized to be under some value. Ravikumar, Wainwright, and Lafferty (2010) employed ℓ_1-regularized logistic regression to estimate the structure of the Ising model via disjoint maximum pseudolikelihood estimation. For each node i the following expression is maximized (Friedman, Hastie, & Tibshirani, 2010):

$$\max_{\tau_i, \boldsymbol{\omega}_i} \left[\mathcal{L}_i \left(\boldsymbol{\tau}, \boldsymbol{\Omega}; \boldsymbol{x} \right) - \lambda \mathrm{Pen} \left(\boldsymbol{\omega}_i \right) \right] \qquad (8.17)$$

Where $\boldsymbol{\omega}_i$ is the ith row (or column due to symmetry) of $\boldsymbol{\Omega}$ and $\mathrm{Pen} \left(\boldsymbol{\omega}_i \right)$ denotes

the penalty function, which is defined in the LASSO as follows:

$$\text{Pen}_{\ell_1}\left(\boldsymbol{\omega}_i\right) = ||\boldsymbol{\omega}_i||_1 = \sum_{j=1,j!=i}^{k} |\omega_{ij}|$$

The λ in (8.17) is the regularization tuning parameter. The problem in above is equivalent to the constrained optimization problem:

$$\max_{\tau_i,\boldsymbol{\omega}_i}\left[\mathcal{L}_i\left(\boldsymbol{\tau},\boldsymbol{\Omega};\boldsymbol{x}\right)\right], \quad \text{subject to } ||\boldsymbol{\omega}_i||_1 < C$$

in which C is a constant that has a one-to-one monotone decreasing relationship with λ (Lee, Lee, Abbeel, & Ng, 2006). If $\lambda = 0$, C will equal the sum of absolute values of the maximum likelihood solution; increasing λ will cause C to be smaller, which forces the estimates of $\boldsymbol{\omega}_i$ to shrink. Because the penalization uses absolute values, this causes parameter estimates to shrink to exactly zero. Thus, in moderately high values for λ a sparse solution to the logistic regression problem is obtained in which many coefficients equal zero; the LASSO results in simple predictive models including only a few predictors.

Ravikumar et al. (2010) used LASSO to estimate the neighborhood—the connected nodes—of each node, resulting in an unweighted graph structure. In this approach, an edge is selected in the model if either ω_{ij} and ω_{ji} is nonzero (the OR-rule) or if both are nonzero (the AND-rule). To obtain estimates for the weights ω_{ij} and ω_{ji} can again be averaged. The λ parameter is typically specified such that an optimal solution is obtained, which is commonly done through cross-validation or, more recently, by optimizing the extended Bayesian information criterion (EBIC; Chen & Chen, 2008; Foygel & Drton, 2010; Foygel Barber & Drton, 2015; van Borkulo et al., 2014).

In K-fold cross-validation, the data are subdivided in K (usually $K = 10$) blocks. For each of these blocks a model is fitted using only the remaining $K - 1$ blocks of data, which are subsequently used to construct a prediction model for the block of interest. For a suitable range of λ values, the predictive accuracy of this model can be computed, and subsequently the λ under which the data were best predicted is chosen. If the sample size is relatively low, the predictive accuracy is typically much better for $\lambda > 0$ than it is at the maximum likelihood solution of $\lambda = 0$; it is preferred to regularize to avoid over-fitting.

Alternatively, an information criterion can be used to directly penalize the likelihood for the number of parameters. The EBIC (Chen & Chen, 2008) augments the Bayesian information Criterion (BIC) with a hyperparameter γ to additionally penalize the large space of possible models (networks):

$$\text{EBIC} = -2\mathcal{L}_i\left(\boldsymbol{\tau},\boldsymbol{\Omega};\boldsymbol{x}\right) + |\boldsymbol{\omega}_i|\ln\left(N\right) + 2\gamma\left|\boldsymbol{\omega}_i\right|\ln\left(k-1\right)$$

in which $|\boldsymbol{\omega}_i|$ is the number of nonzero parameters in $\boldsymbol{\omega}_i$. Setting $\gamma = 0.25$ works well for the Ising model (Foygel Barber & Drton, 2015). An optimal λ can be chosen either for the entire Ising model, which improves parameter estimation, or for each node separately in disjoint pseudolkelihood estimation, which improves neighborhood selection. While K-fold cross-validation does not require the computation of the intractable likelihood function, EBIC does. Thus, when using

EBIC estimation λ need be chosen per node. We have implemented ℓ_1-regularized disjoint pseudolikelihood estimation of the Ising model using EBIC to select a tuning parameter per node in the *R* package *IsingFit* (van Borkulo & Epskamp, 2014; van Borkulo et al., 2014), which uses *glmnet* for optimization (Friedman et al., 2010).

The LASSO works well in estimating sparse network structures for the Ising model and can be used in combination with cross-validation or an information criterion to arrive at an interpretable model. However, it does so under the assumption that the true model in the population is sparse. So what if reality is not sparse, and we would not expect many missing edges in the network? As discussed earlier in this chapter, the absence of edges indicate conditional independence between nodes; if all nodes are caused by an unobserved cause we would not expect missing edges in the network but rather a low-rank network structure. In such cases, ℓ_2 regularization—also called ridge regression—can be used which uses a quadratic penalty function:

$$\text{Pen}_{\ell_2}\left(\boldsymbol{\omega}_i\right) = ||\boldsymbol{\omega}_i||_2 = \sum_{j=1, j!=i}^{k} \omega_{ij}^2$$

With this penalty parameters will not shrink to exactly zero but more or less smooth out; when two predictors are highly correlated the LASSO might pick only one where ridge regression will average out the effect of both predictors. Zou and Hastie (2005) proposed a compromise between both penalty functions in the *elastic net*, which uses another tuning parameter, α, to mix between ℓ_1 and ℓ_2 regularization:

$$\text{Pen}_{\text{ElasticNet}}\left(\boldsymbol{\omega}_i\right) = \sum_{j=1, j!=i}^{k} \frac{1}{2}(1-\alpha)\omega_{ij}^2 + \alpha|\omega_{ij}|$$

If $\alpha = 1$, the elastic net reduces to the LASSO penalty, and if $\alpha = 0$ the elastic net reduces to the ridge penalty. When $\alpha > 0$ exact zeroes can still be obtained in the solution, and sparsity increases both with λ and α. Since moving towards ℓ_2 regularization reduces sparsity, selection of the tuning parameters using EBIC is less suited in the elastic net. Crossvalidation, however, is still capable of sketching the predictive accuracy for different values of both α and λ. Again, the *R* package *glmnet* (Friedman et al., 2010) can be used for estimating parameters using the elastic net. We have implemented a procedure to compute the Ising model for a range of λ and α values and obtain the predictive accuracy in the *R* package *elasticIsing* (Epskamp, 2016).

One issue that is currently debated is inference of regularized parameters. Since the distribution of LASSO parameters is not well-behaved (Bühlmann & van de Geer, 2011; Bühlmann, 2013), Meinshausen, Meier, and Bühlmann (2009) developed the idea of using repeated sample splitting, where in the first sample the sparse set of variables are selected, followed by multiple comparison corrected p-values in the second sample. Another interesting idea is to remove the bias introduced by regularization, upon which 'standard' procedures can be used (van de

Geer, Bühlmann, & Ritov, 2013). As a result the asymptotic distribution of the so-called de-sparsified LASSO parameters is normal with the true parameter as mean and efficient variance (i.e., achieves the Cramér-Rao bound).. Standard techniques are then applied and even confidence intervals with good coverage are obtained. The limitations here are (i) the sparsity level, which has to be $\leq \sqrt{n/\ln(P)}$, and (ii) the 'beta-min' assumption, which imposes a lower bound on the value of the smallest obtainable coefficient (Bühlmann & van de Geer, 2011).

Finally, we can use the equivalence between MIRT and the Ising model to estimate a low-rank approximation of the Ising Model. MIRT software, such as the R package *mirt* (Chalmers, 2012), can be used for this purpose. More recently, Marsman et al. (2015) have used the equivalence also presented in this chapter as a method for estimating low-rank Ising model using Full-data-information estimation. A good approximation of the Ising model can be obtained if the true Ising model is indeed low-rank, which can be checked by looking at the eigenvalue decomposition of the elastic Net approximation or by sequentially estimating the first eigenvectors through adding more latent factors in the MIRT analysis or estimating sequentially higher rank networks using the methodology of Marsman et al. (2015).

Example Analysis

To illustrate the methods described in this chapter we simulated two datasets, both with 500 measurements on 10 dichotomous scored items. The first dataset, dataset A, was simulated according to a multidimensional Rasch model, in which the first five items are determined by the first factor and the last five items by the second factor. Factor levels where sampled from a multivariate normal distribution with unit variance and a correlation of 0.5, while item difficulties where sampled from a standard normal distribution. The second dataset, dataset B, was sampled from a sparse network structure according to a Boltzmann Machine. A scale-free network was simulated using the Barabasi game algorithm (Barabási & Albert, 1999) in the R package *igraph* (Csardi & Nepusz, 2006) and a random connection probability of 5%. The edge weights where subsequently sampled from a uniform distribution between 0.75 and 1 (in line with the conception that most items in psychometrics relate positively with each other) and thresholds where sampled from a uniform distribution between -3 and -1. To simulate the responses the R package *IsingSampler* was used. The datasets where analyzed using the *elasticIsing* package in R (Epskamp, 2016); 10-fold cross-validation was used to estimate the predictive accuracy of tuning parameters λ and α on a grid of 100 logarithmically spaced λ values between 0.001 and 1 and 100 α values equally spaced between 0 and 1.

Figure 8.5 shows the results of the analyses. The left panels show the results for dataset A and the right panel shows the result for dataset B. The top panels show the negative mean squared prediction error for different values of λ and α. In both datasets, regularized models perform better than unregularized models. The plateaus on the right of the graphs show the performance of the independence graph in which all network parameters are set to zero. Dataset A obtained a maximum accuracy at $\alpha = 0$ and $\lambda = 0.201$, thus in dataset A ℓ_2-regularization

(a) (b)

(c) (d)

(e) (f)
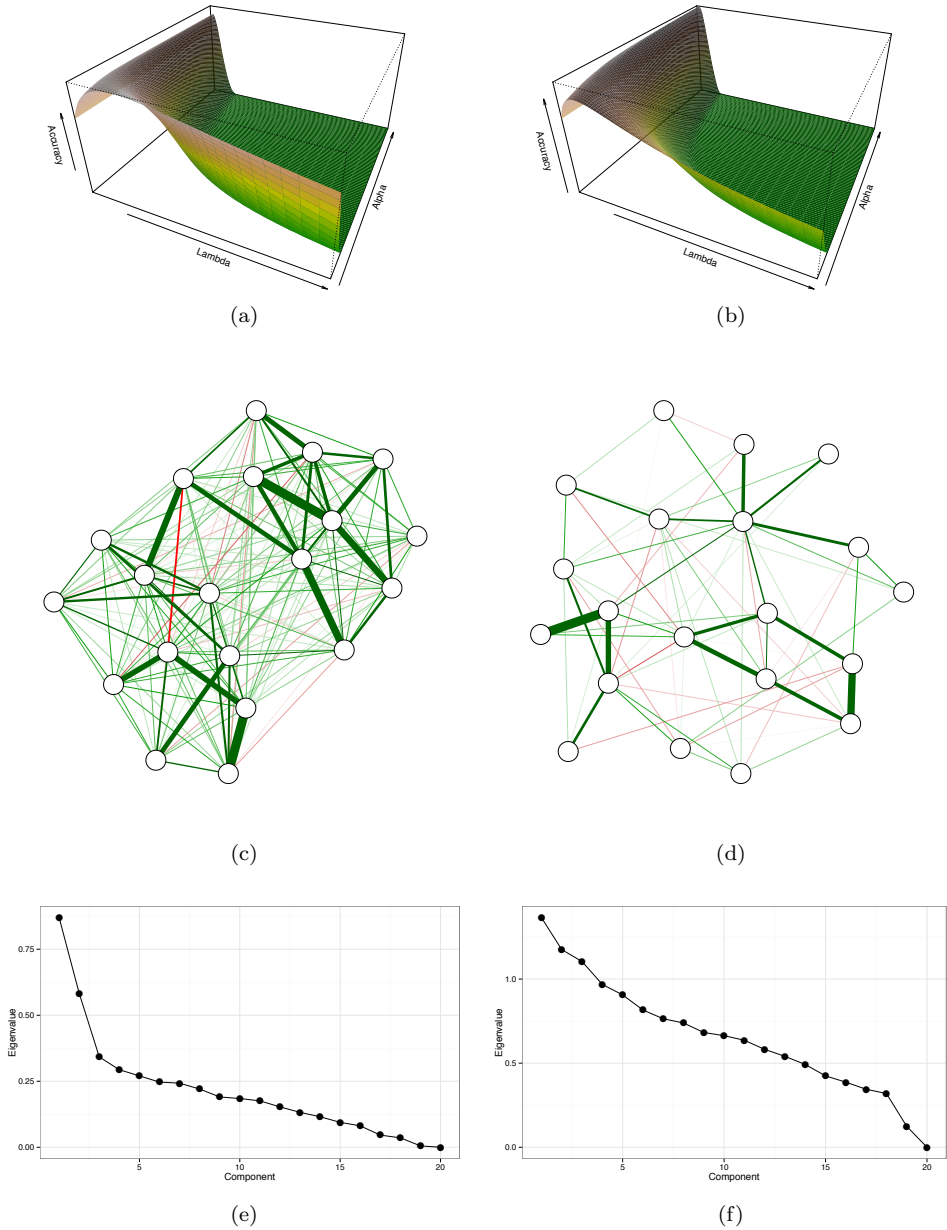
Figure 8.5: Analysis results of two simulated datasets; left panels show results based on a dataset simulated according to a 2-factor MIRT Model, while right panels show results based on a dataset simulated with a sparse scale-free network. Panels (a) and (b) show the predictive accuracy under different elastic net tuning parameters λ and α, panels (c) and (d) the estimated optimal graph structures and panels (e) and (f) the eigenvalues of these graphs.

is preferred over ℓ_1 regularization, which is to be expected since the data were simulated under a model in which none of the edge weights should equal zero. In dataset B a maximum was obtained at $\alpha = 0.960$ and $\lambda = 0.017$, indicating that in dataset B regularization close to ℓ_1 is preferred. The middle panels show visualizations of the obtained best performing networks made with the *qgraph* package (Epskamp et al., 2012); green edges represent positive weights, red edges negative weights and the wider and more saturated an edge the stronger the absolute weight. It can be seen that dataset A portrays two clusters while Dataset B portrays a sparse structure. Finally, the bottom panels show the eigenvalues of both graphs; Dataset A clearly indicates two dominant components whereas Dataset B does not indicate any dominant component.

These results show that the estimation techniques perform adequately, as expected. As discussed earlier in this chapter, the eigenvalue decomposition directly corresponds to the number of latent variables present if the common cause model is true, as is the case in dataset A. Furthermore, if the common cause model is true the resulting graph should not be sparse but low rank, as is the case in the results on dataset A.

8.5   Interpreting Latent Variables in Psychometric Models

Since Spearman's (1904) conception of general intelligence as the common determinant of observed differences in cognitive test scores, latent variables have played a central role in psychometric models. The theoretical status of the latent variable in psychometric models has been controversial and the topic of heated debates in various subfields of psychology, like those concerned with the study of intelligence (e.g., Jensen, 1998) and personality (McCrae & Costa, 2008). The pivotal issue in these debates is whether latent variables posited in statistical models have referents outside of the model; that is, the central question is whether latent variables like g in intelligence or "extraversion" in personality research refer to a property of individuals that exists independently of the model fitting exercise of the researcher (Borsboom et al., 2003; Van Der Maas et al., 2006; Cramer et al., 2010). If they do have such independent existence, then the model formulation appears to dictate a causal relation between latent and observed variables, in which the former cause the latter; after all, the latent variable has all the formal properties of a common cause because it screens off the correlation between the item responses (a property denoted local independence in the psychometric literature; Borsboom, 2005; Reichenbach, 1991). The condition of *vanishing tetrads*, that Spearman (1904) introduced as a model test for the veracity of the common factor model is currently seen as one of the hallmark conditions of the common cause model (Bollen & Lennox, 1991).

This would suggest that the latent variable model is intimately intertwined with a so-called reflective measurement model interpretation (Edwards & Bagozzi, 2000; Howell, Breivik, & Wilcox, 2007), also known as an effect indicators model (Bollen & Lennox, 1991) in which the measured attribute is represented as the cause of the test scores. This conceptualization is in keeping with causal accounts of measurement and validity (Borsboom et al., 2003; Markus & Borsboom, 2013b)

and indeed seems to fit the intuition of researchers in fields where psychometric models dominate, like personality. For example, McCrae and Costa (2008) note that they assume that extraversion causes party-going behavior, and as such this trait determines the answer to the question "do you often go to parties" in a causal fashion. Jensen (1998) offers similar ideas on the relation between intelligence and the g-factor. Also, in clinical psychology, Reise and Waller (2009, p. 26) note that "to model item responses to a clinical instrument [with IRT], a researcher must first assume that the item covariation is caused by a continuous latent variable".

However, not all researchers are convinced that a causal interpretation of the relation between latent and observed variable makes sense. For instance, McDonald (2003) notes that the interpretation is somewhat vacuous as long as no substantive theoretical of empirical identification of the latent variable can be given; a similar point is made by Borsboom and Cramer (2013). That is, as long as the sole evidence for the existence of a latent variable lies in the structure of the data to which it is fitted, the latent variable appears to have a merely statistical meaning and to grant such a statistical entity substantive meaning appears to be tantamount to overinterpreting the model. Thus, the common cause interpretation of latent variables at best enjoys mixed support.

A second interpretation of latent variables that has been put forward in the literature is one in which latent variables do not figure as common causes of the item responses, but as so-called behavior domains. Behavior domains are sets of behaviors relevant to substantive concepts like intelligence, extraversion, or cognitive ability (Mulaik & McDonald, 1978; McDonald, 2003). For instance, one can think of the behavior domain of addition as being defined through the set of all test items of the form $x + y = \ldots$. The actual items in a test are considered to be a sample from that domain. A latent variable can then be conceptualized as a so-called *tail-measure* defined on the behavior domain (Ellis & Junker, 1997). One can intuitively think of this as the total test score of a person on the infinite set of items included in the behavior domain. Ellis and Junker (1997) have shown that, if the item responses included in the domain satisfy the properties of monotonicity, positive association, and vanishing conditional independence, the latent variable can indeed be defined as a tail measure. The relation between the item responses and the latent variable is, in this case, not sensibly construed as causal, because the item responses are a part of the behavior domain; this violates the requirement, made in virtually all theories of causality, that cause and effect should be separate entities (Markus & Borsboom, 2013b). Rather, the relation between item responses and latent variable is conceptualized as a sampling relation, which means the inference from indicators to latent variable is not a species of causal inference, but of statistical generalization.

Although in some contexts the behavior domain interpretation does seem plausible, it has several theoretical shortcomings of its own. Most importantly, the model interpretation appears to beg the important explanatory question of why we observe statistical associations between item responses. For instance, Ellis and Junker (1997) manifest conditions specify that the items included in a behavior domain should look exactly as if they were generated by a common cause; in essence, the only sets of items that would qualify as behavior domains are infinite sets of items that would fit a unidimensional IRT model perfectly. The question of why

such sets would fit a unidimensional model is thus left open in this interpretation. A second problem is that the model specifies infinite behavior domains (measures on finite domains cannot be interpreted as latent variables because the axioms of Ellis and Junker will not be not satisfied in this case). In many applications, however, it is quite hard to come up with more than a few dozen of items before one starts repeating oneself (e.g., think of psychopathology symptoms or attitude items), and if one does come up with larger sets of items the unidimensionality requirement is typically violated. Even in applications that would seem to naturally suit the behavior domain interpretation, like the addition ability example given earlier, this is no trivial issue. Thus, the very property that buys the behavior domain interpretation its theoretical force (i.e., the construction of latent variables as tail measures on an infinite set of items that satisfies a unidimensional IRT model) is its substantive Achilles' heel.

Thus, the common cause interpretation of the latent variable model seems too make assumptions about the causal background of test scores that appear overly ambitious given the current scientific understanding of test scores. The behavior domain interpretation is much less demanding, but appears to be of limited use in situations where only a limited number of items is of interest and in addition offers no explanatory guidance with respect to answering the question why items hang together as they do. The network model may offer a way out of this theoretical conundrum because it specifies a third way of looking at latent variables, as explained in this chapter. As Van Der Maas et al. (2006) showed, data generated under a network model could explain the positive manifold often found in intelligence research which is often described as the g factor or general intelligence; a g factor emerged from a densely connected network even though it was not "real". This idea suggests the interpretation of latent variables as functions defined as cliques in a network of interacting components (Borsboom et al., 2011; Cramer et al., 2010; Cramer, Sluis, et al., 2012). As we have shown in this chapter, this relation between networks and latent variables is quite general: given simple models of the interaction between variables, as encoded in the Ising model, one expects data that conform to psychometric models with latent variables. The theoretical importance of this result is that (a) it allows for a model interpretation that invokes no common cause of the item responses as in the reflective model interpretation, but (b) does not require assumptions about infinite behavior domains either.

Thus, network approaches can offer a theoretical middle ground between causal and sampling interpretations of psychometric models. In a network, there clearly is nothing that corresponds to a causally effective latent variable, as posited in the reflective measurement model interpretation (Bollen & Lennox, 1991; Edwards & Bagozzi, 2000). The network model thus evades the problematic assignment of causal force to latent variables like the g-factor and extraversion. These arise out of the network structure as epiphenomena; to treat them as causes of item responses involves an unjustified reification. On the other hand, however, the latent variable model as it arises out of a network structure does not require the antecedent identification of an infinite set of response behaviors as hypothesized to exist in behavior domain theory. Networks are typically finite structures that involve a limited number of nodes engaged in a limited number of interactions. Each clique in the network structure will generate one latent variable with entirely

transparent theoretical properties and an analytically tractable distribution function. Of course, for a full interpretation of the Ising model analogous to that in physics, one has to be prepared to assume that the connections between nodes in the network signify actual interactions (i.e., they are not merely correlations); that is, connections between nodes are explicitly not spurious as they are in the reflective latent variable model, in which the causal effect of the latent variable produces the correlations between item responses. But if this assumption is granted, the theoretical status of the ensuing latent variable is transparent and may in many contexts be less problematic than the current conceptions in terms of reflective measurement models and behavior domains are.

Naturally, even though the Ising and IRT models have statistically equivalent representations, the interpretations of the model in terms of common causes and networks are not equivalent. That is, there is a substantial difference between the causal implications of a reflective latent variable model and of an Ising model. However, because for a given dataset the models are equivalent, distinguishing network models from common cause models requires the addition of (quasi-) experimental designs into the model. For example, suppose that in reality an Ising model holds for a set of variables; say we consider the depression symptoms "insomnia" and "feelings of worthlessness". The model implies that, if we were to causally intervene on the system by reducing or increasing insomnia, a change in feelings of worthlessness should ensue. In the latent variable model, in which the association between feelings of worthlessness and insomnia is entirely due to the common influence of a latent variable, an experimental intervention that changes insomnia will not be propagated through the system. In this case, the intervention variable will be associated only with insomnia, which means that the items will turn out to violate measurement invariance with respect to the intervention variable (Mellenbergh, 1989; Meredith, 1993). Thus, interventions on individual nodes in the system can propagate to other nodes in a network model, but not in a latent variable model. This is a testable implication in cases where one has experimental interventions that plausibly target a single node in the system. Fried, Nesse, Zivin, Guille, and Sen (2014) have identified a number of factors in depression that appear to work in this way.

Note that a similar argument does not necessarily work with variables that are causal consequences of the observed variables. Both in a latent variable model and in a network model, individual observed variables might have distinct outgoing effects, i.e., affect unique sets of external variables. Thus, insomnia may directly cause bags under the eyes, while feelings of worthlessness do not, without violating assumptions of either model. In the network model, this is because the outgoing effects of nodes do not play a role in the network if they do not feed back into the nodes that form the network. In the reflective model, this is because the model only speaks on the question of where the systematic variance in indicator variables comes from (i.e., this is produced by a latent variable), but not on what that systematic variance causes. As an example, one may measure the temperature of water by either putting a thermometer into the water, or by testing whether one can boil an egg in it. Both the thermometer reading and the boiled egg are plausibly construed as effects of the temperature in the water (the common cause latent variable in the system). However, only the boiled egg has the outgoing

effect of satisfying one's appetite.

In addition to experimental interventions on the elements of the system, a network model rather than a latent variable model allows one to deduce what would happen upon changing the connectivity of the system. In a reflective latent variable model, the associations between variables are a function of the effect of the latent variable and the amount of noise present in the individual variables. Thus, the only ways to change the correlation between items is by changing the effect of the latent variable (e.g., by restricting the variance in the latent variable so as to produce restriction of range effects in the observables) or by increasing noise in the observed variables (e.g., by increasing variability in the conditions under which the measurements are taken). Thus, in a standard reflective latent variable model, the connection between observed variables is purely a correlation, and one can only change it indirectly through the variable that have proper causal roles in the system (i.e., latent variables and error variables).

However, in a network model, the associations between observed variables are not spurious; they are real, causally potent pathways, and thus externally forced changes in connection strengths can be envisioned. Such changes will affect the behavior of the system in a way that can be predicted from the model structure. For example, it is well known that increasing the connectivity of an Ising model can change its behavior from being linear (in which the total number of active nodes grows proportionally to the strength of external perturbations of the system) to being highly nonlinear. Under a situation of high connectivity, an Ising network features tipping points: in this situation, very small perturbations can have catastrophic effects. To give an example, a weakly connected network of depression symptoms could only be made depressed by strong external effects (e.g., the death of a spouse), whereas a strongly connected network could tumble into a depression through small perturbations (e.g., an annoying phone call from one's mother in law). Such a vulnerable network will also feature very specific behavior; for instance, when the network is approaching a transition, it will send out early warning signals like increased autocorrelation in a time series (Scheffer et al., 2009). Recent investigations suggest that such signals are indeed present in time series of individuals close to a transition (van de Leemput et al., 2014). Latent variable models have no such consequences.

Thus, there are at least three ways in which network models and reflective latent variable models can be distinguished: through experimental manipulations of individual nodes, through experimental manipulations of connections in the network, and through investigation of the behavior of systems under highly frequent measurements that allow one to study the dynamics of the system in time series. Of course, a final and direct refutation of the network model would occur if one could empirically identify a latent variable (e.g., if one could show that the latent variable in a model for depression items was in fact identical with a property of the system that could be independently identified; say, serotonin shortage in the brain). However, such identifications of abstract psychometric latent variables with empirically identifiable common causes do not appear forthcoming. Arguably, then, psychometrics may do better to bet on network explanations of association patterns between psychometric variables than to hope for the empirical identification of latent common causes.

8.6   Conclusion

The correspondence between the Ising model and the MIRT model offers novel interpretations of long standing psychometric models, but also opens a gateway through which the psychometric can be connected to the physics literature. Although we have only begun to explore the possibilities that this connection may offer, the results are surprising and, in our view, offer a fresh look on the problems and challenges of psychometrics. In the current chapter, we have illustrated how network models could be useful in the conceptualization of psychometric data. The bridge between network models and latent variables offers research opportunities that range from model estimation to the philosophical analysis of measurement in psychology, and may very well alter our view of the foundations on which psychometric models should be built.

As we have shown, network models may yield probability distributions that are exactly equivalent to this of IRT models. This means that latent variables can receive a novel interpretation: in addition to an interpretation of latent variables as common causes of the item responses (Bollen & Lennox, 1991; Edwards & Bagozzi, 2000), or as behavior domains from which the responses are a sample (Ellis & Junker, 1997; McDonald, 2003), we can now also conceive of latent variables as mathematical abstractions that are defined on cliques of variables in a network. The extension of psychometric work to network modeling fits current developments in substantive psychology, in which network models have often been motivated by critiques of the latent variable paradigm. This has for instance happened in the context of intelligence research (Van Der Maas et al., 2006), clinical psychology (Cramer et al., 2010; Borsboom & Cramer, 2013), and personality (Cramer, Sluis, et al., 2012; Costantini, Epskamp, et al., 2015). It should be noted that, in view of the equivalence between latent variable models and network models proven here, even though these critiques may impinge on the common cause interpretation of latent variable models, they do not directly apply to latent variable models themselves. Latent variable models may in fact fit psychometric data well *because* these data result from a network of interacting components. In such a case, the latent variable should be thought of as a convenient fiction, but the latent variable model may nevertheless be useful; for instance, as we have argued in the current chapter, the MIRT model can be profitably used to estimate the parameters of a (low rank) network. Of course, the reverse holds as well: certain network structures may fit the data because cliques of connected network components result from unobserved common causes in the data. An important question is under which circumstances the equivalence between the MIRT model and the Ising model breaks down, i.e., which experimental manipulations or extended datasets could be used to decide between a common cause versus a network interpretation of the data. In the current chapter, we have offered some suggestions for further work in this direction, which we think offers considerable opportunities for psychometric progress.

As psychometrics starts to deal with network models, we think the Ising model offers a canonical form for network psychometrics, because it deals with binary data and is equivalent to well-known models from IRT. The Ising model has several intuitive interpretations: as a model for interacting components, as an asso-

ciation model with at most pairwise interactions, and as the joint distribution of response and predictor variables in a logistic regression. Especially the analogy between networks of psychometric variables (e.g., psychopathology symptoms such as depressed mood, fatigue, and concentration loss) and networks of interacting particles (e.g., as in the magnetization examples) offers suggestive possibilities for the construction of novel theoretical accounts of the relation between constructs (e.g., depression) and observables as modeled in psychometrics (e.g., symptomatology). In the current chapter, we only focused on the Ising model for binary data, but of course the work we have ignited here invites extensions in various other directions. For example, for polymotous data, the generalized Potts model could be used, although it should be noted that this model does require the response options to be discrete values that are shared over all variables, which may not suit typical psychometric applications. Another popular type of PMRF is the Gaussian Random Field (GRF; Lauritzen, 1996), which has exactly the same form as the model in (8.18) except that now $\boldsymbol{x}$ is continuous and assumed to follow a multivariate Gaussian density. This model is considerably appealing as it has a tractable normalizing constant rather than the intractable partition function of the Ising model. The inverse of the covariance matrix—the precision matrix—can be standardized as a partial correlation matrix and directly corresponds to the $\boldsymbol{\Omega}$ matrix of the Ising model. Furthermore, where the Ising model reduces to a series of logistic regressions for each node, the GRF reduces to a multiple linear regression for each node. It can easily be proven that also in the GRF the rank of the (partial) correlation matrix—cliques in the network—correspond to the latent dimensionality if the common cause model is true (Chandrasekaran et al., 2012). A great body of literature exists on estimating and fitting GRFs even when the amount of observations is limited versus the amount of nodes (Meinshausen & Bühlmann, 2006; Friedman et al., 2008; Foygel & Drton, 2010). Furthermore, promising methods are now available for the estimation of a GRF even in non-Gaussian data, provided the data are continuous (Liu et al., 2009, 2012).

8.7 Appendix A: Proof of Equivalence Between the Ising Model and MIRT

To prove the equivalence between the Ising model and MIRT, we first need to rewrite the Ising Model in matrix form:

$$p(\boldsymbol{X} = \boldsymbol{x}) = \frac{1}{Z} \exp\left(\boldsymbol{\tau}^\top \boldsymbol{x} + \frac{1}{2} \boldsymbol{x}^\top \boldsymbol{\Omega} \boldsymbol{x} \right), \tag{8.18}$$

in which $\boldsymbol{\Omega}$ is an $P \times P$ matrix containing network parameters ω_{ij} as its elements, which corresponds in graph theory to the adjacency or weights matrix. Note that, in this representation, the diagonal values of $\boldsymbol{\Omega}$ are used. However, since x_i can be only -1 or 1, $x_i^2 = 1$ for any combination, and the diagonal values are cancelled out in the normalizing constant Z. Thus, arbitrary values can be used in the diagonal of $\boldsymbol{\Omega}$. Since $\boldsymbol{\Omega}$ is a real and symmetrical matrix, we can take the usual eigenvalue decomposition:

$$\boldsymbol{\Omega} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^\top,$$

in which $\mathbf{\Lambda}$ is a diagonal matrix containing eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_P$ on its diagonal, and $\mathbf{Q}$ is an orthonormal matrix containing eigenvectors $\mathbf{q}_1, \ldots, \mathbf{q}_P$ as its columns. Inserting the eigenvalue decomposition into (8.18) gives:

$$p(\mathbf{X} = \mathbf{x}) = \frac{1}{Z} \exp\left(\sum_i \tau_i x_i \right) \prod_j \exp\left(\frac{\lambda_j}{2} \left(\sum_i q_{ij} x_i \right)^2 \right). \tag{8.19}$$

Due to the unidentified and arbitrary diagonal of $\mathbf{\Omega}$ we can force $\mathbf{\Omega}$ to be positive semi-definite—requiring all eigenvalues to be nonnegative—by shifting the eigenvalues with some constant c:

$$\mathbf{\Omega} + c\mathbf{I} = \mathbf{Q}\left(\mathbf{\Lambda} + c\mathbf{I}\right)\mathbf{Q}^\top.$$

Following the work of Kac (1966), we can use the following identity:

$$e^{y^2} = \int_{-\infty}^{\infty} \frac{e^{-2ct - t^2}}{\sqrt{\pi}} \, \mathrm{d}t,$$

with $y = \sqrt{\frac{\lambda_j}{2}\left(\sum_i q_{ij} x_i\right)^2}$ and $t = \theta_j$ to rewrite (8.19) as follows:

$$p(\mathbf{X} = \mathbf{x}) = \frac{1}{Z} \int_{-\infty}^{\infty} \frac{\exp\left(\sum_j -\theta_j^2\right)}{\sqrt{\pi^P}} \prod_i \exp\left(x_i \left(\tau_i + \sum_j -2\sqrt{\frac{\lambda_j}{2}} q_{ij} \theta_j \right) \right) \, \mathrm{d}\boldsymbol{\theta}.$$

Reparameterizing $\tau_i = -\delta_i$ and $-2\sqrt{\frac{\lambda_j}{2}} q_{ij} = \alpha_{ij}$ we obtain:

$$p(\mathbf{X} = \mathbf{x}) = \int_{-\infty}^{\infty} \frac{1}{Z} \frac{\exp\left(\sum_j -\theta_j^2\right)}{\sqrt{\pi^P}} \prod_i \exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i \right) \right) \, \mathrm{d}\boldsymbol{\theta}. \tag{8.20}$$

The same transformations can be used to obtain a different expression for Z:

$$Z = \int_{-\infty}^{\infty} \frac{\exp\left(\sum_j -\theta_j^2\right)}{\sqrt{\pi^P}} \sum_{\mathbf{x}} \prod_i \exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i \right) \right) \, \mathrm{d}\boldsymbol{\theta}$$

$$= \int_{-\infty}^{\infty} \frac{\exp\left(\sum_j -\theta_j^2\right)}{\sqrt{\pi^P}} \prod_i \sum_{x_i} \exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i \right) \right) \, \mathrm{d}\boldsymbol{\theta}. \tag{8.21}$$

Finally, inserting (8.21) into (8.20), multiplying by $\frac{\prod_i \sum_{x_i} \exp\left(x_i\left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i\right)\right)}{\prod_i \sum_{x_i} \exp\left(x_i\left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i\right)\right)}$, and rearranging gives:

$$p(\mathbf{X} = \mathbf{x}) = \int_{-\infty}^{\infty} \frac{\frac{\exp\left(\sum_j -\theta_j^2\right)}{\sqrt{\pi^P}} \prod_i \sum_{x_i} \exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i \right) \right)}{\int_{-\infty}^{\infty} \frac{\exp\left(\sum_j -\theta_j^2\right)}{\sqrt{\pi^P}} \prod_i \sum_{x_i} \exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i \right) \right) \, \mathrm{d}\boldsymbol{\theta}}$$

$$\cdot \prod_i \frac{\exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i \right) \right)}{\sum_{x_i} \exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i \right) \right)} \, \mathrm{d}\boldsymbol{\theta}. \tag{8.22}$$

The first part of the integral on the right hand side of (8.22) corresponds to a distribution that sums to 1 for a P-dimensional random vector $\boldsymbol{\Theta}$:

$$f(\boldsymbol{\theta}) \propto \frac{\exp\left(\sum_j -\theta_j^2\right)}{\sqrt{\pi^P}} \prod_i \sum_{x_i} \exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i\right)\right),$$

and the second part corresponds to the 2-parameter logistic MIRT probability of the response vector as in (8.13):

$$P(\boldsymbol{X} = \boldsymbol{x} \mid \boldsymbol{\Theta} = \boldsymbol{\theta}) = \prod_i \frac{\exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i\right)\right)}{\sum_{x_i} \exp\left(x_i \left(\boldsymbol{\alpha}_i^\top \boldsymbol{\theta} - \delta_i\right)\right)}.$$

We can look further at this distribution by using Bayes' rule to examine the conditional distribution of $\boldsymbol{\theta}$ given $\boldsymbol{X} = \boldsymbol{x}$:

$$f(\boldsymbol{\theta} \mid \boldsymbol{X} = \boldsymbol{x}) \propto \Pr\left(\boldsymbol{X} = \boldsymbol{x} \mid \boldsymbol{\Theta} = \boldsymbol{\theta}\right) f\left(\boldsymbol{\theta}\right)$$

$$\propto \exp\left(\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{\theta} - \boldsymbol{\theta}^\top \boldsymbol{\theta}\right)$$

$$\propto \exp\left(-\frac{1}{2}\left(\boldsymbol{\theta} - \frac{1}{2}\boldsymbol{A}^\top \boldsymbol{x}\right)^\top 2\boldsymbol{I}\left(\boldsymbol{\theta} - \frac{1}{2}\boldsymbol{A}^\top \boldsymbol{x}\right)\right)$$

and see that the posterior distribution of $\boldsymbol{\Theta}$ is a multivariate Gaussian distribution:

$$\boldsymbol{\Theta} \mid \boldsymbol{X} = \boldsymbol{x} \sim N_P\left(\pm\frac{1}{2}\boldsymbol{A}^\top \boldsymbol{x}, \sqrt{\frac{1}{2}}\boldsymbol{I}\right), \tag{8.23}$$

in which $\boldsymbol{A}$ is a matrix containing the discrimination parameters $\boldsymbol{\alpha}_i$ as its rows and $\pm$ indicates that columns $\boldsymbol{a}_j$ could be multiplied with -1 due to that both the positive and negative root can be used in $\sqrt{\frac{\lambda_j}{2}}$, simply indicating whether the items overall are positively or negatively influenced by the latent trait $\boldsymbol{\theta}$. Additionally, Since the variance–covariance matrix of $\boldsymbol{\theta}$ equals zero in all nondiagonal elements, $\boldsymbol{\theta}$ is orthogonal. Thus, the multivariate density can be decomposed as the product of univariate densities:

$$\Theta_j \mid \boldsymbol{X} = \boldsymbol{x} \sim N\left(\pm\frac{1}{2}\sum_i a_{ij} x_i, \sqrt{\frac{1}{2}}\right).$$

8.8 Appendix B: Glossary of Notation

Symbol	Dimension	Description
$\{\ldots\}$		Set of distinct values.
(a, b)		Interval between a and b.
P	$\mathbb{N}$	Number of variables.
N	$\mathbb{N}$	Number of observations.
$\boldsymbol{X}$	$\{-1, 1\}^P$	Random vector of binary variables.
$\boldsymbol{x}$	$\{-1, 1\}^P$	A possible realization of $\boldsymbol{X}$.
$n(\boldsymbol{x})$	$\mathbb{N}$	Number of observations with response pattern $\boldsymbol{x}$.
i, j, k and l	$\{1, 2, \ldots, P\}, j \neq i$	Subscripts of random variables.
$\boldsymbol{X}^{-(i)}$	$\{-1, 1\}^{P-1}$	Random vector of binary variables without X_i.
$\boldsymbol{x}^{-(i)}$	$\{-1, 1\}^{P-1}$	A possible realization of $\boldsymbol{X}^{-(i)}$.
$\boldsymbol{X}^{-(i,j)}$	$\{-1, 1\}^{P-2}$	Random vector of binary variables without X_i and X_j.
$\boldsymbol{x}^{-(i,j)}$	$\{-1, 1\}^{P-2}$	A possible realization of $\boldsymbol{X}^{-(i)}$.
$\Pr(\ldots)$	$\to (0, 1)$	Probability function.
$\phi_i(x_i)$	$\{-1, 1\} \to \mathbb{R}_{>0}$	Node potential function.
$\phi_i(x_i, x_j)$	$\{-1, 1\}^2 \to \mathbb{R}_{>0}$	Pairwise potential function.
τ_i	$\mathbb{R}$	Threshold parameter for node X_i in the Ising model. Defined as $\tau_i = \ln \phi_i(1)$.
$\boldsymbol{\tau}$	$\mathbb{R}^P$	Vector of threshold parameters, containing τ_i as its ith element.
ω_{ij}	$\mathbb{R}$	Network parameter between nodes X_i and X_j in the Ising model. Defined as $\omega_{ij} = \ln \phi_{ij}(1, 1)$.
$\boldsymbol{\Omega}$	$\mathbb{R}^{P \times P}$ and symmetrical	Matrix of network parameters, containing ω_{ij} as its ijth element.
$\boldsymbol{\omega}_i$	$\mathbb{R}^P$	The ith row or column of $\boldsymbol{\Omega}$.
$\text{Pen}(\boldsymbol{\omega}_i)$	$\mathbb{R}^P \to \mathbb{R}$	Penalization function of $\boldsymbol{\omega}_i$.
β	$\mathbb{R}_{>0}$	Inverse temperature in the Ising model.
$H(\boldsymbol{x})$	$\{-1, 1\}^P \to \mathbb{R}$	Hamiltonian function denoting the energy of state $\boldsymbol{x}$ in the Ising model.
$\nu_{\ldots}(\ldots)$	$\to \mathbb{R}$	The log potential functions, used in loglinear analysis.
M	$\mathbb{N}$	The number of latent factors.
$\boldsymbol{\Theta}$	$\mathbb{R}^M$	Random vector of continuous latent variables.
$\boldsymbol{\theta}$	$\mathbb{R}^M$	Realization of $\boldsymbol{\Theta}$.
$\mathcal{L}(\boldsymbol{\tau}, \boldsymbol{\Omega}; \boldsymbol{x})$	$\to \mathbb{R}$	Likelihood function based on $\Pr(\boldsymbol{X} = \boldsymbol{x})$.
$\mathcal{L}_i(\boldsymbol{\tau}, \boldsymbol{\Omega}; \boldsymbol{x})$	$\to \mathbb{R}$	Likelihood function based on $\Pr\left(X_i = x_i \mid \boldsymbol{X}^{-(i)} = \boldsymbol{x}^{-(i)}\right)$.
λ	$\mathbb{R}_{>0}$	LASSO tuning parameter
α	$(0, 1)$	Elastic net tuning parameter

Part III

Visualizing Psychometrics and Personality Research

Network Visualizations of Relationships in Psychometric Data

Abstract

We present the *qgraph* package for R, which provides an interface to visualize data through network modeling techniques. For instance, a correlation matrix can be represented as a network in which each variable is a node and each correlation an edge; by varying the width of the edges according to the magnitude of the correlation, the structure of the correlation matrix can be visualized. A wide variety of matrices that are used in statistics can be represented in this fashion, for example matrices that contain (implied) covariances, factor loadings, regression parameters and p values. *qgraph* can also be used as a psychometric tool, as it performs exploratory and confirmatory factor analysis, using *sem* (Fox, 2006) and *lavaan* (Rosseel, 2012); the output of these packages is automatically visualized in *qgraph*, which may aid the interpretation of results. In this article, we introduce *qgraph* by applying the package functions to data from the NEO-PI-R, a widely used personality questionnaire.

9.1 Introduction

The human visual system is capable of processing highly dimensional information naturally. For instance, we can immediately spot suggestive patterns in a scatterplot, while these same patterns are invisible when the data is numerically represented in a matrix.

We present *qgraph*[1], an R package that accommodates this capacity for spotting patterns by visualizing data in a novel way: through networks. Networks consist

---

This chapter has been adapted from: Epskamp, S., Cramer, A.O.J., Waldorp, L.J., Schmittmann, V.D., and Borsboom, D. (2012). qgraph: Network Visualizations of Relationships in Psychometric Data. *Journal of Statistical Software, 48* (1), 1–18.

[1] http://cran.r-project.org/web/packages/qgraph/index.html

of nodes (also called 'vertices') that are connected by edges (Harary, 1969). Each edge has a certain weight, indicating the strength of the relevant connection, and in addition edges may or may not be directed. In most applications of network modeling, nodes represent entities (e.g., people in social networks, or genes in gene networks). However, in statistical analysis it is natural to represent variables as nodes. This representation has a longstanding tradition in econometrics and psychometrics (e.g., see Bollen & Lennox, 1991; Edwards & Bagozzi, 2000), and was a driving force behind the development of graphical models for causal analysis (Spirtes, Glymour, & Scheines, 2000; Pearl, 2000). By representing relationships between variables (e.g., correlations) as weighted edges important structures can be detected that are hard to extract by other means. In general, *qgraph* enables the researcher to represent complex statistical patterns in clear pictures, without the need for data reduction methods.

qgraph was developed in the context of network approaches to psychometrics (Cramer et al., 2010; Borsboom, 2008; Schmittmann et al., 2013), in which theoretical constructs in psychology are hypothesized to be networks of causally coupled variables. In particular, *qgraph* automates the production of graphs such as those proposed in Cramer et al. (2010). However, the techniques in the package have also proved useful as a more general tool for visualizing data, and include methods to visualize output from several psychometric packages like *sem* (Fox, 2006) and *lavaan* (Rosseel, 2012).

A number of R packages can be used for the visualization and analysis of networks (e.g., *network*, Butts, Handcock, & Hunter, 2011; *statnet* Handcock, Hunter, Butts, Goodreau, & Morris, 2008; *igraph* Csardi & Nepusz, 2006). In visualizing graphs *qgraph* distinguishes itself by being specifically aimed at the visualization of statistical information. This usually leads to a special type of graph: a non-sparse weighted graph. Such graphs typically contain many edges (e.g., a fully connected network with 50 nodes has 2450 edges) thereby making it hard to interpret the graph; as well as inflating the file size of vector type image files (e.g., PDF, SVG, EPS). *qgraph* is specifically aimed at presenting such graphs in a meaningful way (e.g., by using automatic scaling of color and width, cutoff scores and ordered plotting of edges) and to minimize the file size of the output in vector type image files (e.g., by minimizing the amount of polygons needed). Furthermore, *qgraph* is designed to be usable by researchers new to R, while at the same time offering more advanced customization options for experienced R users.

qgraph is not designed for numerical analysis of graphs (Boccaletti, Latora, Moreno, Chavez, & Hwang, 2006), but can be used to compute the node centrality measures of weighted graphs proposed by Opsahl et al. (2010). Other R packages as well as external software can be used for more detailed analyses. *qgraph* facilitates these methods by using commonly used methods as input. In particular, the input is the same as used in the *igraph* package for R, which can be used for many different analyses.

In this article we introduce *qgraph* using an included dataset on personality traits. We describe how to visualize correlational structures, factor loadings and structural equation models and how these visualizations should be interpreted. Finally we will show how *qgraph* can be used as a simple unified interface to perform several exploratory and confirmatory factor analysis routines available in

R.

9.2 Creating Graphs

Throughout this article we will be working with a dataset concerning the five factor model of personality (Benet-Martinez & John, 1998; Digman, 1989; Goldberg, 1990a, 1993; McCrae & Costa, 1997). This is a model in which correlations between responses to personality items (i.e., questions of the type 'do you like parties?', 'do you enjoy working hard?') are explained by individual differences in five personality traits: *neuroticism*, *extraversion*, *agreeableness*, *openness to experience* and *conscientiousness*. These traits are also known as the 'Big Five'. We use an existing dataset in which the Dutch translation of a commonly used personality test, the NEO-PI-R (Costa & McCrae, 1992; Hoekstra, de Fruyt, & Ormel, 2003), was administered to 500 first year psychology students (Dolan, Oort, Stoel, & Wicherts, 2009). The NEO-PI-R consists of 240 items designed to measure the five personality factors with items that cover six facets per factor[2]. The scores of each subject on each item are included in *qgraph*, as well as information on the factor each item is designed to measure (this information is in the column names). All graphs in this chapter were made using R version 2.14.1 (2011-12-22) and *qgraph* version 1.0.0.

First, we load *qgraph* and the NEO-PI-R dataset:

```
library("qgraph")
data("big5")
```

Input Modes

The main function of *qgraph* is called `qgraph()`, and its first argument is used as input for making the graph. This is the only mandatory argument and can either be a weights matrix, an edge-list or an object of class `"qgraph"`, `"loadings"` and `"factanal"` (*stats*; R Core Team, 2016), `"principal"` (*psych*; Revelle, 2010), `"sem"` and `"semmod"` (*sem*; Fox, 2006), `"lavaan"` (*lavaan*; Rosseel, 2012), `"graph-NEL"` (*Rgraphviz*; Gentry et al., 2011) or `"pcAlgo"` (*pcalg*; Kalisch, Maechler, & Colombo, 2010). In this chapter we focus mainly on *weights matrices*, information on other input modes can be found in the documentation.

A weights matrix codes the connectivity structure between nodes in a network in matrix form. For a graph with n nodes its weights matrix $\boldsymbol{A}$ is a square n by n matrix in which element a_{ij} represents the strength of the connection, or weight, from node i to node j. Any value can be used as weight as long as (a) the value zero represents the absence of a connection, and (b) the strength of connections is symmetric around zero (so that equal positive and negative values are comparable in strength). By default, if $\boldsymbol{A}$ is symmetric an undirected graph is plotted and otherwise a directed graph is plotted. In the special case where all edge weights are either 0 or 1 the weights matrix is interpreted as an adjacency matrix and an unweighted graph is made.

---

[2]A facet is a subdomain of the personality factor; e.g., the factor neuroticism has depression and anxiety among its subdomains.
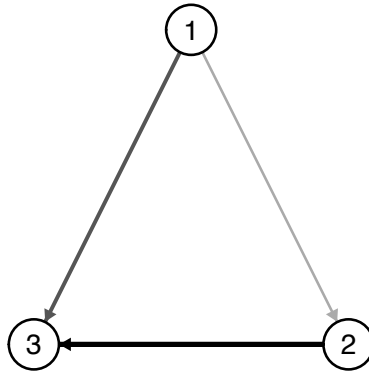
Figure 9.1: A directed graph based on a 3 by 3 weights matrix with three edges of different strengths.

For example, consider the following weights matrix:

$$\begin{bmatrix} 0 & 1 & 2 \\ 0 & 0 & 3 \\ 0 & 0 & 0 \end{bmatrix}$$

This matrix represents a graph with 3 nodes with weighted edges from node 1 to nodes 2 and 3, and from node 2 to node 3. The resulting graph is presented in Figure 9.1.

Many statistics follow these rules and can be used as edge weights (e.g., correlations, covariances, regression parameters, factor loadings, log odds). Weights matrices themselves also occur naturally (e.g., as a correlation matrix) or can easily be computed. Taking a correlation matrix as the argument of the function `qgraph()` is a good start to get acquainted with the package.

With the NEO-PI-R dataset, the correlation matrix can be plotted with:

```
qgraph(cor(big5))
```

This returns the most basic graph, in which the nodes are placed in a circle. The edges between nodes are colored according to the sign of the correlation (green for positive correlations, and red for negative correlations), and the thickness of the edges represents the absolute magnitude of the correlation (i.e., thicker edges represent higher correlations).

Visualizations that aid data interpretation (e.g., are items that supposedly measure the same construct closely connected?) can be obtained either by using the `groups` argument, which groups nodes according to a criterion (e.g., being in the same psychometric subtest) or by using a layout that is sensitive to the correlation structure. First, the `groups` argument can be used to specify which nodes belong together (e.g., are designed to measure the same factor). Nodes belonging together have the same color, and are placed together in smaller circles. The `groups` argument can be used in two ways. First, it can be a list in which each
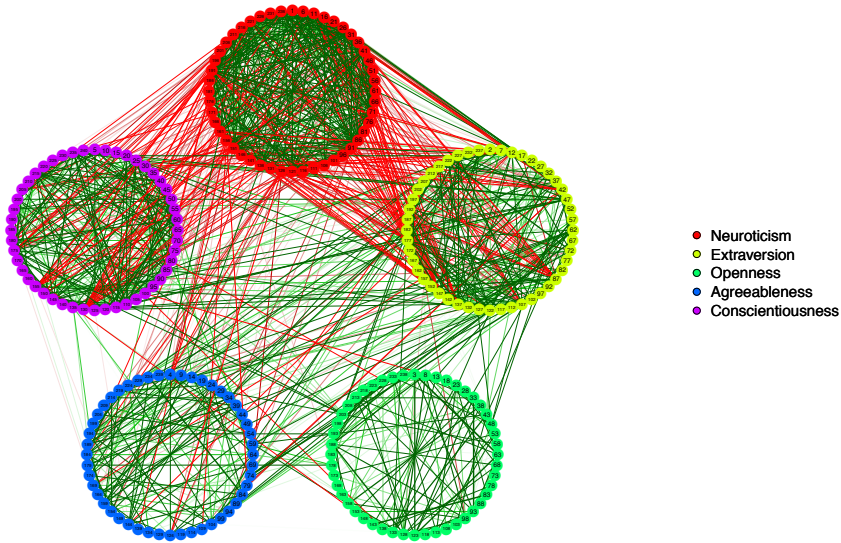
Figure 9.2: A visualization of the correlation matrix of the NEO-PI-R dataset. Each node represents an item and each edge represents a correlation between two items. Green edges indicate positive correlations, red edges indicate negative correlations, and the width and color of the edges correspond to the absolute value of the correlations: the higher the correlation, the thicker and more saturated is the edge.

element is a vector containing the numbers of nodes belonging together. Secondly, it can be a factor in which the levels belong together. The names of the elements in the list or the levels in the factor are used in a legend of requested.

For the Big 5 dataset, the grouping of the variables according to the NEO-PI-R manual is included in the package. The result of using the `groups` argument is a network representation that readily facilitates interpretation in terms of the five personality factors:

```
data("big5groups")
Q <- qgraph(cor(big5), groups = big5groups)
```

Note that we saved the graph in the object `Q`, to avoid specifying these arguments again in future runs. It is easy to subsequently add other arguments: for instance, we may further optimize the representation by using the `minimum` argument to omit correlations we are not interested in (e.g., very weak correlations), `borders` to omit borders around the nodes, and `vsize` to make the nodes smaller:

```
Q <- qgraph(Q, minimum = 0.25, borders = FALSE, vsize = 2)
```

The resulting graph is represented in Figure 9.2.

Layout Modes

Instead of predefined circles (as was used in Figure 9.2), an alternative way of facilitating interpretations of correlation matrices is to let the placement of the nodes be a function of the pattern of correlations. Placement of the nodes can be controlled with the `layout` argument. If `layout` is assigned `"circular"`, then the nodes are placed clockwise in a circle, or in smaller circles if the `groups` argument is specified (as in Figure 9.2). If the nodes are placed such that the length of the edges depends on the strength of the edge weights (i.e., shorter edges for stronger weights), then a picture can be generated that shows how variables cluster. This is a powerful exploratory tool, that may be used as a visual analogue to factor analysis. To make the length of edges directly correspond to the edge weights an high dimensional space would be needed, but a good alternative is the use of force-embedded algorithms (Di Battista, Eades, Tamassia, & Tollis, 1994) that iteratively compute the layout in two-dimensional space.

A modified version of the Fruchterman and Reingold (1991) algorithm is included in *qgraph*. This is a C function that was ported from the *sna* package (Butts, 2010). A modification of this algorithm for weighted graphs was taken from *igraph* (Csardi & Nepusz, 2006). This algorithm uses an iterative process to compute a layout in which the length of edges depends on the absolute weight of the edges. To use the Fruchterman-Reingold algorithm in `qgraph()` the `layout` argument needs to be set to `"spring"`. We can do this for the NEO-PI-R dataset, using the graph object `Q` that we defined earlier, and omitting the legend:

```
qgraph(Q, layout = "spring", legend = FALSE)
```

Figure 9.3 shows the correlation matrix of the Big Five dataset with the nodes placed according to the Fruchterman-Reingold algorithm. This allows us inspect the clustering of the variables. The figure shows interesting structures that are far harder to detect with conventional analyses. For instance, neuroticism items (i.e., red nodes) cluster to a greater extent when compared to other traits; especially openness is less strongly organized than the other factors. In addition, agreeableness and extraversion items are literally intertwined, which offers a suggestive way of thinking about the well known correlation between these traits.

The placement of the nodes can also be specified manually by assigning the `layout` argument a matrix containing the coordinates of each node. For a graph of n nodes this would be a n by 2 matrix in which the first column contains the x coordinates and the second column contains the y coordinates. These coordinates can be on any scale and will be rescaled to fit the graph by default. For example, the following matrix describes the coordinates of the graph in Figure 9.1:

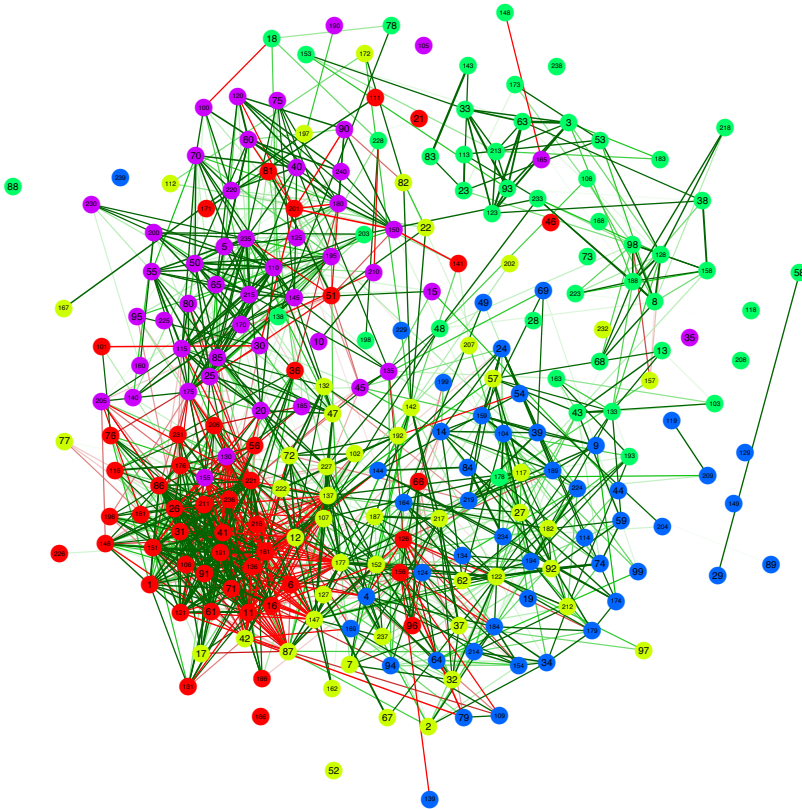$$\begin{bmatrix} 2 & 2 \\ 3 & 1 \\ 1 & 1 \end{bmatrix}$$

Figure 9.3: A graph of the correlation matrix of the NEO-PI-R dataset in which the nodes are placed by the Fruchterman-Reingold algorithm. The specification of the nodes and edges are identical to Figure 9.2

This method of specifying the layout of a graph is identical to the one used in the *igraph* (Csardi & Nepusz, 2006) package, and thus any layout obtained through *igraph* can be used[3].

One might be interested in creating not one graph but an animation of several graphs that are similar to each other. Such animations can, for example, illustrate the growth of a network over time or show the change of correlational structures in repeated measures. For such similar but not equal graphs the Fruchterman-Reingold algorithm might return completely different layouts, which will make the animation unnecessary hard to interpret. This problem can be solved by limiting the amount of space a node may move in each iteration. The function `qgraph.animate()` automates this process and can be used for various types of animations.

Output Modes

To save the graphs, any output device in R can be used to obtain high resolution, publication-ready image files. Some devices can be called directly by `qgraph()` through the `filetype` argument, which must be assigned a string indicating what device should be used. Currently `filetype` can be `"R"` or `"x11"`[4] to open a new plot in R, raster types `"tiff"`, `"png"` and `"jpg"`, vector types `"eps"`, `"pdf"` and `"svg"` and `"tex"`. A *PDF* file is advised, and this can thus be created with `qgraph(\ldots, filetype = "pdf")`.

Often, the number of nodes makes it potentially hard to track which variables are represented by which nodes. To address this problem, one can define mouseover tooltips for each node, so that the name of the corresponding variable (e.g., the item content in the Big Five graph) is shown when one places the cursor over the relevant node. In *qgraph*, mouseover tooltips can be placed on the nodes in two ways. The `"svg"` `filetype` creates a *SVG* image using the *RSVGTipsDevice* package (Plate, 2009)[5]. This filetype can be opened using most browsers (best viewed in Firefox) and can be used to include mouseover tooltips on the node labels. The `tooltips` argument can be given a vector containing the tooltip for each node. Another option for mouseover tooltips is to use the `"tex"` `filetype`. This uses the *tikzDevice* package (Sharpsteen & Bracken, 2010) to create a `.tex` file that contains the graph[6], which can then be built using pdfLATEX. The `tooltips` argument can also be used here to create mouseover tool tips in a PDF file[7].

---

[3]To do this, first create an `"igraph"` object by calling `graph.adjacency()` on the weights matrix with the arguments= `weighted=TRUE`. Then, use one of the layout functions (e.g., `layout.spring()`) on the `"igraph"` object. This returns the matrix with coordinates which can be used in `qgraph()`

[4]*RStudio* users are advised to use `filetype="x11"` to plot in R

[5]*RSVGTipsDevice* is only available for 32bit versions of R

[6]Note that this will load the *tikzdevice* package which upon loading checks for a LATEX compiler. If this is not available the package might fail to load

[7]We would like to thank Charlie Sharpsteen for supplying the tikz codes for these tooltips

Standard visual parameters

In weighted graphs green edges indicate positive weights and red edges indicate negative weights[8]. The color saturation and the width of the edges corresponds to the absolute weight and scale relative to the strongest weight in the graph (i.e., the edge with the highest absolute weight will have full color saturation and be the widest). It is possible to control this behavior by using the `maximum` argument: when `maximum` is set to a value above any absolute weight in the graph then the color and width will scale to the value of `maximum` instead[9]. Edges with an absolute value under the `minimum` argument are omitted, which is useful to keep filesizes from inflating in very large graphs.

In larger graphs the above edge settings can become hard to interpret. With the `cut` argument a cutoff value can be set which splits scaling of color and width. This makes the graphs much easier to interpret as you can see important edges and general trends in the same picture. Edges with absolute weights under the cutoff score will have the smallest width and become more colorful as they approach the cutoff score, and edges with absolute weights over the cutoff score will be full red or green and become wider the stronger they are.

In addition to these standard arguments there are several arguments that can be used to graphically enhance the graphs to, for example, change the size and shape of nodes, add a background or Venn diagram like overlay and visualize test scores of a subject on the graph. The documentation of the `qgraph()` function has detailed instructions and examples on how these can be used.

9.3 Visualizing Statistics as Graphs

Correlation Matrices

In addition to the representations in Figures 9.2 and 9.3, *qgraph* offers various other possibilities for visualizing association structures. If a correlation matrix is used as input, the `graph` argument of `qgraph()` can be used to indicate what *type* of graph should be made. By default this is `"association"` in which correlations are used as edge weights (as in Figures 9.2 and 9.3).

Another option is to assign `"concentration"` to `graph`, which will create a graph in which each edge represents the partial correlation between two nodes: partialling out all other variables. For normally distributed continuous variables, the partial correlation can be obtained from the inverse of the correlation (or covariance) matrix. If $\boldsymbol{P}$ is the inverse of the correlation matrix, then the partial correlation ω_{ij} of variables i and j is given by:

$$\omega_{ij} = \frac{-p_{ij}}{\sqrt{p_{ii}p_{jj}}}$$

Strong edges in a resulting concentration graph indicate correlations between variables that cannot be explained by other variables in the network, and are therefore indicative of causal relationships (e.g., a real relationship between smoking

---

[8] The edge colors can currently not be changed except to grayscale colors using `gray=TRUE`
[9] This must be done to compare different graphs; a good value for correlations is 1

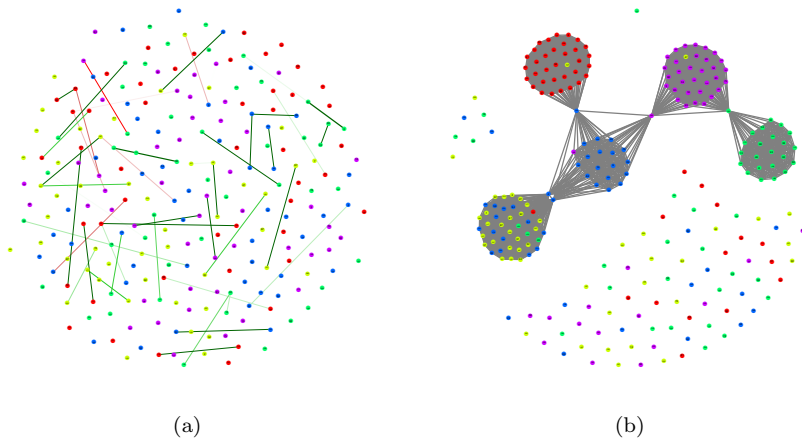(a)   (b)

Figure 9.4: Additional visualizations based on the correlations of the NEO-PI-R dataset. Panel (a) shows a concentration graph with partial correlations and panel (b) shows a graph in which connections are based on an exploratory factor analysis.

and lung cancer that cannot be explained by other factors, for example gender), provided that all relevant variables are included in the network.

The left panel of Figure 9.4 shows a concentration graph for the NEO-PI-R dataset. This graph shows quite a few partial correlations above 0.3. Within the factor model, these typically indicate violations of local independence and/or simple structure.

A third option for the `graph` argument is `"factorial"`, which creates an unweighted graph based on an exploratory factor analysis on the correlation (or covariance) matrix with promax rotation (using `factanal()` from *stats*). By default the number of factors that are extracted in the analysis is equal to the number of eigenvalues over 1 or the number of groups in the `groups` argument (if supplied). Variables are connected with an edge if they both load higher on the same factor than the cutoff score that is set with the `cut` argument. As such, the `"factorial"` option can be used to investigate the structure as detected with exploratory factor analysis.

The right panel of Figure 9.4 shows the factorial graph of the NEO-PI-R dataset. This graph shows five clusters, as expected, but also displays some overlap between the extraversion and agreeableness items.

qgraph has two functions that are designed to make all of the above graphs in a single run. The first option is to use the `qgraph.panel()` function to create a four-panel plot that contains the association graph with circular and spring layouts, a concentration graph with the spring layout, and a factorial graph with the spring layout. We can apply this function to the Big Five data to obtain all graphs at once:

```
qgraph.panel(cor(big5), groups = big5groups, minimum = 0.25,
    borders = FALSE, vsize = 1, cut = 0.3)
```

A second option to represent multiple graphs at once is to use the `qgraph.svg()` function to produce an interactive graph. This function uses *RSVGTipsDevice* (only available for 32bit versions of R; Plate, 2009) to create a host of SVG files for all three types of graphs, using circular and spring layouts and different cutoff scores. These files contain hyperlinks to each other (which can also be used to show the current graph in the layout of another graph) and can contain mouseover tool tips as well. This can be a useful interface to quickly explore the data. A function that does the same in `tex` format will be included in a later version of *qgraph*, which can then be used to create a multi-page pdf file containing the same graphs as `qgraph.panel()`.

Matrices that are similar to correlation matrices, like covariance matrices and lag-1 correlations in time series, can also be represented in *qgraph*. If the matrix is not symmetric (as is for instance the case for lag-1 correlations) then a directed graph is produced. If the matrix has values on the diagonal (e.g., a covariance matrix) these will be omitted by default. To show the diagonal values the `diag` argument can be used. This can be set to `TRUE` to include edges from and to the same node, or `"col"` to color the nodes according to the strength of diagonal entries. Note that it is advisable to only use standardized statistics (e.g., correlations instead of covariances) because otherwise the graphs can become hard to interpret.

Significance

Often a researcher is interested in the significance of a statistic (p value) rather than the value of the statistic itself. Due to the strict cutoff nature of significance levels, the usual representation then might not be adequate because small differences (e.g., the difference between edges based on t statistics of 1.9 and 2) are hard to see.

In *qgraph* statistical significance can be visualized with a weights matrix that contains p values and assigning `"sig"` to the `mode` argument. Because these values are structurally different from the edge weights we have used so far, they are first transformed according to the following function:

$$w_i = 0.7(1 - p_i)^{\log_{0.95} \frac{0.4}{0.7}}$$

where w_i is the edge weight of edge i and p_i the corresponding significance level. The resulting graph shows different levels of significance in different shades of blue, and omits any insignificant value. The levels of significance can be specified with the `alpha` argument. For a black and white representations, the `gray` argument can be set to `TRUE`.

For correlation matrices the *fdrtool* package (Strimmer., 2011) can be used to compute p values of a given set of correlations. Using a correlation matrix as input the `graph` argument should be set to `"sig"`, in which case the p values are computed and a graph is created as if `mode="sig"` was used. For the Big 5 data, a significance graph can be produced through the following code:

```
qgraph(cor(big5), groups = big5groups, vsize = 2,
       graph = "sig", alpha = c(1e-04, 0.001, 0.01))
```

Factor Loadings

A factor-loadings matrix contains the loadings of each item on a set of factors obtained through factor analysis. Typical ways of visualizing such a matrix is to boldface factor loadings that exceed, or omit factor loadings below, a given cutoff score. With such a method smaller, but interesting, loadings might easily be overlooked. In *qgraph*, factor-loading matrices can be visualized in a similar way as correlation matrices: by using the factor loadings as edge weights in a network. The function for this is |qgraph.loadings()— which uses the factor-loadings matrix to create a weights matrix and a proper layout and sends that information to `qgraph()`.

There are two functions in *qgraph* that perform an exploratory analysis based on a supplied correlation (or covariance) matrix and send the results to |qgraph-.loadings()—. The first is `qgraph.efa()` which performs an exploratory factor analysis (EFA; Stevens, 1996) using `factanal()` (*stats*; R Core Team, 2016). This function requires three arguments plus any additional argument that will be sent to |qgraph.loadings()— and `qgraph()`. The first argument must be a correlation or covariance matrix, the second the number of factors to be extracted and the third the desired rotation method.

To perform an EFA on the Big 5 dataset we can use the following code:

```
qgraph.efa(big5, 5, groups = big5groups, rotation = "promax",
    minimum = 0.2, cut = 0.4, vsize = c(1, 15),
    borders = FALSE, asize = 0.07, esize = 4, vTrans = 200)
```

Note that we supplied the `groups` list and that we specified a *promax* rotation allowing the factors to be correlated.

The resulting graph is shown in the left panel of Figure 9.5. The factors are placed in an inner circle with the variables in an outer circle around the factors[10]. The factors are placed clockwise in the same order as the columns in the loadings matrix, and the variables are placed near the factor they load the highest on. Because an EFA is a *reflective measurement model*, the arrows point towards the variables and the graph has residuals (Bollen & Lennox, 1991; Edwards & Bagozzi, 2000).

The left panel of Figure 9.5 shows that the Big 5 dataset roughly conforms to the 5 factor model. That is, most variables in each group of items tend to load on the same factor. However, we also see many crossloadings, indicating departures from simple structure. Neuroticism seems to be a strong factor, and most crossloadings are between extraversion and agreeableness.

The second function that performs an analysis and sends the results to |qgraph-.loadings()— is `qgraph.pca()`. This function performs a principal component analysis (PCA; Jolliffe, 2002) using `princomp()` of the *psych* package (Revelle, 2010). A PCA differs from an EFA in that it uses a *formative measurement model*

---

[10]For a more traditional layout we could set `layout="tree"`
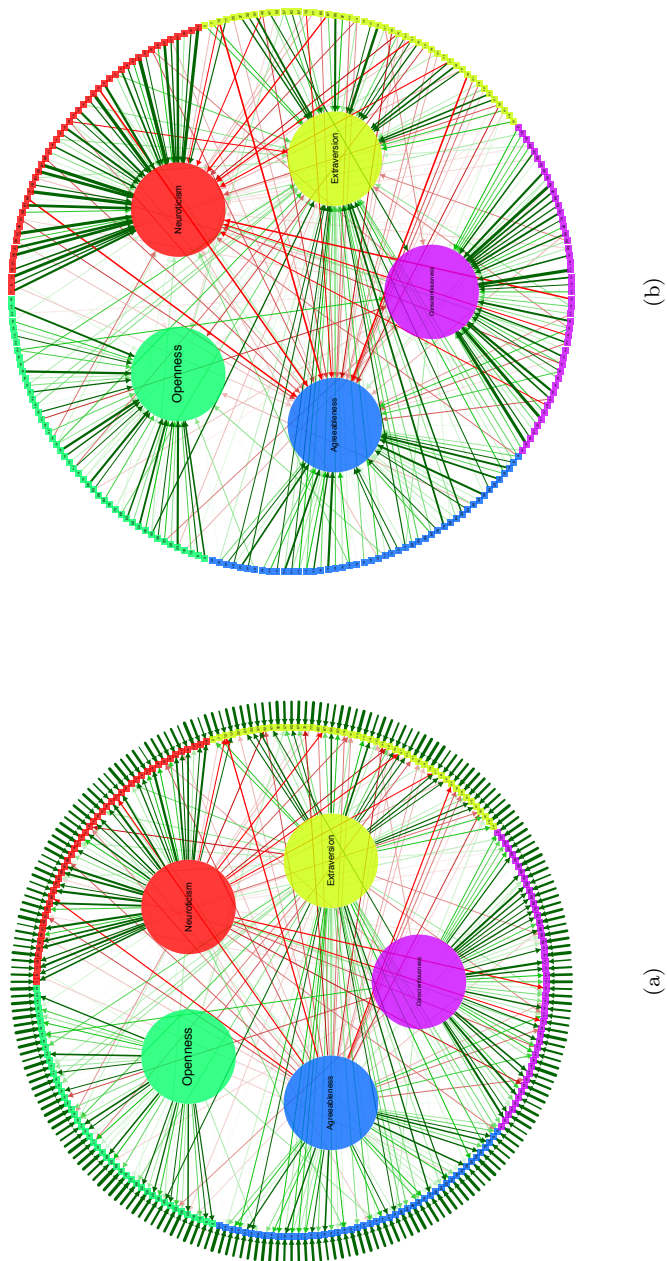
(b)

(a)

Figure 9.5: Visualization of an exploratory factor analysis (a) and a principal component analysis (b) in the NEO-PI-R dataset.

(i.e., it does not assume a common cause structure). It is used in the same way as `qgraph.efa()`; we can perform a PCA on the big 5 dataset using the same code as with the EFA:

```
qgraph.pca(big5, 5, groups = big5groups, rotation = "promax",
    minimum = 0.2, cut = 0.4, vsize = c(1, 15),
    borders = FALSE, asize = 0.07, esize = 4, vTrans = 200)
```

The right panel of Figure 9.5 shows the results. Notice that the arrows now point towards the factors, and that there are no residuals, as is the case in a formative measurement model[11]. Note that the correlations between items, which are not modeled in a formative model, are omitted from the graphical representation to avoid clutter.

Confirmatory Factor Analysis

Confirmatory factor models and regression models involving latent variables can be tested using structural equation modeling (SEM; Bollen, 1989; Pearl, 2000). SEM can be executed in R with three packages: *sem* (Fox, 2006), *OpenMx* (Boker et al., 2011) and *lavaan* (Rosseel, 2012). *qgraph* currently supports *sem* and *lavaan*, with support for *OpenMx* expected in a future version. The output of *sem* (a `"sem"` object) can be sent to (1) `qgraph()` for a representation of the standardized parameter estimates, (2) `qgraph.semModel()` for a path diagram of the specified model, and (3) `qgraph.sem()` for a 12 page pdf containing fit indices and several graphical representations of the model, including path diagrams and comparisons of implied and observed correlations. Similarly, the output of *lavaan* (a `"lavaan"` object) can be sent to `qgraph()` or `qgraph.lavaan()`.

SEM is often used to perform a confirmatory factory analysis (CFA; Stevens, 1996) in which variables each load on only one of several correlated factors. Often this model is identified by either fixing the first factor loading of each factor to 1, or by fixing the variance of each factor to 1. Because this model is so common, it should not be necessary to fully specify this model for each and every run. However, this is currently still the case. Especially using *sem* the model specification can be quite long.

The `qgraph.cfa()` function can be used to generate a CFA model for either the *sem* package or the *lavaan* package and return the output of these packages for further inspection. This function uses the `groups` argument as a measurement model and performs a CFA accordingly. The results can be sent to another function, and are also returned. This is either a `"sem"` or `"lavaan"` object which can be sent to `qgraph()`, `qgraph.sem()`, `qgraph.lavaan()` or any function that can handle the object. We can perform the CFA on our dataset using *lavaan* with the following code:

```
fit <- qgraph.cfa(cov(big5), N = nrow(big5),
    groups = big5groups, pkg = "lavaan",
    opts = list(se = "none"), fun = print)
```

---

[11]In `qgraph.loadings` there are no arrows by default, but these can be set by setting the `model` argument to `"reflective"` or `"formative"`.
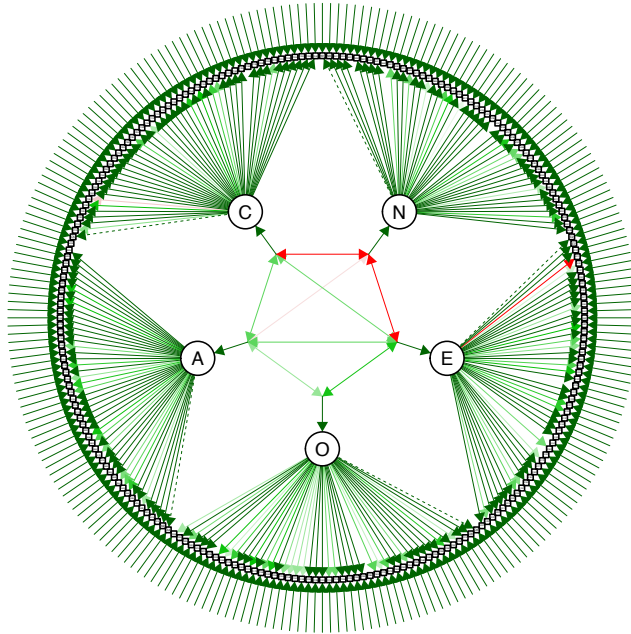
Figure 9.6: Standardized parameter estimations of a confirmatory factor analysis performed on the NEO-PI-R dataset.

```
lavaan (0.4-11) converged normally after 128 iterations

  Number of observations                              500

  Estimator                                            ML
  Minimum Function Chi-square                   60838.192
  Degrees of freedom                                28430
  P-value                                           0.000
```

Note that we did not estimate standard errors to save some computing time. We can send the results of this to `qgraph.lavaan` to get an output document.

Figure 9.6 shows part of this output: a visualization of the standardized parameter estimates. We see that the first loading of each factor is fixed to 1 (dashed lines) and that the factors are correlated (bidirectional arrows between the factors). This is the default setup of `qgraph.cfa()` for both *sem* and *lavaan*[12]. From

---

[12]Using *lavaan* allows to easily change some options by passing arguments to `cfa()` using the `opts` argument. For example, we could fix the variance of the factors to 1 by specifying
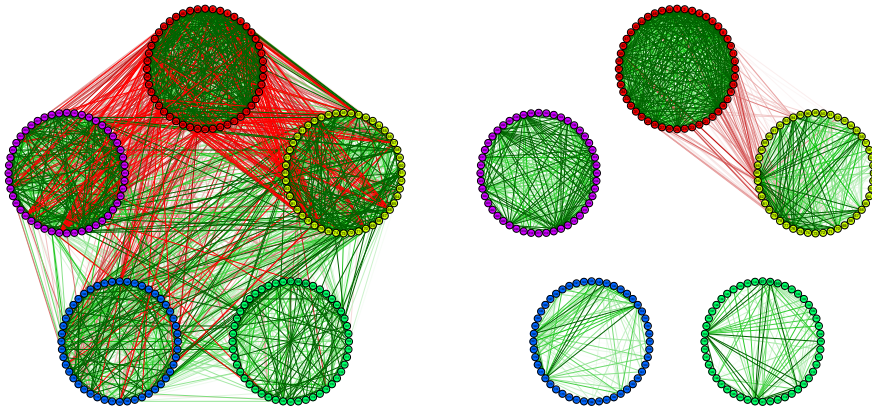
Figure 9.7: The observed correlations in the NEO-PI-R dataset (left) and the correlations that are implied by the model of Figure 9.6 (right).

the output above, we see that this model does not fit very well, and inspection of another part of the output document shows why this is so: Figure 9.7 shows a comparison of the correlations that are implied by the CFA model and the observed correlations, which indicates the source of the misfit. The model fails to explain the high correlations between items that load on different factors; this is especially true for extraversion and agreeableness items. The overlap between these items was already evident in the previous figures, and this result shows that this overlap cannot be explained by correlations among the latent factors in the current model.

9.4 Conclusion

The network approach offers novel opportunities for the visualization and analysis of vast datasets in virtually all realms of science. The *qgraph* package exploits these opportunities by representing the results of well-known statistical models graphically, and by applying network analysis techniques to high-dimensional variable spaces. In doing so, *qgraph* enables researchers to approach their data from a new perspective.

qgraph is optimized to accommodate both inexperienced and experienced R users: The former can get a long way by simply applying `qgraph.panel()` to a correlation matrix of their data, while the latter may utilize the more complex functions in *qgraph* to represent the results of time series modeling. Overall, however, the package is quite accessible and works with carefully chosen defaults,

---

`qgraph.cfa(\ldots,opts=list(std.lv=TRUE)).`

so that it almost always produces reasonable graphs. Hopefully, this will allow the network approach to become a valuable tool in data visualization and analysis.

Since *qgraph* is developed in a psychometric context, its applications are most suitable for this particular field. In this chapter we have seen that *qgraph* can be used to explore several thousands of correlations with only a few lines of code. This resulted in figures that not only showed the structure of these correlations but also suggested where exactly the five-factor model did not fit the data. Another example is the manual of a test battery for intelligence (IST; Liepmann, Beauducel, Brocke, & Amthauer, 2010) in which such graphs were used to argue for the validity of the test. Instead of examining full datasets *qgraph* can also be used to check for statistical assumptions. For example, these methods can be used to examine multicollinearity in a set of predictors or the local independence of the items of a subtest.

Clearly, we are only beginning to scratch the surface of what is possible in the use of networks for analyzing data, and the coming years will see considerable developments in this area. Especially in psychometrics, there are ample possibilities for using network concepts (such as centrality, clustering, and path lengths) to gain insight in the functioning of items in psychological tests.

Chapter 10

State of the aRt Personality Research

Abstract

Network analysis represents a novel theoretical approach to personality. Network approaches motivate alternative ways of analyzing data, and suggest new ways of modeling and simulating personality processes. In the present chapter, we provide an overview of network analysis strategies as they apply to personality data. We discuss different ways to construct networks from typical personality data, and show how to compute and interpret important measures of centrality and clustering. All analyses are illustrated using a data set on the commonly used HEXACO questionnaire using elementary R-code that readers may easily adapt to apply to their own data.

10.1 Introduction

A network is an abstract model composed of a set of nodes or vertices, a set of edges, links or ties that connect the nodes, together with information concerning the nature of the nodes and edges (e.g., De Nooy, Mrvar, & Batagelj, 2011). Figure 10.1 reports the example of a simple network, with six nodes and seven edges. The nodes usually represent entities and the edges represent their relations. This simple model can be used to describe many kinds of phenomena, such as social relations, technological and biological structures, and information networks (e.g., Newman, 2010, Chapters 2–5). Recently networks of relations among thoughts, feelings and behaviors have been proposed as models of personality and of psychopathology: in this framework, traits have been conceived of as emerging

---

This chapter has been adapted from: Costantini, G., Epskamp, S., Borsboom, D., Perugini, M., Mõttus, R., Waldorp, L. J., and Cramer, A. O. J. (2014). State of the aRt personality research: A tutorial on network analysis of personality data in R. *Journal of Research in Personality 54,* 13–29. (The first two authors contributed equally to this work).
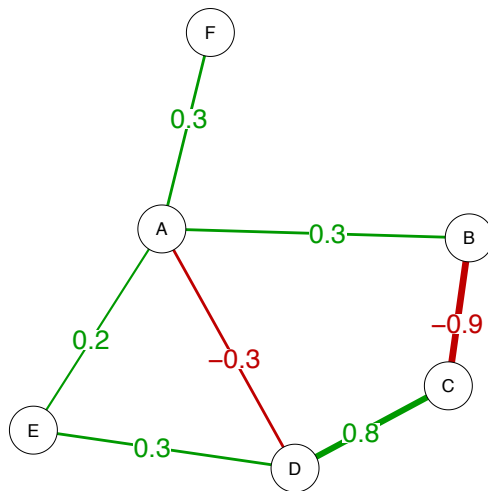
Figure 10.1: A network with six nodes and seven edges. Positive edges are green and negative edges are red. The letters identify the nodes, the numbers represent weights associated to the edges

phenomena that arise from such networks (Borsboom & Cramer, 2013; Cramer, Sluis, et al., 2012; Schmittmann et al., 2013). An R package, *qgraph*, has been developed for the specific purpose of analyzing personality and psychopathology data (Epskamp et al., 2012).

The aim of this contribution is to provide the reader with the necessary theoretical and methodological tools to analyze personality data using network analysis, by presenting key network concepts, instructions for applying them in R (R Core Team, 2016), and examples based on simulated and on real data. First, we show how a network can be defined from personality data. Second, we present a brief overview of important network concepts. Then, we discuss how network concepts can be applied to personality data using R. In the last part of the chapter, we outline how network-based simulations can be performed that are specifically relevant for personality psychology. Both the data and the R code are available for the reader to replicate our analyses and to perform similar analyses on his/her own data.

10.2 Constructing Personality Networks

A typical personality data set consists of cross-sectional measures of multiple subjects on a set of items designed to measure several facets of personality. In standard approaches in personality research, such data are used in factor analysis to search for an underlying set of latent variables that can explain the structural covariation in the data. In a causal interpretation of latent variables (Borsboom

et al., 2003), responses to items such as "I like to go to parties" and "I have many friends" are viewed as being causally dependent on a latent variable (e.g., extraversion). For example, McCrae and Costa's (2008) interpretation of the relation between extraversion and its indicators is explicitly causal: "extraversion causes party-going behavior in individuals" (McCrae & Costa, 2008, p. 288). This approach has culminated in currently influential models such as the Five Factor Model of personality (McCrae & Costa, 2008), in which five dominant latent variables are ultimately held responsible for most of the structural covariation between responses to personality items (additional latent factors such as facets may cause some of the covariation).

Recently, however, this perspective has been challenged in the literature (Cramer, Sluis, et al., 2012). In particular, it has been put forward that the default reliance on latent variable models in personality may be inappropriate, because it may well be that the bulk of the structural covariation in personality scales results from direct interactions between the variables measured through personality items. For instance, one may suppose that people who like to go to parties gain more friends because they meet more people, and people who have more friends get invited to good parties more often. In this way, one can achieve an explanation of the relevant pattern of covariation without having to posit latent variables.

Thus, in this scheme of thinking, one may suppose that, instead of reflecting the pervasive influence of personality factors, the structural covariance in personality is actually due to local interactions between the variables measured. In this way of thinking, personality resembles an ecosystem in which some characteristics and behaviors stimulate each other, while others have inhibitory relations. Under this assumption, the proper way to analyze personality data is not through the a priori imposition of a latent variable structure, but through the construction of a network that represents the most important relations between variables; this way, one may get a hold of the structure of the ecosystem of personality.

It is important to stress that not all personality scholars have embraced a causal view of latent factors. Some researchers for instance consider factors as the common elements shared by many observable variables and not as their causes (Ashton & Lee, 2005; Funder, 1991; Lee, 2012). Also from this different theoretical perspective, the heuristic value of network analysis remains important. Factor and network analysis differ, at the very least, in the fact that they direct the researcher's attention toward different aspects of personality. While factor analysis focuses almost exclusively on the elements shared among the indicators, whether or not interpreted causally, network analysis shifts the focus towards the direct relationships among the observable variables. We do not challenge the use of factor analysis as a statistical technique by itself: network analysis and factor analysis can in principle be combined (Cramer, Sluis, et al., 2012; Lee, 2012)[1]. However, a network perspective may foster important insights in the field that are unlikely to come by relying exclusively on a latent variable perspective.

The current section explains how a network structure can be estimated and visualized in R based on typical personality research data. We explain how networks

---

[1]See also Chapter 7 and Chapter 8 for recent discussions on this topic.

are encoded in weights matrices, discuss the most important kinds of networks and show how to estimate these network.

Directed and Undirected Networks

There are different types of networks, which yield different kinds of information and are useful in different situations. In a *directed* network, relationships between nodes are asymmetrical. Research on directed networks has seen extensive developments in recent years since the work of Pearl and Verma (1995) and others on causal systems. Methodology based on directed networks is most useful if one is willing to accept that the network under consideration is *acyclic*, which means that there are no feedback loops in the system (if A influences B, then B cannot influence A). A directed network without feedback loops is called a Directed Acyclic Graph (DAG). In contrast, in an *undirected* network, all relationships are symmetrical. These networks are most useful in situations where (a) one cannot make the strong assumption that the data-generating model is a DAG, (b) one suspects that some of the relations between elements in the network are reciprocal, and (c) one's research is of an exploratory character and is mainly oriented to visualizing the salient relations between nodes. Since the latter situation appears more realistic for personality research, the current chapter focuses primarily on undirected networks.

Encoding a Network in a Weights Matrix

The structure of a network depends on the relations between its elements. *Unweighted* networks represent only the presence or absence of the edges, while *weighted* networks encode additional information about the magnitude of the connections. When it is important to distinguish large from small connections—such as in personality—weighted networks are preferred. A weighted network can be encoded in a *weights matrix*, which is a square matrix in which each row and column indicate a node in the network. The elements of the matrix indicate the strength of connection between two nodes; a zero in row i and column j indicates that there is no edge between node i and node j. For example, the network of Figure 10.1 can be represented with the following weights matrix:

	A	B	C	D	E	F
A	0	0.3	0	-0.3	0.2	0.3
B	0.3	0	-0.9	0	0	0
C	0	-0.9	0	0.8	0	0
D	-0.3	0	0.8	0	0.3	0
E	0.2	0	0	0.3	0	0
F	0.3	0	0	0	0	0

In this network there are positive connections, for instance between nodes A and B, and negative connections, for instance between nodes A and D. The zeroes in the matrix indicate that there are absent connections in the network, such as

between nodes A and C. Furthermore, we may note that the matrix is symmetric and that the diagonal values are not used in the network.

The *qgraph* package (Epskamp et al., 2012) can be used to visualize such a weights matrix as a network:

```
mat <- matrix(c(
   0,  0.3,    0, -0.3,  0.2, 0.3,
 0.3,    0, -0.9,    0,    0,   0,
   0, -0.9,    0,  0.8,    0,   0,
-0.3,    0,  0.8,    0,  0.3,   0,
 0.2,    0,    0,  0.3,    0,   0,
 0.3,    0,    0,    0,    0,   0), ncol = 6, nrow = 6,
byrow = TRUE)

library("qgraph")

qgraph(mat, layout = "spring", edge.labels = TRUE,
  labels = LETTERS[1:6], fade = FALSE)
```

Here, the first argument in the `qgraph` function—the (`mat`) argument—calls the weights matrix to plot. The other arguments specify graphical layout.

Correlation Networks, Partial Correlation Networks, and LASSO Networks

To illustrate network analysis on personality data we made public a dataset in which nine-hundred-sixty-four participants (704 female and 256 male, M age = 21.1, SD = 4.9, plus four participants who did not indicate gender and age) were administered the HEXACO-60 (Ashton & Lee, 2009). The HEXACO-60 is a short 60-items inventory that assesses six major dimensions of personality: honesty-humility, emotionality, extraversion, Agreeableness vs. anger, conscientiousness and openness to experience (Ashton & Lee, 2007). Each of the major dimensions subsumes four facets, which can be computed as the average of two or three items. Participants indicated their agreement with each statement on a scale from 1 (*strongly disagree*) to 5 (*strongly agree*). An example of an item (of trait emotionality) is "When I suffer from a painful experience, I need someone to make me feel comfortable".

We can load the HEXACO dataset into R as follows:

```
Data <- read.csv("HEXACOfacet.csv")
```

The reader may use `str(Data)` to get an overview of the variables in the dataset. Exploratory factor analysis can be performed to inspect the structure of the dataset, using package psych (Revelle, 2010). The command `fa.parallel(Data)` executes parallel analysis, which suggests six factors. The command `fa(r=Data, nfactors=6, rotate="Varimax")` can be used to extract six orthogonal factors. Factor loadings are reported in Table 10.1 and reproduce the expected structure

	E	C	O	X	H	A	Uniq.	Compl.	Smc
Hsi	-.05	.11	.11	.05	**.60**	-.05	.61	1.17	.26
Hfa	.14	.22	.15	-.04	**.63**	.19	.48	1.69	.39
Hga	.11	-.01	.24	.03	**.54**	.14	.62	1.65	.29
Hmo	.04	-.01	.05	-.05	**.44**	.07	.79	1.12	.16
Efe	**.48**	.03	-.16	-.22	-.07	-.04	.69	1.72	.27
Ean	**.55**	.17	.08	-.12	.11	-.11	.63	1.54	.30
Ede	**.66**	-.01	-.11	-.08	-.01	-.03	.55	1.10	.34
Ese	**.68**	.07	.02	.10	.13	.08	.50	1.18	.36
Xss	-.36	.18	.06	**.53**	-.08	.00	.54	2.14	.38
Xsb	-.05	.08	.07	**.63**	-.02	-.25	.52	1.40	.36
Xso	.17	-.02	.03	**.65**	.06	.01	.55	1.17	.33
Xli	-.11	.06	.02	**.67**	.00	.12	.52	1.13	.37
Afo	.09	-.09	.04	.13	.16	**.43**	.75	1.68	.20
Age	.09	-.06	-.02	.04	.13	**.54**	.68	1.21	.23
Afl	-.06	-.02	-.01	-.10	.06	**.67**	.53	1.08	.29
Apa	-.11	.10	.14	-.01	.09	**.49**	.71	1.45	.22
Cor	.01	**.73**	-.07	.06	.01	.00	.46	1.03	.37
Cdi	.19	**.58**	.19	.21	.18	-.03	.51	1.99	.41
Cpe	.08	**.70**	.18	.05	.06	-.08	.46	1.22	.41
Cpr	-.21	**.52**	.12	-.12	.15	.12	.62	1.87	.32
Oaa	-.04	.17	**.71**	-.04	.15	.04	.44	1.23	.42
Oin	-.25	.09	**.59**	.04	.15	-.01	.56	1.55	.35
Ocr	.15	.01	**.62**	.14	.01	.08	.56	1.26	.32
Oun	-.07	.01	**.57**	.10	.11	-.08	.65	1.22	.29

Table 10.1: Factor loadings. Factors are labeled according to their highest loadings. Note: E = loading on emotionality, C = loading on conscientiousness, O = loading on openness to experience, X = loading on extraversion, H = loading on honesty-humility, A = loading on agreeableness versus anger. Smc = squared multiple correlation of each facet with all the others. Uniq. = uniqueness. Compl. = Hofmann's row-complexity index (1978).

(Ashton & Lee, 2009). For each facet Table 10.1 reports also the squared multiple correlation with all the other facets and the Hofmann's row-complexity index, which represents the number of latent variables needed to account for each manifest variable (Hofmann, 1978; Pettersson & Turkheimer, 2010) and is included in the output of function `fa`.

Correlation networks.   We will construct networks by representing measured variables as nodes, connected by an edge if two variables interact with each other. To do this we can use a simple heuristic: node A is connected to node B if node A *is associated with* node B. A correlation matrix describes pairwise associations between the facets of the HEXACO and therefore can be used for estimating such a network structure. We can compute Pearson correlations on this dataset using the `cor` function:

```
cor(Data)
```

Notice that a correlation matrix is symmetric and that a value of zero indicates no connection. Thus, a correlation matrix, by default, has properties that allow it to be used as a weights matrix to encode an undirected network. Using this connection opens up the possibility to investigate correlation matrices visually as networks. To do so, we can use the *qgraph* package and ask it to plot the correlation matrix as a network; in the remainder, we will indicate this network as a *correlation network*. To facilitate interpretation, we color nodes according to the assignment of facets to traits as specified in the HEXACO manual:

```
groups <- factor(c(
    rep("Honesty Humility", 4),
    rep("Emotionality", 4),
    rep("Extraversion", 4),
    rep("Agreeableness vs. anger", 4),
    rep("Conscientiousness", 4),
    rep("Openness to experience", 4)))

qgraph(cor(Data), layout = "spring", labels = colnames(Data),
groups = groups)
```

Figure 10.2A represents the correlation structure of the facets of the HEXACO dataset. Green lines represent positive correlations, while red lines represent negative correlations. The wider and more saturated an edge is drawn, the stronger the correlation. As the reader may expect, the figure shows that the correlations of facets within traits are generally higher than the correlations of facets between traits, which is likely to reflect the fact that in psychometric practice items are typically grouped and selected on the basis of convergent and discriminant validity (Campbell & Fiske, 1959).

In recent literature correlation networks have been applied to grasp complex co-variation patterns in personality data that would be harder to notice otherwise in, say, factor loading matrices. Epskamp et al. (2012) showed how *qgraph* can be used to visualize the correlational structure of a 240 node dataset (Dolan et al., 2009) in which the NEO-PI-R (Costa & McCrae, 1992; Hoekstra et al., 2003) was used to assess the five factor model for personality (McCrae & Costa, 2008). Cramer, Sluis, et al. (2012) further analyzed this network and showed that it did not correspond to a correlation network that should arise had the data been generated by the five factor model for personality. Ziegler et al. (2013) constructed a correlation network on 113 personality facet scale scores from the NEO-PI-R, HEXACO, 6FPQ, 16PF, MPQ, and JPI and interpreted this network as a nomological network usable in scale development. Schlegel, Grandjean, and Scherer (2013) investigated the overlap of social and emotional effectiveness constructs and found the correlation network to display four meaningful components. Finally, Franić, Borsboom, Dolan, and Boomsma (2014) used correlation networks to show the similarity between genetic and environmental covariation between items of the NEO-FFI.

Partial correlation networks.   Correlation networks are highly useful to visualize interesting patterns in the data that might otherwise be very hard to spot. However, they are not necessarily optimal for the application of network analysis if the goal is to extract the structure of a data-generating network. The reason is that correlations between nodes in the network may be spurious, rather than being due to a genuine interaction between two nodes. For instance, spurious correlations may arise as the consequence of shared connections with a third node. Often, therefore, a network is constructed using the partial correlation matrix, which gives the association that is left between any two variables after conditioning on all other variables. The partial correlation coefficients are directly related to the inverse of the correlation matrix, also called the precision matrix (Lauritzen, 1996; Pourahmadi, 2011). Networks constructed on this basis are called *partial correlation networks* or *concentration graphs* (Cox & Wermuth, 1993), and the statistical data-generating structures that they encode are known as Markov random fields (Kindermann, Snell, et al., 1980).

The partial correlation network can be obtained in *qgraph* by setting the argument `graph` to `"concentration"`:

```
qgraph(cor(Data), layout = "spring", labels = colnames(Data),
groups = groups, graph = "concentration")
```

The partial correlation network is shown in Figure 10.2B. We can see that nodes still cluster together; the partial correlations within traits are generally stronger than the partial correlations between traits. Comparing figures 2A and 2B we can see structure emerging in for example the Openess (purple) cluster: the creativity node (Ocr) is no longer directly connected to the inquisitiveness (Oin) and unconventionality (Oun) nodes but now indirectly via the aesthetic appreciation (Oaa) node. Furthermore, we can see that the conscientiousness node prudence (Cpr) now has a more central role in the network and obtained relatively stronger connections with nodes of different traits: flexibility (Afl) and patience (Apa) of the Agreeableness vs. anger trait and sociability (Xso) and Social self-esteem (Xss) of the extroversion trait.

Adaptive LASSO networks.   In weighted networks, two nodes are connected if and only if the strength of connection between them is nonzero; a value of zero in the weights matrix encodes no connection between two nodes. Both the correlation and the partial correlation networks have been estimated based on an empirical sample and will therefore not result in exact zeroes. Thus, both networks will always be fully connected networks, possibly with arbitrarily small weights on many of the edges.

It has been argued that in social sciences everything is to some extent correlated with everything. This is akin to what Meehl and Lykken have called the *crud factor* or *ambient noise level* (Lykken, 1968, 1991; Meehl, 1990) and what may at least partly be responsible for the controversial general factor of personality (Musek, 2007). If a network model of pairwise interactions is assumed to underlie the data then all nodes that are indirectly connected will be correlated, mainly due to spurious connections. Therefore, even at the population level we can assume
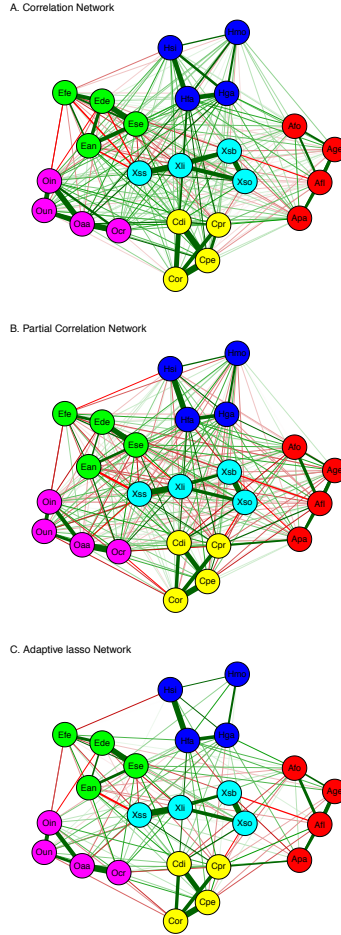
Figure 10.2: Networks of the HEXACO-60. Nodes represent personality facets (a description of each facet is provided in Table 10.2), green lines represent positive connections and red lines represent negative connections. Thicker lines represent stronger connections and thinner lines represent weaker connections. The node placement of all graphs is based on the adaptive LASSO network to facilitate comparison. The width and color are scaled to the strongest edge and not comparable between graphs; edge strengths in the correlation network are generally stronger than edge strengths in the partial correlation network.

that most correlations in personality research will be nonzero, resulting in a fully connected correlation network.

While correlation networks of personality measures are likely to be fully connected in the population, partial correlation networks are not necessarily so. This is of specific interest since the absence of an edge in a partial correlation network entails that two nodes are conditionally independent given all other nodes in the network—they cannot directly interact. The model in which partial correlations are set to zero is called the Gaussian graphical model (GGM; Lauritzen, 1996) as it can be visualized as a network. An optimal GGM is both sparse (many absent edges) while maintaining a high likelihood. Finding such a model corresponds to checking which connections are absent in the population network. Default significance tests can be used for this purpose (Drton & Perlman, 2004). However, significance tests require an arbitrary choice of significance level; different choices yield different results, with more stringent significance levels resulting in sparser networks. If one ignores this issue, one has a multiple testing problem, whereas if one deals with it in standard ways (e.g., through a Bonferroni correction), one faces a loss of power.

A practical way to deal with the issue of arbitrary choices is to construct networks based on different choices and to see how stable the main results are; however, a more principled alternative is to use a LASSO penalty (Friedman, Hastie, & Tibshirani, 2008) in estimating the partial correlation networks. This causes small connections to automatically shrink to be exactly zero and results in a parsimonious network. If the data indeed arose from a sparse network with pairwise interactions, such a procedure will in fact converge on the generating network (Foygel & Drton, 2011).

The adaptive LASSO is a generalization of the LASSO that assigns different penalty weights for different coefficients (Zou, 2006) and outperforms the LASSO in the estimation of partial correlation networks, especially if the underlying network is sparse (Fan, Feng, & Wu, 2009; Krämer et al., 2009). The penalty weights can be chosen in a data-dependent manner, relying on the LASSO regression coefficients (Krämer et al., 2009). In simulation studies, the likelihood of false positives using this method resulted even smaller than that obtained with the LASSO penalization (Krämer et al., 2009), so if an edge is present in the adaptive LASSO network one can reasonably trust that there is a structural relation between the variables in question (of course, the network does not specify the exact nature of the relation, which may for instance be due to a direct causal effect, a logical relation pertaining to item content, a reciprocal effect, or the common effect of an unmodeled latent variable).

The adaptive LASSO is also convenient practically, as it is implemented in the R-package *parcor* (Krämer et al., 2009). Since the adaptive LASSO, as implemented in package parcor, relies on k-fold validation, `set.seed` can be used to ensure the exact replicability of the results, which might be slightly different otherwise. To estimate the network structure of the HEXACO dataset according to the adaptive LASSO, the following code can be used:

```
library("parcor")
library("Matrix")
```

```
set.seed(100)
adls <- adalasso.net(Data)
network <- as.matrix(forceSymmetric(adls$pcor.adalasso))
qgraph(network, layout = "spring", labels = colnames(Data),
groups = groups)
```

The adaptive LASSO network is shown in Figure 10.2C. One can see that, compared to the partial correlation network, the adaptive LASSO yields a more parsimonious graph (fewer connections) that encodes only the most important relations in the data; In this network 134 (48.6%) of the edges are identified as zero.

10.3 Analyzing the Structure of Personality Networks

Once a network is estimated, several indices can be computed that convey information about network structure[2]. Two types of structure are important. First, one is typically interested in the *global* structure of the network: how large is it? Does it feature strong clusters? Does it reveal a specific type of structure, like a small-world (Watts & Strogatz, 1998)? Second, one may be interested in *local* patterns, i.e., one may want to know how nodes differ in various characteristics: which nodes are most central? Which nodes are specifically strongly connected? What is the shortest path from node A to node B? Here we discuss a limited selection of indices that we regard as relevant to personality research, focusing especially on centrality and clustering coefficients. More extensive reviews of network indices may be found in Boccaletti et al. (2006); Butts (2008); de De Nooy et al. (2011); Kolaczyk (2009); and Newman (2010).

Descriptive Statistics

Before the computation of centrality measures, a number of preparatory computations on the data are in order. The network is undirected, therefore the corresponding weights matrix is symmetric and each edge weight is represented twice, above and below the main diagonal. The function `upper.tri` can be used to extract the unique edge weights[3] and save them in a vector:

```
ew <- network[upper.tri(network)]
```

To compute the number of edges in the network, it is sufficient to define a logical vector that has value `TRUE` (= 1) if the edge is different from zero and `FALSE` (

---

[2]The adaptive LASSO networks, the correlation and the partial correlation networks are characterized by the presence of both positive and negative edges. The importance of signed networks is apparent not only in the study of social phenomena, in which it is important to make a distinction between liking and disliking relationships (e.g., Leskovec, Huttenlocher, & Kleinberg, 2010), but also in the study of personality psychology (e.g., Costantini & Perugini, 2014). Some network indices have been generalized to the signed case (e.g., Costantini & Perugini, 2014; Kunegis, Lommatzsch, & Bauckhage, 2009), however most indices are designed to unsigned networks. For the computation of the latter kind of indices, we will consider the edge weights in absolute value.

[3]The function `upper.tri` extracts the elements above the main diagonal. One could equally consider those below the diagonal using the function `lower.tri`.

= 0) if the edge is exactly zero (i.e., absent). The sum of this vector gives the number of nonzero edges. With a similar procedure, it is possible to count the positive and the negative edges: it is sufficient to replace != with > or <:

```
sum(ew != 0) # the number of edges
sum(ew > 0) # the number of positive edges
sum(ew < 0) # the number of negative edges
```

The network has 142 edges, of which 100 are positive and 42 are negative. The function t.test can be used to compare the absolute weights of the positive versus the negative edges:

```
t.test(abs (ew [ew > 0]), abs(ew [ew < 0]) , var.equal = TRUE)
```

In our network, positive edges are generally associated to larger weights (M = .11, SD = .09) than the negative edges (M = .06, SD = .04), and the t-test indicates that this difference is significant, t(140) = 3.13, p = .0022.

Centrality Measures

Not all nodes in a network are equally important in determining the network's structure and, if processes run on the network, in determining its dynamic characteristics (Kolaczyk, 2009). Centrality indices can be conceived of as operationalizations of a node's importance, which are based on the pattern of the connections in which the node of interest plays a role. In network analysis, centrality indices are used to model or predict several network processes, such as the amount of flow that traverses a node or the tolerance of the network to the removal of selected nodes (Borgatti, 2005; Crucitti, Latora, Marchiori, & Rapisarda, 2004; Jeong, Mason, Barabási, & Oltvai, 2001) and can constitute a guide for network interventions (Valente, 2012). Several indices of centrality have been proposed, based on different models of the processes that characterize the network and on a different conception of what makes a node important (Borgatti & Everett, 2006; Borgatti, 2005) The following gives a succinct overview of the most often used centrality measures[4].

Degree and strength. First, degree centrality is arguably the most common centrality index and it is defined as the number of connections incident to the node of interest (Freeman, 1978). The degree centrality of node C in Figure 10.1 is 2 because it has two connections, with nodes B and D. Degree can be straightforwardly generalized to weighted networks by considering the sum of the weights of the connections (in absolute value), instead of their number. This generalization is called *strength* (Barrat, Barthelemy, Pastor-Satorras, & Vespignani, 2004; Newman, 2004). For instance, strength of node C in Figure 10.1 is 1.7, which is

---

[4]The functions to implement centrality indices, clustering coefficients and small-worldness are implemented in the R package *qgraph* (Epskamp et al., 2012). Some of the functions rely on procedures originally implemented in packages *igraph* (Csardi & Nepusz, 2006), *sna* (Butts, 2010), and *WGCNA* (Langfelder & Horvath, 2012). These packages are in our experience among the most useful for network analysis.

the highest in the network. Degree and strength focus only on the paths of unitary length (Borgatti, 2005). A strength-central personality characteristic (e.g., an item, a facet or a trait) is one that can influence many other personality characteristics (or be influenced by them) directly, without considering the mediating role of other nodes.

Closeness and betweenness. Several other measures exist that, differently from degree centrality and the related indices, consider edges beyond those incident to the focal node. An important class of these indices rely on the concepts of *distance* and of *geodesics* (Brandes, 2001; Dijkstra, 1959). The distance between two nodes is defined as the length of the shortest path between them. Since, in typical applications in personality psychology, weights represent the importance of an edge, weights are first converted to lengths, usually by taking the inverse of the absolute weight (Brandes, 2008; Opsahl et al., 2010). The *geodesics* between two nodes are the paths that connect them that have the shortest distance. *Closeness centrality* (Freeman, 1978; Sabidussi, 1966) is defined as the inverse of the sum of the distances of the focal node from all the other nodes in the network[5]. In terms of network flow, closeness can be interpreted as the expected speed of arrival of something flowing through the network (Borgatti, 2005). A closeness-central personality characteristic is one that is likely to be quickly affected by changes in another personality characteristic, directly or through the changes in other personality features. Its influence can reach other personality features more quickly than the influence of those that are peripheral according to closeness, because of the short paths that connect itself and the other traits. In the network in Figure 10.1, node D has the highest closeness. To compute the exact value of closeness, one should first compute the distances between D and all the other nodes: A (1/0.3), B (1/0.8 + 1/0.9), C (1/0.8), E (1/0.3) and F (1/.3 + 1/.3). The sum of all the distances is 16.94 and the inverse, 0.59, is the closeness centrality of D.

Betweenness centrality is defined as the number of the geodesics between any two nodes that pass through the focal one. To account for the possibility of several geodesics between two nodes, if two geodesics exist, each one is counted as a half path and similarly for three or more (Brandes, 2001; Freeman, 1978). Betweenness centrality assumes that shortest paths are particularly important (Borgatti, 2005): if a node high in betweenness centrality is removed, the distances among other nodes will generally increase. Both closeness and betweenness centrality can be applied to weighted and directed networks, as long as the weights and/or the directions of the edges are taken into account when computing the shortest paths (e.g., Opsahl et al., 2010).

The betweenness centrality of node A in Figure 10.1 is 4 and is the highest in the network. The four shortest paths that pass through A are those between F and the nodes B, C, D, and E. Betweenness centrality can also be extended to evaluate the centrality of edges instead of nodes, by considering the geodesics

---

[5]The computation of closeness assumes that the network is connected (i.e., a path exists between any two nodes), otherwise, being the distance of disconnected nodes infinite, the index will result to zero for all the nodes. Variations of closeness centrality that address this issue have been proposed (Kolaczyk, 2009; Opsahl et al., 2010). Alternatively it can be computed only for the largest component of the network (Opsahl et al., 2010).

that pass through an edge: this generalization is called *edge betweenness centrality* (Brandes, 2008; Newman, 2004; Newman & Girvan, 2004). For instance, the edge-betweenness centrality of the edge (D,E) is 3 and the three shortest paths that pass through (D,E) are the one between D and E, the one between C and E (through D), and the between B and E (through C and D).

Betweenness-central personality characteristics and betweenness-central edges are particularly important for other personality characteristics to quickly influence each other. It is interesting to investigate the conditions in which some nodes become more or less central. For instance, a study that analyzed a network of moods showed that the mood "worrying" played a more central role for individuals high in neuroticism than for those with low neuroticism (Bringmann et al., 2013): the prominent role of worrying for neuroticism was recently confirmed by an experimental fMRI study (Servaas, Riese, Ormel, & Aleman, 2014).

Brandes (2008) discusses several other variants of the shortest-paths betweenness, some of which are implemented in package *sna* (Butts et al., 2008). Generalizations of betweenness centrality that account for paths other than the shortest ones have been also proposed (Brandes & Fleischer, 2005; Freeman, Borgatti, & White, 1991; Newman, 2005). In addition, Opsahl and colleagues (2010) proposed generalizations of degree, closeness, and betweenness centralities by combining in the formula both the number and the weights of the edges. They introduced a tuning parameter that allows setting their relative importance: a higher value of the tuning parameter emphasizes the importance of the weights over the mere presence of the ties and vice versa. Another important family of centrality indices defines the centrality of a node as recursively dependent on the centralities of their neighbors. Among the most prominent of those indices are *eigenvector centrality* (Bonacich, 1972, 2007), Bonacich power (Bonacich, 1987) and *alpha centrality* (Bonacich & Lloyd, 2001).

Clustering Coefficients

Besides centrality, other network properties have been investigated that are relevant also for personality networks. The local *clustering coefficient* is a node property defined as the number of connections among the neighbors of a focal node over the maximum possible number of such connections (Watts & Strogatz, 1998). If we define a triangle as a triple of nodes all connected to each other, the clustering coefficient can be equally defined as the number of triangles to which the focal node belongs, normalized by the maximum possible number of such triangles. The clustering coefficient is high for a node i if most of i's neighbors are also connected to each other and it is important to assess the small-world property (Watts & Strogatz, 1998; Humphries & Gurney, 2008), as we detail below. Consider for instance the node D in Figure 10.1, which has three neighbors, A C, and E. Of the three possible connections among its neighbors, only one is present (the one between A and E), therefore its clustering coefficient is 1/3.

The clustering coefficient can be also interpreted as a measure of how much a node is redundant (Latora, Nicosia, & Panzarasa, 2013; Newman, 2010): if most of a node's neighbors are also connected with each other, removing that node will not make it harder for its neighbors to reach or influence each other. A

personality characteristic that has a high clustering coefficient is mainly connected to other personality features that are directly related to each other. In personality questionnaires the strongest connections are usually among nodes of the same subscale: in these cases, having a high clustering coefficient may coincide with having most connections with other nodes belonging to the same subscale, while having no large connection with nodes of other scales.

While in its original formulation the clustering coefficient can be applied only to unweighted networks (or to weighted networks, disregarding the information about weights), it has been recently generalized to consider positive edge weights (Saramäki, Kivelä, Onnela, Kaski, & Kertesz, 2007). The first of such generalizations was proposed by Barrat and colleagues (2004) and has been already discussed in the context of personality psychology and psychopathology (Borsboom & Cramer, 2013). Onnela and colleagues (2005) proposed a generalization that is based on the geometric averages of edge weights of each triangle centered on the focal node. A different generalization has been proposed in the context of gene co-expression network analysis by Zhang and Horvath, which is particularly suited for networks based on correlations (Kalna & Higham, 2007; Zhang, Horvath, et al., 2005). All of these generalizations coincide with the unweighted clustering coefficient when edge weights become binary (Saramäki et al., 2007). Recently three formulations of clustering, the unweighted clustering coefficient (Watts & Strogatz, 1998), the index proposed by Onnela et al. (2005) and the one proposed by Zhang et al. (2005) have been generalized to signed networks and the properties of such indices have been discussed in the context of personality networks (Costantini & Perugini, 2014).

Transitivity (or global clustering coefficient) is a concept closely connected to clustering coefficient that considers the tendency for two nodes that share a neighbor to be connected themselves for the entire network, instead than for the neighborhood of each node separately. It is defined as three times the number of triangles, over the number of connected triples in the network, where a connected triple is a node with two edges that connect it to an unordered pair of other nodes (Newman, 2003). Differently from the local clustering coefficient, transitivity is a property of the network and not of the single nodes. For instance, the network in Figure 10.1 has one triangle (A, D, E) and 12 connected triples, therefore its transitivity is $3 \times 1)/12 = 1/4$. Transitivity has been extended by Opsahl and Panzarasa (2009) to take into account edge weights and directions, and by Kunegis and collaborators to signed networks (Kunegis et al., 2009).

Small Worlds

The transitivity and clustering coefficient can be used to assess the network *small-world property*. The small-world property was initially observed in social networks as the tendency for any two people to be connected by a very short chain of acquaintances (Milgram, 1967). The small-world property is formally defined as the tendency of a network to have both a high clustering coefficient and a short average path length (Watts & Strogatz, 1998). Small-world networks are therefore characterized by both the presence of dense local connections among the nodes and of links that connect portions of the network otherwise far away from each

other. An index of *small-worldness* for unweighted and undirected networks has been proposed as the ratio of transitivity to the average distance between two nodes. Both transitivity and path length are standardized before the computation of small-worldness, by comparing them to the corresponding values obtained in equivalent random networks (with the same N and the same degree distribution). Alternatively, the index can be computed using the average of local clustering coefficients instead of transitivity. A network with a small-worldness value higher than three can be considered as having the small-world property, while a small-worldness between one and three is considered a borderline value (Humphries & Gurney, 2008). Because the assessment of small-worldness relies on shortest paths between all the pairs of nodes, it can be computed only for a connected network or the giant component of a disconnected network.

Application to the HEXACO Data

Centrality analyses. The function `centrality_auto` allows to quickly compute several centrality indices. It requires the weights matrix as input. The function automatically detects the type of network and can handle both unweighted and weighted networks, and both directed and undirected networks. For a weighted and undirected network, the function gives as output the node strength, the weighted betweenness and the weighted closeness centralities. The edge betweenness centrality is also computed.

```
centrality <- centrality_auto(network)
nc <- centrality$node.centrality
ebc <- centrality$edge.betweenness.centrality
```

The centrality values are computed and stored in variable `centrality`. Node centralities are then saved in the variable `nc` while edge betweenness centralities are saved in the variable `ebc`. The values of centrality for each node are reported in Table 10.2. The command `centralityPlot(network)` can be used to plot the centrality indices in a convenient way, that allows to quickly compare them. Table 10.3 reports the correlations among the three indices of node centrality together with Hofmann's (1978) row-complexity and the squared multiple correlation of each facet with all the others. All the indices of centrality have positive significant correlations with each other. Strength centrality and, to a lower extent, betweenness centrality, seem to be favored by row-complexity: sharing variance with more than one factor allows a facet to play a more central role. These results suggest that, in this network, facets tend to be central to the whole network and not only to their purported parent traits. All centrality indices, especially strength and closeness, correlate with the squared multiple correlations: The more variance a facet shares with other facets, the stronger are its connections and the more central results the corresponding node[6].

---

[6]Despite being substantial, the correlations of centrality indices with row-complexity and squared multiple correlations do not suggest that the indices fully overlap. Moreover, the relations can vary substantially and it is possible to imagine situations in which the relations are absent or even in the opposite direction.

Node	Dimension	Facet	Betweenness	Closeness	Strength	
Hsi	Honesty-Humility	Sincerity	5	2.66	0.73	
Hfa	Honesty-Humility	Fairness	**31**	3.03	**1.46**	
Hga	Honesty-Humility	Greed-avoidance	14	2.83	1.13	
Hmo	Honesty-Humility	Modesty	0	2.14	0.45	
Efe	Emotionality	Fearfulness	6	2.70	1.03	
Ean	Emotionality	Anxiety	2	3.04	1.10	
Ede	Emotionality	Dependence	3	3.02	1.05	
Ese	Emotionality	Sentimentality	17	3.17	**1.40**	
Xss	Extraversion	Social self-esteem	11	3.11	**1.35**	
Xsb	Extraversion	Social boldness	**23**	**3.33**	1.21	
Xso	Extraversion	Sociability	7	**3.19**	1.07	
Xli	Extraversion	Liveliness	12	3.12	1.29	
Afo	Agreeableness vs. anger	Forgiveness	5	2.70	1.00	
Age	Agreeableness vs. anger	Gentleness	5	2.66	0.80	
Afl	Agreeableness vs. anger	Flexibility	14	2.90	1.02	
Apa	Agreeableness vs. anger	Patience	5	2.85	0.85	
Cor	Conscientiousness	Organization	7	3.09	0.99	
Cdi	Conscientiousness	Diligence	**26**	**3.34**	1.30	
Cpe	Conscientiousness	Perfectionism	5	3.13	1.26	
Cpr	Conscientiousness	Prudence	**19**	**3.52**	**1.45**	
Oaa	Openness to experience	Aesthetic appreciation	14	2.95	1.24	
Oin	Openness to experience	Inquisitiveness	5	2.71	1.08	
Ocr	Openness to experience	Creativity	10	3.00	1.26	
Oun	Openness to experience	Unconventionality	3	2.63	0.98	

Table 10.2: Centrality Indices. Note: the four most central nodes according to each index are reported in bold.

	1	2	3	4	5
1. Betweenness	1	.61**	.72***	.32	.54**
2. Closeness	.61**	1	.75***	.15	.69***
3. Strength	.70***	.82***	1	.47*	.75***
4. Complexity	.41*	.28	.43*	1	.11
5. SMC	.56**	.73***	.79***	.12	1

Table 10.3: Correlation of node centralities, row-complexity and squared multiple correlation (SMC). Note: $* = p < .05$, $** = p < .01$, $*** = p < .001$. Pearson correlations are reported below the diagonal, Spearman correlations are reported above the diagonal. Complexity = Hofmann's row-complexity index. SMC = squared multiple correlation.

The three indices of centrality converge in indicating that node Cpr (prudence) is among the four most central nodes in this network. Cpr is also the more closeness central node and owes its high centrality to the very short paths that connect it to other traits. For instance, facets Apa (patience), Xso (sociability), and Xss (social self-esteem) are even closer to Cpr than other conscientiousness facets are[7]. This suggests that in the personality network it is very easy that a change in some portion of the network will eventually make a person either more reckless or more prudent. On the other hand, if a person becomes more reckless or more prudent, we can expect important changes in the overall network. This result, although it should be considered as preliminary, is in line with studies that investigated the evolution of conscientiousness. Impulse-control, a facet of conscientiousness that is very similar to prudence (Cpr), shows the most marked variation through the individual development compared to other conscientiousness facets (Jackson et al., 2009). It is possible that this is the case also because changes in other personality traits are expected to affect prudence more quickly than other facets, as revealed by its high closeness.

Hfa (fairness) is the most betweenness-central and strength-central node, but it is not particularly closeness-central (it is ranked 10th in closeness centrality). Figure 10.3 highlights the edges lying on the shortest paths that travel through node Hfa, in a convenient layout (the code for producing this figure is in the supplemental materials). The high betweenness centrality of Hfa is due the role that Hfa plays in transmitting the influence of other honesty-humility facets to different traits, and vice versa. The edge between nodes Hsi (sincerity) and Hfa is also the most betweenness-central in the whole network: most of the shortest paths between Hsi and other personality traits travel through this edge and therefore through Hfa. These results suggest that, if it was possible to reduce the possibility for fairness (Hfa) to vary, the influence of the other honesty-humility facets would propagate less easily to the rest of personality facets and vice versa. Such hypotheses could be tested for instance by comparing the personality networks of individuals that typically face situations in which their fairness is allowed to become active to the networks of individuals that usually face situations in which their fairness cannot be activated (Tett & Guterman, 2000). The characteristics of situations for instance could be assessed by using valid instruments such as the Riverside Situational Q-sort (Sherman, Nave, & Funder, 2010), which includes items such as "It is possible for P to deceive someone", or "Situation raises moral

---

[7]As an anonymous reviewer pointed out, one could wonder how can the length of the path between Cpr and other conscientiousness facets be longer than the path between Cpr and other nodes, given that Cpr's strongest correlations are those with the other conscientiousness facets. This happens because we did not consider the network defined by the zero-order correlations, but the adaptive LASSO penalized network of partial correlations (Krämer et al., 2009). As an example, consider the shortest path between Cpr and Cdi (diligence), which is slightly longer (8.80) than the shortest path between Cpr and Apa (patience; 6.82). Although the correlation between Cpr and Cdi is stronger ($r = .26$) than the correlation between Cpr and Apa ($r = .22$), in the adaptive LASSO network, the direct connection between Cpr and Cdi is smaller ($pr = .04$) than the one with Apa ($pr = .15$). While the shortest path between Cpr and Apa travels through their direct connection, the shortest path between Cpr and Cdi travels through node Cor (organization): prudence seems to influence (or to be influenced by) diligence especially through changes in orderliness, but this path of influence is longer than the direct path between Cpr and Apa.

Figure 10.3: Shortest paths that pass through node Hfa (fairness). The edges belonging to the shortest-paths are full, while the other edges are dashed.

or ethical issues" that would be relevant for this case.

Clustering coefficients. Many indices of clustering coefficient can be easily computed using function `clustcoef_auto`. The function requires the same input as `centrality_auto` and is similarly programmed to recognize the kind of data given as input and to choose an appropriate network representation for the data. By applying the function, we can immediately collect the results:

```
clustcoef <- clustcoef_auto(network)
```

The command `clusteringPlot(network, signed = TRUE)` can be used to plot the clustering coefficients in a convenient layout. Table 10.4 reports the correlation among several clustering coefficients. The unsigned indices are computed using the absolute values of the weights. In the following analyses we will use the signed version of the Zhang's clustering coefficient (Costantini & Perugini, 2014; Zhang et al., 2005), which resulted more resistant to random variations in the network.

Combining clustering coefficients and centrality. The signed clustering coefficient can be interpreted as an index of a node's redundancy in a node's neighborhood (Costantini & Perugini, 2014): the importance of the unique causal role of highly clustered nodes is strongly reduced by the presence of strong connections among their neighbors. In general, it is interesting to inspect whether there is a relation between centrality indices and clustering coefficients: in our experience, we found that the centrality indices were often inflated by the high

	1	2	3	4	5	6	7
1. Watts and Strogatz (1998)	1	.25	.65***	.51*	.90***	.57**	.94***
2. Watts and Strogatz, signed (Costantini & Perugini, 2014)	.26	1	.28	.45*	.29	.76***	.25
3. Zhang and Horvath (2005)	.49*	.30	1	.89***	.50*	.59**	.71***
4. Zhang and Horvath, signed (Costantini & Perugini, 2014)	.34	.33	.94***	1	.37	.79***	.53**
5. Onnela et al. (2005)	.89***	.25	.37	.24	1	.55**	.84***
6. Onnela et al., signed (Costantini & Perugini, 2014)	.61**	.76**	.59**	.64**	.66***	1	.53**
7. Barrat et al. (2004)	.94***	.30	.57**	.37	.87***	.60**	1

Table 10.4: Correlation among indices of local clustering coefficient. Note: $* = p < .05$, $** = p < .01$, $*** = p < .001$. Pearson correlations are reported below the diagonal, Spearman correlations are reported above the diagonal.

clustering in correlation networks. However this might be not true for networks defined with adaptive LASSO, which promotes sparsity (Krämer et al., 2009).

The following plots can be used to visualize both the centrality and the clustering coefficient of each node. The code reported here is for betweenness centrality, but it is easy to extend it to other indices by just replacing `"Betweenness"` with the index of interest. First the plot is created and then the node labels are added in the right positions, using the command text. Command `abline` can be used to trace lines in the plot. A horizontal line is created to visually identify the median value of betweenness and a vertical line to identify the median value of the clustering coefficient.

```
plot(clustcoef$signed_clustZhang, nc$Betweenness,
     col = "white")
text(clustcoef$signed_clustZhang, nc$Betweenness,
     rownames(nc))
abline(h = median(nc$Betweenness), col = "grey")
abline(v = median(clustcoef$signed_clustZhang),
     col = "grey")
```

The resulting plots are shown in Figure 10.4. It is apparent that the most central nodes do not have a particularly high clustering coefficient in this case and this is especially true for nodes Hfa and Cpr, which are among the most central in this network. The clustering coefficient correlates negatively with closeness centrality ($r = -.67$, $p < .001$), with strength ($r = -.82$, $p < .001$), and with betweenness centrality ($r = -.50$, $p = .013$).

One node, Hmo (modesty), emerges as both particularly high in clustering coefficient and low in all the centrality measures. Modesty correlates almost exclusively with other honesty-humility facets and has the lowest multiple correlation with all the other variables in our dataset and this is likely to have determined its

peripherality. A closer exam of its connections reveals that Hmo has seven neighbors, the three other facets of honesty-humility (His, Hfa, and Hga), facets anxiety and fearfulness of emotionality (Ean), facet social boldness of extraversion (Xsb) and facet prudence of conscientiousness (Cpr), the connections with fearfulness, social boldness and prudence having very small weights. Moreover many of its neighbors are connected with each other. Even if the edges incident in node Hmo were blocked, its neighbors would be nonetheless connected to each other directly or by a short path. Modesty therefore does not seem to play a very important unique role in the overall personality network.

Transitivity and small-worldness. The function `smallworldness` computes the small-worldness index (Humphries & Gurney, 2008). First the function converts the network to an unweighted one, which considers only the presence or the absence of an edge. Then the average path length and the global transitivity of the network are computed and the same indices are calculated on B=1000 random networks, with the same degree distribution of the focal network. The resulting values are entered in the computation of the small-worldness index. The output includes the small-worldness index, the transitivity of the network, and its average path length. It also returns summaries of the same indices computed on the random networks: the mean value and the .005 and .995 quantiles of the distribution. Function `set.seed` can be used to ensure the exact replicability of the results. The function requires the network as input and it is optionally possible to set the values of three parameters, `B`, `up` and `lo`, which are respectively the number of random networks and the upper and lower probabilities for the computation of the quantiles.

```
set.seed(100)
smallworldness(network)
```

The small-worldness value for our network is 1.01. An inspection of the values of transitivity and of average path length shows that they are not significantly different from those emerged from similar random networks. Therefore we may conclude that this personality network does not show a clear small-world topology.

Emerging insights. In this section, we showed how it is possible to perform a network analysis on a real personality dataset. We identified the most central nodes and edges, discussed centrality in the light of clustering coefficient and investigated some basic topological properties of the network, such as the small-world property. Two nodes resulted particularly central in the network and were the facet prudence of conscientiousness (Cpr) and the facet fairness of honesty-humility (Hfa).

Our network did not show the small-world property. The absence of a strong transitivity means that the connection of two nodes with a common neighbor does not increase the probability of a connection between themselves. The absence of a particularly short path length implies that it is not generally possible for any node to influence any other node using a short path. This result is not in line with the small-worldness property that emerged in the DSM-IV network reported by
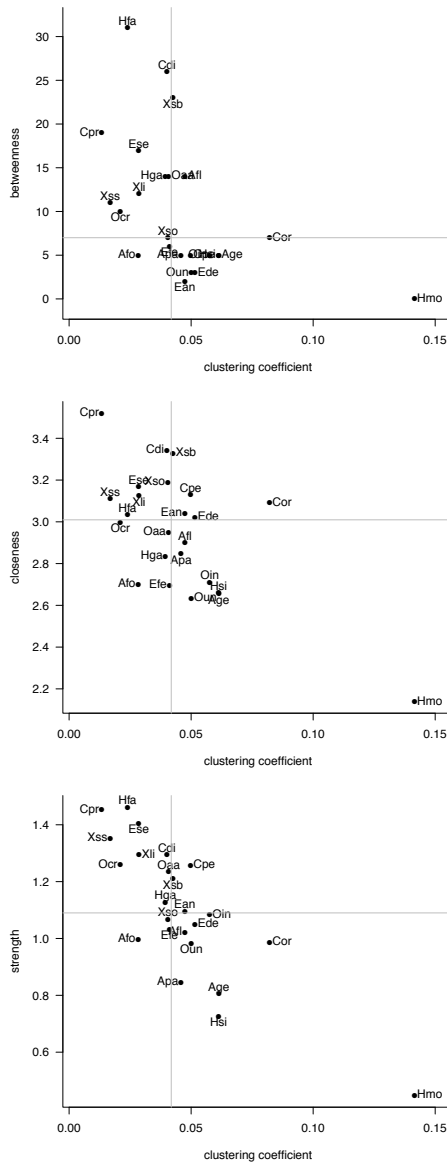
Figure 10.4: Centrality and clustering coefficient. The horizontal and the vertical lines represent the median values of centrality and clustering coefficient respectively. The closeness values are multiplied by 1000.
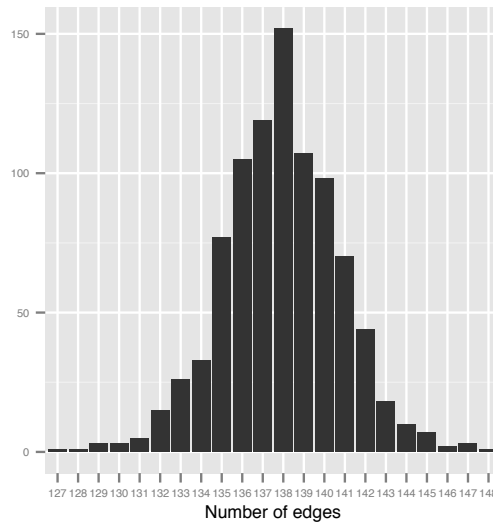
Figure 10.5: Histogram of the number of edges estimated in 900 replications of the adaptive LASSO.

Borsboom et al. (2011). It has been hypothesized that the small-world property might be at the basis of phenomena connected to the comorbidity that arise in psychopathology (Cramer et al., 2010); this also may simply not be a property of normal personality. This difference could reflect the fact that different personality characteristics represent distinct systems, while psychopathology systems seem to be more integrated. This result may be also attributable to the strategies that were used for defining this network and the DSM-IV network and may have been influenced by the particular personality scales under study. Future research may be directed towards the question of what network structure characterizes normal versus abnormal personality.

Stability of Results

The adaptive LASSO chooses the LASSO penalty parameter based on k-fold cross-validation, subdividing the dataset in k (10 by default) random samples. Because of this, under different random seeds slightly different network structures will be obtained. To investigate the stability of the results discussed in this section, we repeated the network estimation procedure 900 times under different random seeds and recomputed the strength, closeness and betweenness centrality measures and the signed versions of the clustering coefficients proposed by Zhang and by Onnela. The codes to replicate these findings can be found in the supplementary materials.

Visually the resulting graphs looked remarkably similar and only differed in the weakest edges in the graph. Figure 10.5 shows a histogram of the amount of nonzero connections present in each of the replications; the median amount of
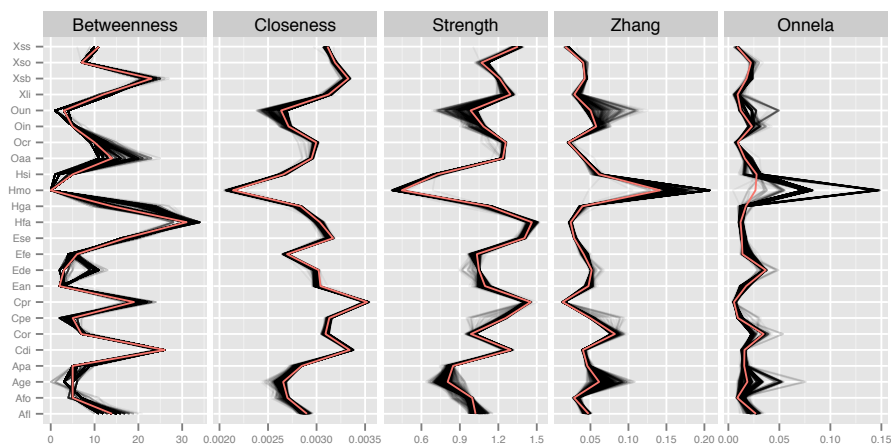
Figure 10.6: Estimated centrality and clustering coefficients under 900 replications of the adaptive LASSO. The colored line represents the results discussed in the chapter.

estimated edges was 138. Figure 10.6 shows the estimated centrality and clustering coefficients for both the graph used in the analyses (colored line) and the 900 replications (vague gray lines). It can be seen that overall the measures are stable across different replications. Among the three centrality measures, more stable results were obtained for closeness and strength than for betweenness. Between the clustering coefficients we can see that Zhang's clustering coefficient is much more stable than Onnela's; in Onnela's clustering coefficient especially the Hmo node shows divergent behavior. This behavior is due to the number small of connections of Hmo obtained in each replication, ranging from 3 to 11 (M = 3.96, SD = 0.64). Onnela's clustering coefficient is scaled to the number of connections regardless of weight. Therefore the relatively small difference in connections can have a large impact on this clustering coefficient.

From these results, we advise that Zhang's clustering coefficient should be preferred over Onnela's clustering coefficient in adaptive LASSO networks. Furthermore, we advise the reader to replicate these measures under different random seeds and to check for the stability of the results before substantively interpreting them.

10.4   Conclusion

Network approaches offer a rich trove of novel insights into the organization, emergence, and dynamics of personality. By integrating theoretical considerations (Cramer et al., 2010), simulation models (Mõttus, Penke, Murray, Booth, & Allerhand, 2014; Van Der Maas et al., 2006), and flexible yet user-friendly data-analytic techniques (Epskamp et al., 2012), network approaches have potential to

achieve a tighter fit between theory and data analysis than has previously been achieved in personality research. At the present time, the basic machinery for generating, analyzing, and simulating networks is in place. Importantly, the R platform offers an impressive array of packages and techniques for the researcher to combine, and most of the important analyses are currently implemented. We hope that, in the present chapter, we have successfully communicated the most important concepts and strategies that characterize the approach, and have done so in such a way that personality researchers may benefit from using network modeling in the analysis of their own theories and datasets.

In the present chapter, we have applied network modeling to an illustrative dataset, with several intriguing results that may warrant further investigation. However, we do stress that many of our results are preliminary in nature. The primary reason for this is that current personality questionnaires are built according to psychometric methodology that is tightly coupled to factor analysis and classical test theory (Borsboom, 2005). This makes their behavior predictable from pure design specifications, which in turn limits their evidential value. That is, if one makes the a priori decision to have, say, 10 items per subscale, and selects items on the basis of their conformity to such a structure, many of the correlations found in subsequent research are simply built into the questionnaire. Therefore, it is hardly possible to tell to what extent results reflect a genuine structure, or are an artifact of the way personality tests are constructed. Trait perspectives are not immune to this problem, as in some cases the factors of personality may simply appear from questionnaire data because they have been carefully placed there. Future research should investigate potential solutions to this issue, for instance by considering variable sets consisting of ratings on the familiar personality-descriptive adjectives of a language, as in lexical studies (e.g., Ashton & Lee, 2005, 2007; De Raad et al., 2014; Goldberg, 1990b; Saucier et al., 2014), and by comparing the characteristics of such networks to networks that emerge from questionnaire data.

An interesting question is whether all individuals are scalable on all items, as current methodology presumes. It is entirely possible, if not overwhelmingly likely, that certain items assess variables that simply do not apply to a given individual. Current psychometric methods have never come to grip with the "n.a." answer category, and in practice researchers simply force all individuals to answer all items. In networks, it is easier to deal with the n.a.-phenomenon, as nodes deemed to be inapplicable to a given person could simply be omitted from that person's network. This would yield personality networks that may differ in both structure and in size across individuals, an idea that resonates well with the notion that different people's personalities might in fact be also understood in terms of distinct theoretical structures (Borsboom et al., 2003; Cervone, 2005; Lykken, 1991). The application of experience sampling methodology and of other ways to gather information on dynamical processes personality may also offer an inroad into this issue (Fleeson, 2001; Hamaker, Dolan, & Molenaar, 2005; Bringmann et al., 2013).

The notion that network structures may differ over individuals, and that these differences may in fact be the key for understanding both idiosyncrasies and communalities in behavior, was illustrated in the simulation work reported by

Costantini and Perugini (2014). Future research might be profitably oriented to questions such as (a) what kind of structural differences in networks could be expected based on substantive theory, (b) how such differences relate to well-established findings in personality research (see also Mõttus et al., 2014), (c) which network growth processes are theoretically supported by developmental perspectives. Of course, ultimately, such theoretical models would have to be related back to empirical data of the kind discussed in the data-analysis part of this chapter; therefore, a final highly important question is to derive testable implications from such perspectives. This includes the investigation of how we can experimentally or quasi-experimentally distinguish between explanations based on latent variables, and explanations based on network theory.

Ideally, these future developments are coupled with parallel developments in statistical and technical respects. Several important extensions of network models are called for. First, in this work we focused on the adaptive lasso, which is an effective method to extract a network from empirical data that has been profitably used in other fields (Krämer et al., 2009). However network analysis is a field in rapid evolution and alternative methods are being developed and studied. Among these, we consider particularly promising the graphical lasso (Friedman et al., 2008), for which adaptations exist that take into account the presence of latent variables in the network (Chandrasekaran et al., 2012; Yuan, 2012). Alternative methods based on Bayesian approaches have also been proposed and implemented (Mohammadi, Wit, et al., 2015). Further research is needed to systematically compare these and other methods in the complex scenarios that are usually encountered in personality psychology. Second, as noted in this chapter, many network analytics were originally designed for unweighted networks. Although some of the relevant analyses have now been extended to the weighted case (see Boccaletti et al., 2006; Opsahl et al., 2010; Costantini & Perugini, 2014, several other techniques still await such generalization. One important such set of techniques, which were also illustrated in the present work, deals with the determination of network structure. Both the theoretical definition of global structures, such as in terms of small-worlds, scale-free networks (Barabási, 2009), and random networks, and the practical determination of these structures (e.g., through coefficients such as small-worldness or through fitting functions on the degree distribution) are based on unweighted networks. It would be highly useful if these notions, and the accompanying techniques, would be extended to the weighted network case. Another technical improvement that should be within reach is how to deal with data that likely reflect mixtures of distinct networks. In the case of time series data, such approaches have already been formulated through the application of mixture modeling (Bringmann et al., 2013); however, statistical techniques suited to this problem may also be developed for the case of cross-sectional data. The issue is important in terms of modeling idiosyncrasies in behavior, but may also be key in terms of relating normal personality to psychopathology (Cramer et al., 2010). Naturally, this includes the question of how we should think about the relation between normal personality and personality disorders.

Unified Visualizations of Structural Equation Models

Abstract

Structural Equation Modeling (SEM) has a long history of representing models graphically as path diagrams. This chapter presents the freely available semPlot package for R, which fills the gap between advanced, but time-consuming, graphical software and the limited graphics produced automatically by SEM software. In addition, semPlot offers more functionality than drawing path diagrams: it can act as a common ground for importing SEM results into R. Any result useable as input to semPlot can be also represented in any of the three popular SEM frameworks, as well as translated to input syntax for the R packages sem (Fox, 2006) and lavaan (Rosseel, 2012). Special considerations are made in the package for the automatic placement of variables, using three novel algorithms that extend earlier work of Boker, McArdle, and Neale (2002). The chapter concludes with detailed instructions on these node-placement algorithms.

11.1 Introduction

The semPlot package for the freely available statistical programming language R (R Core Team, 2016) extends various popular structural equation modeling (SEM) software packages with a free, easy to use and flexible way of producing high quality graphical model representations—commonly termed *path diagrams*—as well as providing a bridge between these software packages and the main SEM frameworks.

A path diagram utilizes a network representation, in which variables are represented as nodes—square nodes indicating manifest variables, circular nodes indicating latent variables and triangular indicating constants—and relations between

---

This chapter has been adapted from: Epskamp, S. (2015). semPlot: Unified visualizations of Structural Equation Models. *Structural Equation Modeling, 22*, 474–483.

variables are represented by a set of unidirectional and bidirectional edges, which typically represent regression equations and (co)variances respectively.

Currently there are two common ways of drawing path diagrams. Many of the available SEM software packages have an option to display the path diagram graphically, either directly in the package (e.g., LISREL; Jöreskog & Sörbom, 1996), by creating syntax for external network drawing software (e.g., sem; Fox, 2006) or through third party extensions (e.g., Lispath; Marcoulides & Papadopoulos, 1993). Some packages in addition allow the model to be specified in a graphical way, by letting the user draw the path diagram directly in an interactive command window (e.g., Amos; Arbuckle, 2010, MPlus; Muthén & Muthén, 1998–2012, PLSgraph; Chin, 2001, and Onyx; von Oertzen, Brandmaier, & Tsang, 2013). Alternatively, instead of generating a path diagram from a given model, the path diagram can also be drawn manually, using many free and commercial software packages (e.g., Cytoscape; Shannon et al., 2003, Microsoft® Powerpoint® and igraph; Csardi & Nepusz, 2006).

Both of these methods, however, have important limitations. The path diagrams created by SEM packages produces path diagrams that are hardly customizable, and produce images unsuited for publication. On the other hand, manually drawing path diagrams in external software can take a very long time and is prone to error. The semPlot package offers a middle way; it is designed to automatically produce high quality path diagrams from the output of various popular SEM software packages, while retaining a high level of customizability. Thus, in semPlot, the user feeds a raw output file to the program, which then returns a high-quality image ready for publication. In addition, as will be described below, semPlot creates an internal model representation that can serve as a translator between SEM programs; for instance, on the basis of, say, LISREL model *output*, semPlot automatically generates the corresponding lavaan (Rosseel, 2012) *input*.

The semPlot package supports the output from R packages sem (Fox, 2006), lavaan (Rosseel, 2012), OpenMx (RAM specification only; Boker et al., 2011) and standalone software MPlus (Muthén & Muthén, 1998–2012, using R package MplusAutomation for the import; Hallquist & Wiley, 2013), LISREL (Jöreskog & Sörbom, 1996, using R package lisrelToR for import; Epskamp, 2013) and Onyx. Several base R functions for related statistical techniques such as exploratory factor analysis and general linear modeling are also supported. In addition, semPlot can also be used without the need of fitting a SEM using the lavaan modeling syntax, or matrix specification according to the RAM (McArdle & McDonald, 1984), LISREL (Hayduk, 1987) and Mplus (Muthén, 1998–2004) modeling frameworks.

The graphs that semPlot produces are drawn using the qgraph package, which itself is designed as a network drawing package aimed at applications in statistical visualizations. Customization of the graphs can be done either via semPlot itself (using many options designed for SEM models, such as omitting exogenous variances) or post-hoc via the qgraph package (using options designed to visualize graphs, such as manually recoloring edges).

This chapter consists of two sections: the first section describes the functionality of the package and in the second section describes the algorithms used for automatically constructing a path diagram.
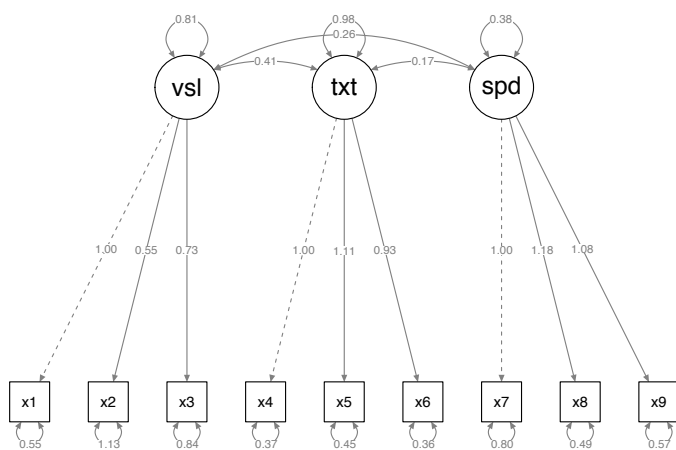
(a)
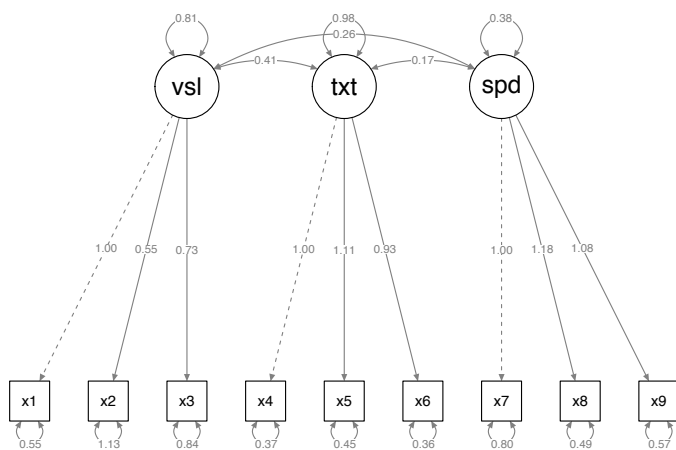
(b)

Figure 11.1: Generated path diagram of the Holzinger-Swineford CFA example. Panel (a) shows a visualization of the path diagram with estimates as labels and Panel (b) shows a visualization of the standardized parameter estimates.

11.2 General Use of the semPlot Package

The semPlot package can be downloaded from CRAN or installed directly in R:

```
install.packages("semPlot")
```

After which the package can be loaded:

```
library("semPlot")
```

223

This will load the functions from the semPlot package into R.

Drawing Path Diagrams

The `semPaths` function can be used to plot path diagrams and visualize (standardized) parameter estimates. It takes as first argument either a SEM object (from R packages) or a string indicating the location of an output file from external SEM software (MPlus or LISREL). The second and third arguments can be assigned strings indicating what the edge color and label respectively indicate. For example, the following code plots a model where the edges are colored according to standardized values and the edge labels indicate the unstandardized estimates:

```
semPaths(input, "standardized", "estimates", ...)
```

Where `...` indicate any number of other arguments controlling the output which are further explained in the package manual:

```
?semPaths
```

To illustrate this, one could use one of the lavaan package documentation examples to compute a confirmatory factor analysis (CFA) on the famous Holzinger and Swineford (1939) example:

```
library("lavaan")
example(cfa)
```

Next, sending the resulting `fit` object to `semPaths` plots a path diagram of the model with parameter estimates on the labels:

```
semPaths(fit, "model", "estimates")
```

We could also visualize the parameter estimates by coloring positive parameters green or red indicating positive or negative estimates and varying the width and color of an edge to indicate the strength of the estimate (see Chapter 9). This works best with standardized parameters:

```
semPaths(fit, "standardized", "hide")
```

The resulting graphs can be seen in Figure 11.1. This figure also shows that fixed parameters—in this case scaling by fixing factor loadings—are visualized by default by using dashed lines.

The semPlot package can handle larger complicated measurement models. The next example is based on the Mplus output of the multilevel factor analysis model as described by Little (2013), in which the factor structure of the Life Skills Profile-16 (LSP-16) was assessed. The following codes produce the two plots in Figure 11.2:

```
semPaths(file.choose(), "model", "estimates",
         style = "lisrel", curve = 0.8, nCharNodes = 0,
         sizeLat = 12, sizeLat2 = 6, title = TRUE,
         mar = c(5, 1, 5, 1), edge.label.cex = 0.5)
```
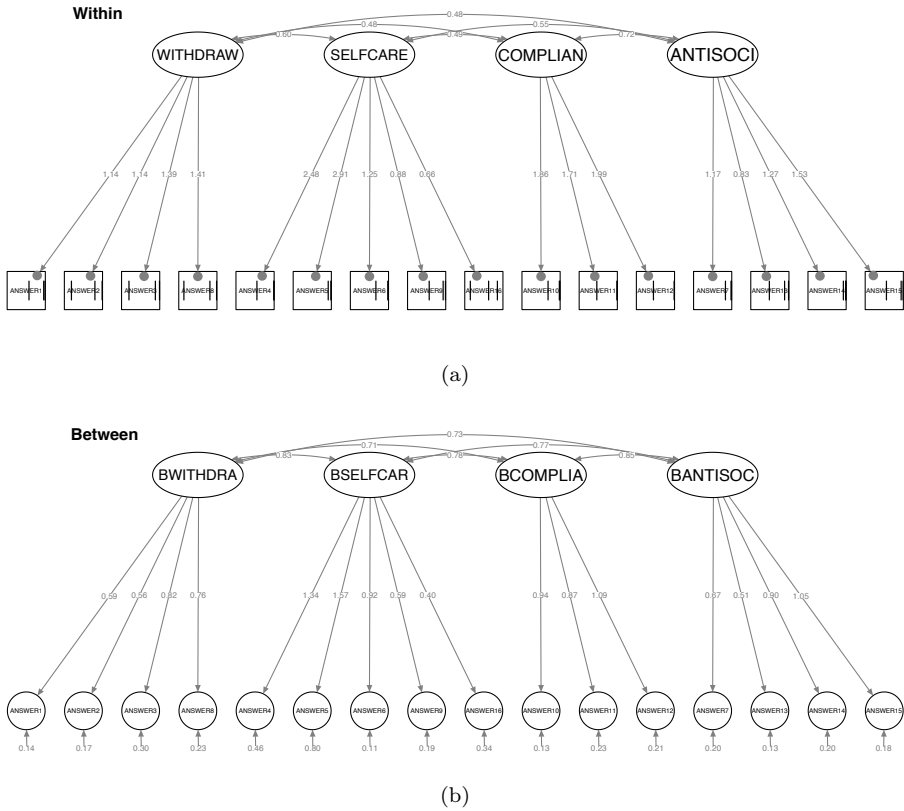
(a)

(b)

Figure 11.2: Generated path diagram for multilevel factor analysis model of LSP-16. Panel (a) shows the within-cluster model, with vertical bars representing the estimated thresholds of each of the ordinal variables. Panel (b) shows the between-cluster model.

In which `file.choose()` is a base R function that opens a convenient file browser to select the Mplus output file.

Figure 11.2 shows that two plots are now generated: one indicating the within-cluster model and one indicating the between-cluster model. In the within-cluster model the closed orbs inside manifest indicate random intercepts and the vertical bars inside the manifest variables indicate the estimated thresholds; in the between-cluster model the indicators are represented by a circle for random intercepts.

The argument `style = "lisrel"` specifies that (residual) variances are plotted similar to the way LISREL plots these: as arrows without origin on endogenous variables only. The default, `style = "ram"`, would plot these residuals as described by Boker et al. (2002): as double-headed self-loops on both endogenous and exogenous variables. To illustrate this consider an example of the famous 'Industrialization and Political Democracy' dataset used by Bollen (1989), which

225

has been implemented as example in the Lavaan package:

```
library("lavaan")
example(sem)
semPaths(fit, "model", "hide", style = "lisrel",
        rotation = 2)
semPaths(fit, "model", "hide", style = "ram", rotation = 2,
        cardinal = "man cov")
```

The resulting graphs can be seen in Figure 11.3.

Color can also indicate equality constrains: by coloring parameters that are constrained to be equal with the same color (unconstrained parameters are still colored gray)—especially useful in identifying the different steps in assessing measurement invariance (Meredith, 1993). For example, the semTools package (Pornprasertmanit, Miller, Schoemann, & Rosseel, 2013) can be used to test for measurement invariance using lavaan on the Holzinger and Swineford (1939) example:

```
library("semTools")
fits <- example(measurementInvariance)
semPaths(fits$value$fit.intercepts, "equality", "estimates",
        sizeLat = 5, title = FALSE, ask = FALSE,
        levels = c(1, 2, 4), edge.label.cex = 0.5)
```

Figure 11.4 shows one of the steps in testing for measurement invariance: strict measurement invariance with free factor means. It can be seen that the factor loadings and intercepts are constrained to be equal over groups, but the factor means and variances are not.

Investigating Correlational Structures

SEM models are usually fit by comparing the observed covariances to the model implied covariances. The qgraph package used as back-end to semPlot supplies a novel framework for visualizing correlational structures as networks (as is described in Chapter 9): a correlation matrix can be visualized as a network in which each variable is represented by a node and each correlation as a weighted edge between two nodes.

In the semPlot package, the `semCors` function visualizes the model implied correlation matrix (which is either provided as input or computed from data) and the observed correlation matrix (must be provided as input) using qgraph with parameters automatically chosen such that the graphs are comparable. To illustrate this, consider the following simulated dataset (using lavaan):

```
library("lavaan")

Mod <- '
A =~ 1*a1 + 0.6*a2 + 0.8*a3
B =~ 1*b1 + 0.7*b2 + 0.9*b3
a1 ~~ 1*b1
A ~~ -0.3* B
```
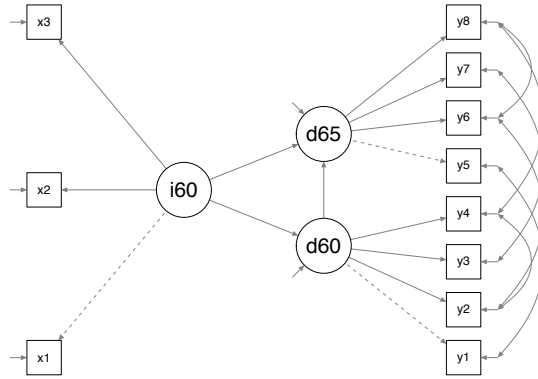
(a)
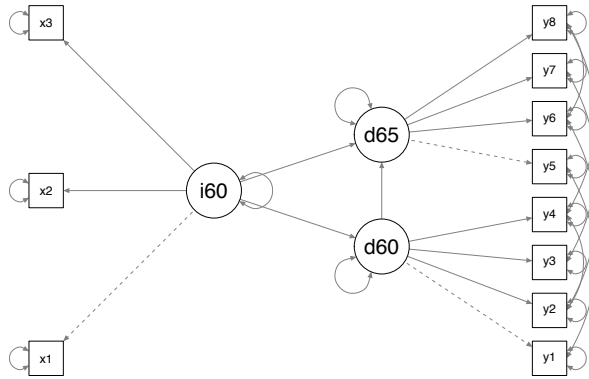
(b)

Figure 11.3: Generated path diagrams for Industrialization and Political Democracy dataset example. Panel (a) shows the path diagram with residuals drawn in 'lisrel' style and Panel (b) shows the path diagram with residuals drawn in 'ram' style.
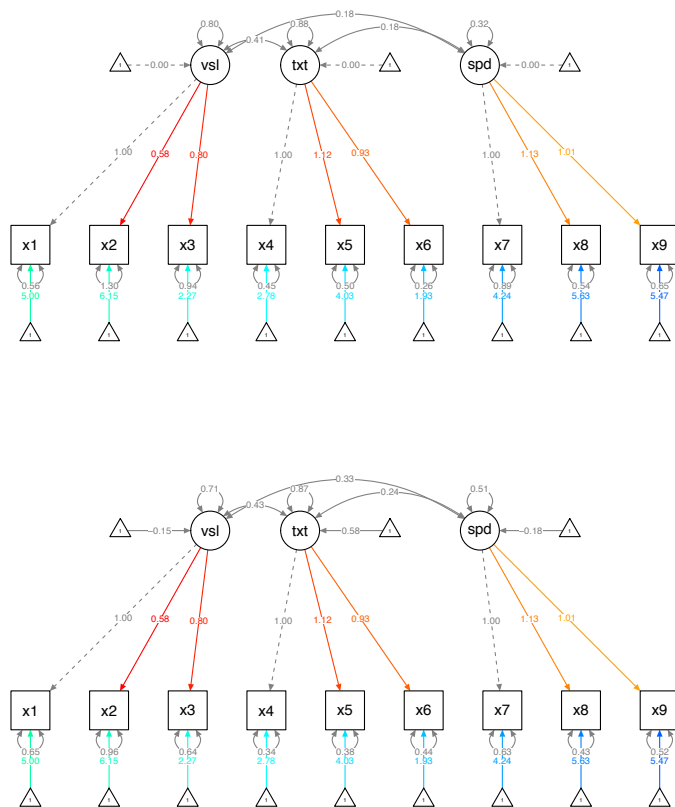
Figure 11.4: Path diagrams for two groups in the Holzinger-Swineford CFA example, testing for strict measurement invariance with free factor means.

'

```
set.seed(5)
Data <- simulateData(Mod)
```

This dataset, called `Data`, is simulated under a two-factor model with two negatively correlated factors. However, the residuals of the first indicator of each factor are strongly positively correlated. After fitting a general CFA model to this data, not including the residual correlation, the implied and observed correlation matrices can be inspected:

```
Mod <- '
A =~ a1 + a2 + a3
B =~ b1 + b2 + b3
```
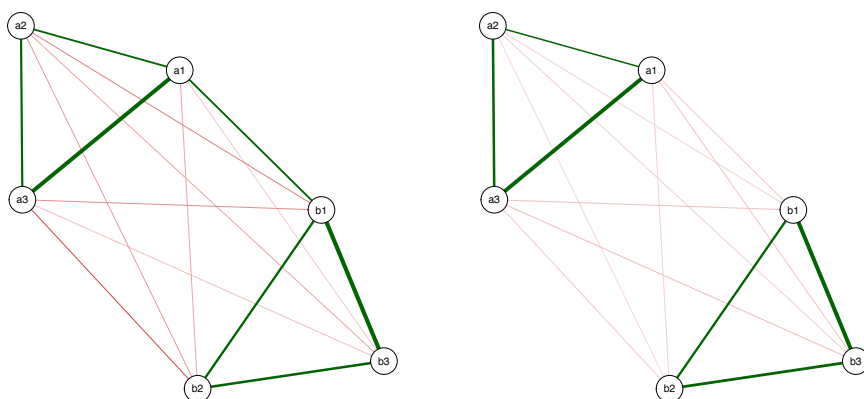
Figure 11.5: Observed (left) and model implied (right) correlation matrices of simulated data example.

```
’
fit <- cfa(Mod, data=Data)
semCors(fit, layout = "spring", cut = 0.3, esize = 20)
```

Figure 11.5 shows that the observed and implied correlation matrices are very similar except for the correlation between `a1` and `b1`, which cause the misfit in this model. This provides a visual way of judging the fit of a SEM model and a way of seeing where misfit is occurring.

Linking SEM Software Packages and Models

An important design philosophy of semPlot is unifying different SEM software packages in a freely available interface. To this end, the package can also be used as bridge between different SEM software packages and SEM models. First, the `semSyntax` function generates model syntax for R packages sem and lavaan given any input supported in semPlot. For example, the output file of example 5.1 of the MPlus user guide (Muthén & Muthén, 1998–2012) can be imported:

```
ex5.1 <- tempfile(fileext = ".out")
url <- "http://www.statmodel.com/usersguide/chap5/ex5.1.out"
download.file(url, ex5.1)
```

Next, the file can be used to generate a model to use in the lavaan package:

```
lavMod <- semSyntax(ex5.1, "lavaan")
## Model <- ’
## F1 =~ 1*Y1
## F1 =~ Y2
## (...)
```

```
## Y5 ~~ Y5
## Y6 ~~ Y6
## '
```

The function returns an object, and prints the R script needed to create this object. A useful application of this bridge is to simulate data in R given any SEM output file using lavaan's `simulateData` function. To do this, first specify the model with all estimated parameters set to fixed:

```
lavMod <- semSyntax(ex5.1, "lavaan", allFixed = TRUE)
```

Next the model can be sent to `simulateData`:

```
head(simulateData(lavModFixed))
##           Y1       Y2       Y3       Y4       Y5       Y6
## 1   0.88695   0.2414   0.8060   0.6778   1.2228  -0.34377
## 2   1.30715  -0.4904   0.8651   0.4772   0.4611   0.58303
## 3  -0.62939  -1.5140  -0.3916   1.0225   1.2060  -0.65448
## 4   0.99210  -1.8682  -1.0856   0.3514  -0.3357  -2.01952
## 5   0.02836  -0.4113  -0.3776  -1.1781   0.1050  -1.23260
## 6   1.12654   1.9011   1.0472   0.6976  -0.8670  -0.03874
```

Second, the `semMatrixAlgebra` The `semMatrixAlgebra` function offers a unified interface for extracting model matrices of any of the three major SEM frameworks, RAM (McArdle & McDonald, 1984), LISREL (Hayduk, 1987) and Mplus (Muthén, 1998–2004), using any of the supported input software packages. For example, the RAM framework uses three model matrices: $\boldsymbol{A}$, $\boldsymbol{S}$ and $\boldsymbol{F}$:

$$\boldsymbol{v} = \boldsymbol{A}\boldsymbol{v} + \boldsymbol{u}$$
$$\boldsymbol{u} \sim N(\boldsymbol{0}, \boldsymbol{S})$$
$$\text{Var}(\boldsymbol{v}) = \boldsymbol{F}(\boldsymbol{I} - \boldsymbol{A})\,\boldsymbol{S}\,(\boldsymbol{I} - \boldsymbol{A})^{-1\top}\,\boldsymbol{F}^{\top}$$

In which $\boldsymbol{v}$ is a vector containing both manifest and latent variables, $\boldsymbol{A}$ a matrix of regression parameters (usually termed the asymmetric matrix), $\boldsymbol{S}$ a matrix of (residual) variances (usually termed the symmetric matrix) and $\boldsymbol{F}$ (usually termed the filter matrix) can be used to distinguish between latent and manifest variables. `semMatrixAlgebra` can be used to extract e.g., the $\boldsymbol{A}$ matrix of Mplus user guide example 5.1:

```
semMatrixAlgebra(ex5.1, A)
##          F1      F2 Y1 Y2 Y3 Y4 Y5 Y6
## F1 0.000  0.000   0  0  0  0  0  0
## F2 0.000  0.000   0  0  0  0  0  0
## Y1 1.000  0.000   0  0  0  0  0  0
## Y2 1.126  0.000   0  0  0  0  0  0
## Y3 1.019  0.000   0  0  0  0  0  0
## Y4 0.000  1.000   0  0  0  0  0  0
## Y5 0.000  1.059   0  0  0  0  0  0
## Y6 0.000  0.897   0  0  0  0  0  0
```

Note that the use of `A` automatically let `semMatrixAlgebra` detect that we are interested in the RAM framework specifically. Requesting matrices from other frameworks, such as the $\mathbf{\Lambda}$ matrix—containing factor loadings—from the MPlus modeling framework, works in the same way:

```
semMatrixAlgebra(ex5.1, Lambda)
##       F1    F2
## Y1 1.000 0.000
## Y2 1.126 0.000
## Y3 1.019 0.000
## Y4 0.000 1.000
## Y5 0.000 1.059
## Y6 0.000 0.897
```

The `semMatrixAlgebra` function cannot only be used for extracting individual model matrices but also for extracting the result of algebraic computations using these model matrices. For example, one could compute the implied covariances on the same example model as follows—using helper function `Imin(A,TRUE)` to compute $(\mathbf{I} - \mathbf{A})^{-1}$:

```
semMatrixAlgebra(ex5.1,
  F %*% Imin(A,TRUE) %*% S %*% t(Imin(A, TRUE)) %*% t(F))
##           Y1       Y2       Y3       Y4       Y5       Y6
## Y1  1.97100  1.02128  0.92423 -0.03000 -0.03177 -0.02691
## Y2  1.02128  1.94796  1.04069 -0.03378 -0.03577 -0.03030
## Y3  0.92423  1.04069  1.95179 -0.03057 -0.03237 -0.02742
## Y4 -0.03000 -0.03378 -0.03057  2.05000  0.80484  0.68172
## Y5 -0.03177 -0.03577 -0.03237  0.80484  1.70633  0.72194
## Y6 -0.02691 -0.03030 -0.02742  0.68172  0.72194  1.67750
```

`semMatrixAlgebra` returns the results in a list rather than a single matrix if the model contains multiple groups.

11.3 Algorithms for Drawing Path Diagrams

When drawing a path diagram the variables need to be placed in a structured way, such that the diagram is easily interpretable (Boker et al., 2002). Manually defining such a graph layout can be tedious and time-consuming work; an automated solution to placing variables would work best in most situations. This section introduces three novel algorithms—which are implemented in semPlot—that can be used to automatically place variables such that complex SEM models are easily interpretable.

The three layout algorithms are each designed to place variables in a tree-like structure next to each other on horizontal levels. They are chosen such that first, the structural part of the model—especially the relationship between exogenous and endogenous variables—is clearly visible; and second, Indicators of a latent

variable are placed next to each other and either below or above the latent variable. To achieve this, all three algorithms start with exogenous variables[1] or their indicators placed at the top level of the graph (level 0) and expand downwards to the bottom of the graph (level n).

The first algorithm is based on the way the LISREL program (Jöreskog & Sörbom, 1996) plots path diagrams. In this algorithm variables are placed on one of four horizontal levels. The top level contains manifest variables that are either exogenous themselves or only indicators of exogenous latent variables. The second level contains latent variables that are either exogenous themselves or regressed only on exogenous manifest variables. The third level contains all other (endogenous) latent variables and the fourth level contains all other (endogenous) manifest variables. Intercepts can be added by placing a representation of the unit vector next to or below/above each variable. In defining the horizontal placement the latent variables are placed in the order they appear in the model, and manifest variables are placed such that they are closest to latent variables they are connected to.

The second algorithm is a variation of the Reingold-Tilford algorithm (Reingold & Tilford, 1981) which places variables in a tree structure originating from a set of user defined root nodes at the top. The igraph package (Csardi & Nepusz, 2006) can be used to compute the Reingold-Tilford algorithm. However, in the presence of intercepts, exogenous latents, or covariances, this algorithm does not produce proper diagram structures out of the box. To solve this, the algorithm is applied to a modified version of the network representation of the model: by removing all arrows (making edges undirected) and removing all covariances. Through a specific choice of root variables, a tree structure is obtained in which exogenous variables are placed on top and endogenous variables at the bottom.

Finally, the third algorithm uses a variation of the placement algorithm described by Boker et al. (2002). This algorithm computes for each node the longest outgoing path, and places nodes accordingly on horizontal levels from highest (top) to lowest (bottom) longest outgoing path-length. For more stable results (e.g., indicators of exogenous latents should be placed above the latent), this algorithm can be enhanced by not using the original network representation of a model but one in which the direction of the edges between exogenous latent variables and their indicators is reversed and all double-headed edges (covariances) are removed.

In all three algorithms, horizontal levels that do not contain any nodes are not included in the graph, and if there are only exogenous latent variables and no regressions between manifest variables (e.g., factor analysis models) the layout is flipped. In cases which feature many indicators per latent variable, it is more useful to place variables in a circle-like fashion; here, the origin of the tree placement is not at the top, expanding to the bottom, but at the center, expanding outward. To do this, we may transform the horizontal levels to nested circles; the higher the level, the smaller the circle.

Often, the structural part of a model—containing only regressions between latent variables—is the only part that requires specifically thoughtful placement of variables; for the measurement parts—the factor loadings of indicators on each

---

[1]A variable is treated as exogenous if it has no incoming directed edges attached.
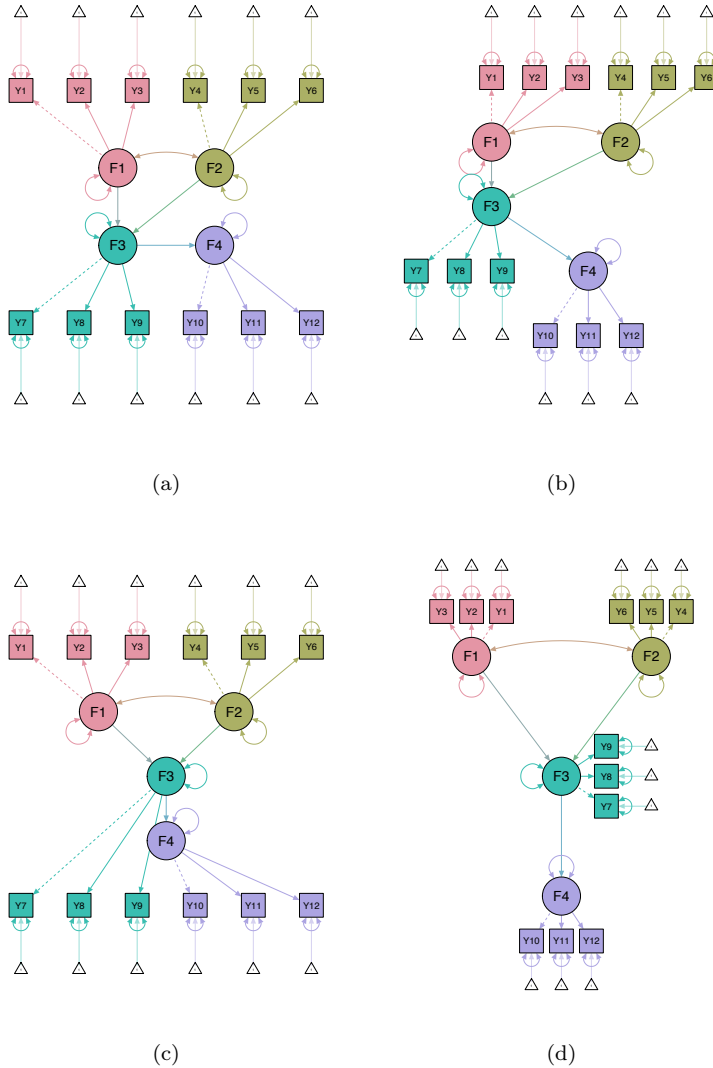
(a)

(b)

(c)

(d)

Figure 11.6: Path diagram including parameter estimates of example 5.25 of the Mplus user guide. Panel (a) shows default placement, panel (b) the adjusted Reingold-Tilford algorithm, panel (c) the adjusted Boker-McArdle-Neale algorithm and panel (d) a layout where only the structural part is based on the adjusted Boker-McArdle-Neale algorithm and the measurement sub-models are drawn around the latent variables.

latent variable—indicators simply need be placed on a straight line under, over, or next to the latent. To this end, it might not be necessary to run a complicated placement algorithm over all variables, but rather only over the structural part of a model, followed by placing indicators near the latent. Such a placement of nodes for the structural part of a model could be used on the basis of any of the above mentioned algorithms, but also through any network drawing algorithm (e.g., by using a force-embedded algorithm; Fruchterman & Reingold, 1991).

In `semPaths`, the `layout` argument can be used to control which algorithm is used to define the placement of the nodes. This argument can be set to `"tree"` to obtain the default layout, `"tree2"` to obtain the adjusted Reingold-Tilford algorithm or `"tree3"` to obtain the adjusted Boker-McArdle-Neale algorithm. To obtain circular versions of these algorithms, `"circle"`, `"circle2"` and `"circle3"` can be used. To split the layout algorithm for structural and measurement models, the `layoutSplit` argument can be used. Finally, the `layout` argument can also be used to manually define the placement of nodes (see package documentation for examples). Figure 11.6 shows the result of these algorithms on example 2.25 from the MPlus user's guide (Muthén & Muthén, 1998–2012).

11.4 Conclusion

The semPlot package extends many popular SEM software packages with advanced visualization functions. These functions can be used to display specified models, parameter estimates, model constraints, and implied correlation structures. Furthermore, semPlot provides a bridge between these software packages and different modeling frameworks. The package uses several novel algorithms for automatic placement of variables in the path diagrams and allows for detailed manual customizations[2].

semPlot is sufficiently user-friendly to be used by researchers with limited experience in R, while it presents more advanced users with a broad scope of functionality and flexibility. Several features are open to further development. First, the use of semPlot key be extended in various ways—such as though web interfaces (RStudio & Inc., 2013). Second, support is to be added for additional SEM software packages such as Amos (Arbuckle, 2010), EQS (Bentler & Wu, 1993, using the REQS R package; Mair & Wu, 2012) and R packages semPLS (Monecke & Leisch, 2012) and lava (Holst & Budtz-Joergensen, 2013). The developmental version of semPlot is available at GitHub, `http://github.com/SachaEpskamp/semPlot`, where new ideas for the package can also be submitted.

---

[2]See for detailed instruction the package website: http://sachaepskamp.com/semPlot

Part IV

Conclusion

Chapter 12

Discussion: The Road Ahead

12.1 Introduction

This dissertation provided an overview of network models applicable to psychological data as well as descriptions of how these methods relate to general psychometrics. The visualization methods outlined in the final part of this dissertation are based on the oldest publications and relate to the state-of-the-art when this PhD project started. At the start of this PhD project, 4 years ago, network estimation in psychology consisted of not much more than drawing networks based on marginal correlation coefficients. This can be shown in publications from this period. Cramer et al. (2010) marks the first psychological network estimated from data and shows a network in which edges are based on associations. The *qgraph* package was based on this and, for the first time, provided psychologists with a simple method for constructing networks based on correlations (Epskamp et al., 2012). Key publications of that time mostly outlined conceptual and theoretical implications of the network perspective and often relied on correlation networks to showcase what such a network could possibly look like (e.g., Cramer, Sluis, et al., 2012; Borsboom & Cramer, 2013; Schmittmann et al., 2013). Partial correlation networks were proposed and published (e.g., Epskamp et al., 2012; Cramer, Sluis, et al., 2012) but were not yet worked out in enough detail to provide the powerful visualizations now used in psychology.[1] In addition, time-series models showed promise (e.g., Borsboom & Cramer, 2013) but had not yet been worked out in detail and implemented in easy-to-use software.

The use of network estimation on psychological data has come a long way since then. In fact, the achieved progress warrants the birth of a new field of research: *network psychometrics*. This progress has been marked by a gradual increase in the understanding of both the interpretation and the applicability of network models as well as by key turning points in the development of the methodology. Some of these key turning points came with the emergence of new software routines that

---

[1] In retrospect, the original promise of partial correlation networks might have been taken too strong. For example, we now know that the partial correlation network shown by Epskamp et al. (2012) consists of far too many nodes compared to the number of observations to likely lead to stable results.

make network estimation accessible to psychological researchers. In particular, the development of the *IsingFit* package (van Borkulo et al., 2014) and new versions of the *qgraph* package[2] changed network psychometrics from a conceptual framework to a concrete methodology that psychologists could readily apply. More recently, network psychometrics has further matured by the development of software packages that include data-driven statistical procedures which assess the properties of the estimated networks, such as comparing network structures of different samples (*NetworkComparisonTest*; van Borkulo, 2016), and assess the accuracy and differences in network properties, such as centrality indices (*bootnet*; Epskamp, Borsboom, & Fried, 2016; see Chapter 3). In addition, promising new software packages became available that allow for network estimation on time-series data, on multiple subjects (*mlVAR*; see Chapter 6), in clinical practice (*graphicalVAR*; see Chapter 5 for an example), and in datasets using variables of different distributions without the assumption of stationarity (*mgm*; Haslbeck & Waldorp, 2016a). Finally, the *lvnet* package (Epskamp, Rhemtulla, & Borsboom, 2016; see Chapter 7) marks the first software package that combines undirected network models with latent variable modeling.

We have come a long way, but there is still a long road ahead. As more and more technical details and conceptual interpretation of these models are worked out, more and more questions emerge. Network psychometrics is now being fleshed out as its own field of research—a field that many talented researchers are entering. Therefore, I wish to conclude this dissertation with an overview of open questions and potential future directions for this young field of science.

12.2   Open Questions in Network Psychometrics

Handling Missing Data and Ordinal Data

The methods used in network psychometrics mostly come from the fields of statistical learning, statistical physics, and econometrics. Data from such fields are very different from data typically found in psychology. Two properties of psychological data especially do not often occur in other fields of science: data with missing values and data on an ordinal scale of measurement. Network estimation can learn from a long history of handling such problems in psychometrics. As such, network psychometrics should focus on the two problems noted above. Both problems are far from trivial and will require substantive future research.

Missing data.   In psychology, missing data is usually not the exception but rather the norm. Network estimation methods, however, are not yet capable of handling missing data in an appropriate way. When estimating a GGM, an estimate of the variance–covariance matrix is used as input to the graphical LASSO (see Chapter 2). When data are missing, such an estimate can be obtained by deleting cases containing one missing value or by pairwise estimation. Typically,

---

[2]Version 1.2.5 (revamping the choice of cutoff selection in visualizing networks and introducing standardized centrality plots) and version 1.3 (introducing EBIC model selection of glasso networks; Foygel & Drton, 2010).

pairwise estimation is used; however, this relies on certain assumptions on why the data are missing (i.e., are the data missing at random or not) and might result in variance–covariance matrices that are not positive definite. In addition, it is questionable what the sample size (e.g., for EBIC model selection) is when the variance–covariance matrix is pairwise estimated. When estimating the Ising model, mixed graphical models, or time-series models, researchers often delete full cases as a method for handling missing data, but they lose a significant amount of information in the process.

Psychometricians have worked out in detail many ways of handling missing data in various modeling frameworks (Enders, 2001). Powerful methods involve (multiple) imputation techniques and full-information maximum likelihood (FIML) estimation. Such methods could, in theory, also be applied to network estimation but further research is needed. Klaiber, Epskamp, and van der Maas (2015) proposed imputation techniques to estimate the Ising model iteratively. First the model is fit to the data, then the data are imputed given the Ising model, then the model is fit to the imputed data, and so on until the parameter estimates are stable. In theory, FIML is possible for estimating the GGM (the GGM can be framed in terms of a typical SEM model; see Chapter 7), but this could only work for confirmatory models. Usually, regularization techniques such as the LASSO are applied in the estimation. Perhaps a penalized version of FIML can be worked out in future research, combining the strengths of FIML with LASSO estimation.

Ordinal data. Another well-known problem in psychometrics is the scale of measurement on which items are assessed (Stevens, 1946). Researchers seek to measure concepts that are not directly observable, such as the severity of a person's rumination, using psychological items. Such items are frequently measured on Likert scales and cannot readily be treated as continuous (Rhemtulla, Brosseau-Liard, & Savalei, 2012). This problem is especially prominent in data on psychopathological symptoms, often measured on a 4-point scale (e.g., Fried, van Borkulo, et al., 2016), ranging from 0 (*not present*) to 3 (*severe problems*). Often, these data are highly skewed (i.e., many people report 0, especially when a general population sample is used).

Although network psychometrics is often applied to ordinal data, the handling of such data should also be a topic of future research. Currently, no method of appropriately handling ordinal data exists. There are four methods often applied to handle such data, all of which can be problematic:

1. The method most commonly used is to compute polychoric correlations and to use these as input to the EBIC model selection of GGM networks using the graphical LASSO (see Chapter 2). This methodology, however, is not without problems. First, researchers employ the methodology to estimate the model in two steps, first by computing the polychoric variance–covariance matrix and next by treating this as the sample variance–covariance matrix of continuous variables in computing the likelihood. Even though simulation studies show that this works well, it is not the most appropriate way of handling such data (e.g., in SEM, the thresholds of the polychoric correlations are estimated at the same time as the SEM model). Second, this method-

ology assumes an underlying normally distributed variable, which might be problematic because zero usually means the absence of any symptoms (a strict boundary). Third, polychoric correlations can lead to strange results (see Chapter 2 for an overview) when pairwise marginal crosstabulations of items contain zeroes, which could be expected in highly skewed ordinal data.

2. Data can be dichotomized, and the Ising model can be computed. Although setting the cutoff between 0 and 1 seems appropriate and is defensible, doing so will lose information on the severity of items.

3. Mixed graphical models can be used, in which case the variables are treated as categorical. This method takes all responses into account but loses information pertaining to the order of responses (e.g., 3 is higher than 2, and 2 is higher than 1) and instead treats each response as a categorical outcome.

4. Ordinal data can be ignored and treated as continuous. This method is not recommended because simulation studies have shown that doing so has a lower sensitivity than when using polychoric correlations and also features an inflated Type 1 error rate when statistically comparing centrality indices.

Future researchers should focus on better estimation methods for graphical models on ordinal data. Such estimation methods will likely come from psychometrics because ordinal data has long been handled in many ways. Because the GGM can be included in the SEM framework (see Chapter 7), handling ordinal data in the same manner as in SEM (e.g., by using weighted least squares estimation; Muthén, 1984) seems a logical first step. However, extending such methodology to include high-dimensional model selection will be challenging.

Evidence for Sparsity

As strongly argued in Chapter 4, using the LASSO estimation leads to sparsity (edge weights of zero) in the corresponding network model. As such, observing zeroes is not evidence that the true network is sparse. The same is true when edges are thresholded for significance (as in Chapter 6) or when step-wise model search is used (as in Chapter 7). The goal of these methods is to maximize *specificity* (see Chapter 2). Closely related to null hypothesis testing: removing an edge is not evidence that the edge weight is zero (i.e., the null-hypothesis being true); an edge might also be removed because the data are too noisy. Classical tests, LASSO regularization, and frequentist model search cannot differentiate between noisy data and the null-hypothesis being true (Wagenmakers, 2007).

The question whether a missing edge is due to the null hypothesis being true, however, is a very important one. An edge weight of *zero* in a pairwise Markov random field, such as the GGM or the Ising model, indicates that two variables are *conditionally independent*. This is important for two reasons. First, as already outlined in this dissertation, conditional independence plays a crucial role in causality (Pearl, 2000). For example, the causal structure $A \rightarrow B \rightarrow C$ implies that A and C are conditionally independent given B. Second, when the latent common cause model is true, no two variables should be conditionally independent given any other variable in the dataset. Conceptually, this implies that the

only variable on which one could condition to make observed variables independent is the latent variable. Network models only show conditional associations after conditioning on observed variables. As such, when we find strong evidence that several pairs of variables become conditionally independent given a third, the common cause model will not be true.

In network psychometrics, we are interested in finding *conditional independence* in addition to finding strong conditional *dependencies*. However, the methods used today only allow for the latter. Future researchers should aim to develop methods in which both can be found. That is, for every edge, we should want to know the evidence for that edge existing (strong relationship) and the evidence for that edge not existing (conditional independence). This is difficult to accomplish in the frequentist framework, typically used in network psychometrics, but it is possible in a *Bayesian* framework. In recent years, Bayesian analysts have worked out the Bayes factor (Kass & Raftery, 1995; Ly, Verhagen, & Wagenmakers, 2016) as a default method for quantifying both the evidence for the null and for the alternative hypothesis. Such Bayes factors can possibly be computed for every edge in the graph, allowing a researcher to identify which edges are likely present, which edges are likely zero, and the edges whose data are too noisy to make such a distinction. The Bayes factors for partial correlations have been worked out (Wetzels & Wagenmakers, 2012). Node-wise estimation of graphical models could possibly also be used to obtain two Bayes factors per edge (Gelman, Jakulin, Pittau, & Su, 2008), using regular regression for the GGM and logistic regression for the Ising model. Finally, the work of Mulder (2014) on testing constraints on correlation matrices could possibly be extended to testing constraints on partial correlation matrices. An additional challenge will be to combine such methods with high-dimensional model selection, such as the LASSO, for which the Bayesian LASSO could possibly be used (Park & Casella, 2008).

Should Graph Theory Be Used to Analyze Probabilistic Graphical Models?

In this dissertation I have presented several methods for estimating network structures on psychological data. As nodes represent variables and edges are typically unknown, all of these models belong to a class known as *probabilistic graphical models* (Koller & Friedman, 2009; Lauritzen, 1996). These models aim to characterize the joint likelihood of observed variables, and allow for results to be represented through networks. Although graphically depicting these models as networks is a powerful technique for communicating such high-dimensional analysis results, it is questionable if measures from graph theory, such as centrality indices, could be readily applied to the networks estimated this way. Meaning, can probabilistic graphical models be interpreted in the same way as, say, a railroad network?

As described several times in this dissertation, typical methodology for analyzing weighted networks—such as computing centrality measures (Opsahl et al., 2010)—are often used on models obtained in network psychometrics. In this methodology, each edge is first transformed into a *length* (i.e., the inverse absolute value of an edge weights), then the resulting network is analyzed as, for example, a railroad network would. The distance between two nodes in a network is
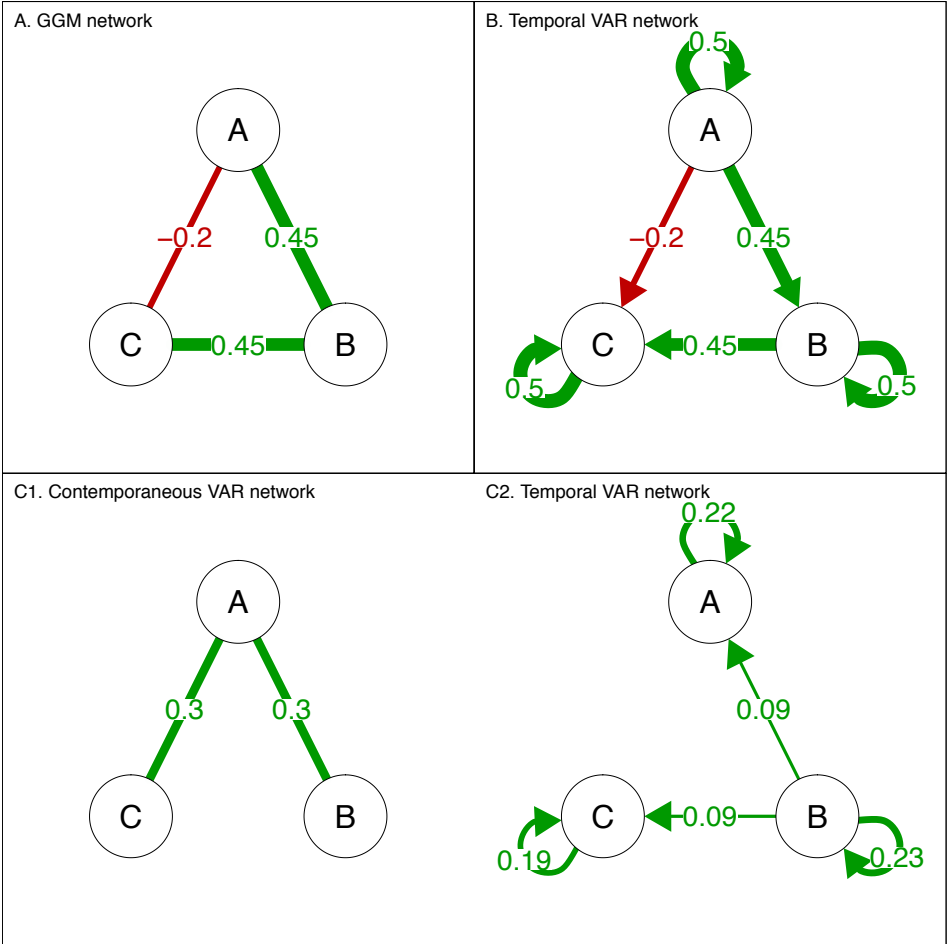
Figure 12.1: Three hypothetical graphical models for which computing network descriptives might be problematic. Panel A shows a Gaussian graphical model (a network of partial correlation coefficients), Panel B shows the temporal structure of a VAR analysis and Panel C shows both the contemporaneous and temporal structure of a VAR analysis.

defined through computing *shortest paths*, an important property in computing both closeness node centrality (i.e., the inverse sum of such distances) and the betweenness node centrality (i.e., measuring how often a node lies on the shortest path). Such methods are insightful when researchers can interpret these shortest paths to be sensible (e.g., passengers using a rail network will probably only travel through the shortest path), but it might make less sense in the context of probabilistic graphical models.

Figure 12.1 illustrates some examples of where interpreting such structures as networks can go wrong. In Panel A, a three-node GGM is depicted. When investigating the distance between nodes A and C, a typical network analysis would indicate that the *shortest path* is the path $A - B - C$, resulting in B having the highest *betweenness* and the direct path between A and C being ignored. However, such a result would not take all the information of the model into account. In fact, A and C are marginally independent; the correlation between A and C is exactly zero, indicating that knowing A contains no information on C and vise versa. Such a structure could emerge if B is a common effect of A and C, in which case disturbing A can, in no way, have any effect on C. As such, it is questionable what it means for B to have a high betweenness if no causal effect goes through B.

Panel B shows a vector auto-regression (VAR) temporal model in which the contemporaneous structure is ignored. This network would lead to a similar conclusion because of the network in Panel A: The shortest path from A to C goes via B. In this network, however, edges indicate Lag-1 effects. This means that the path $A \rightarrow B \rightarrow C$ indicates a Lag-2 effect, whereas the direct path $A \rightarrow B$ indicates a Lag-1 effect. Such paths are not even comparable because they indicate completely different temporal structures. Finally, Panel C shows both model matrices obtained from a VAR model. Suppose a researcher is interested in identifying which node is best able to predict all nodes at later measurement. In this case, only investigating the temporal structure would lead to the conclusion that the most important node is B. However, such an analysis would not take the contemporaneous network, in which Node A is highly central, into account.

Information theory. A potential solution for such problems is to *not* interpret probabilistic graphical models as networks, but rather, for what they are: full characterizations of the joint likelihood. In this line of thinking, the graphical representation is only useful for visualizing the statistical results but should not be over interpreted. The estimated model, nonetheless, is extremely powerful because it captures the associational structure of a dataset without the need for underlying theory on the causal mechanisms. A possible solution for inference methods then lies in the use of *information theory* (Cover & Thomas, 2012), which has shown to be a promising gateway to understanding the full complexity of such systems (Quax, Apolloni, & Sloot, 2013; Quax, Kandhai, & Sloot, 2013).

In information theory, we can make use of the *Shannon entropy* (Cover & Thomas, 2012) of a set of random variables, $\boldsymbol{Y}$, which denotes the average amount of bits of information needed to communicate a discrete outcome. When dealing with continuous variables, as we do in the GGM and VAR models, we can define

the *differential entropy*:

$$h\left(\boldsymbol{Y}\right) = -\mathbb{E}\left[\log_2 f\left(\boldsymbol{Y}\right)\right],$$

in which $f\left(\boldsymbol{y}\right)$ denotes the density function of $\boldsymbol{Y}$. This measure can be computed for any number of variables and quantifies their volatility—in the case of a single continuous variable, the entropy is directly related to the variance. Now, divide $\boldsymbol{Y}$ in two subsets $\boldsymbol{Y}^{(1)}$ and $\boldsymbol{Y}^{(2)}$. We can then quantify the association in the two subsets using the *mutual information*:

$$I\left(\boldsymbol{Y}^{(1)};\boldsymbol{Y}^{(2)}\right) = h\left(\boldsymbol{Y}^{(1)}\right) + h\left(\boldsymbol{Y}^{(2)}\right) - h\left(\boldsymbol{Y}^{(1)},\boldsymbol{Y}^{(2)}\right).$$

This measure can act as a general measure for strength of association between any set of variables to any other set of variables.

In the network perspective, we treat strongly associated variables as densely connected and usually take the analogy of such strongly connected variables to be *close* to one another. Mutual information can then be seen as a new form of quantifying *closeness* between nodes or sets of nodes—the inverse of distance. Therefore, mutual information is an alternative to the shortest path length. This measure not only takes the shortest paths into account but all other paths as well. For example, the mutual information of two variables or nodes (we often use these terms interchangeably), $I(Y_i; Y_j)$, can be taken as a measure of how close these two nodes are to each other. The mutual information between two sets, $I\left(\boldsymbol{Y}^{(1)};\boldsymbol{Y}^{(2)}\right)$—for example, in which Set (1) contains the symptoms of depression and Set (2) contains the symptoms of generalized anxiety—can be taken as a measure of closeness between two groups of nodes. Furthermore, the mutual information of one variable, (Y_1), with respect to all other variables, $(Y_{-(i)})$ and $I(Y_1; Y_{-(i)})$, can be used as a centrality measure. Finally, when temporal information is present, only computing the information one node has on all nodes at the next measurement, $I(Y_{t1}; Y_{t+1})$, can be taken as a temporal centrality measure, which takes into account both the contemporaneous and temporal network (see Chapter 5 and Chapter 6).

When $\boldsymbol{Y}$ has a multivariate normal distribution with size P and variance–covariance matrix $\boldsymbol{\Sigma}$, as is the case in both the GGM and graphical VAR model, the differential entropy becomes (Cover & Thomas, 2012, p. 250):

$$h\left(\boldsymbol{Y}\right) = \frac{1}{2}\log_2\left((2\pi e)^P \mid \boldsymbol{\Sigma}\mid\right).$$

As can be seen, this measure is a direct function of the size of the variance–covariance matrix $\boldsymbol{\Sigma}$. This expression allows us to compute all mutual informations described above. For example, the mutual information between two variables can be computed as:

$$I(Y_i; Y_j) = -\frac{1}{2}\log_2\left(1 - \sigma_{ij}^2\right).$$

As can be seen, this measure is a direct property of the explained variance σ_{ij}^2 between two variables. The mutual information of one variable with all other variables becomes:

$$I(Y_1; Y_{-(i)}) = \frac{1}{2}\log_2\left(\frac{|\boldsymbol{\Sigma}_{-(i)}|}{|\boldsymbol{\Sigma}|}\right),$$

in which $\boldsymbol{\Sigma}_{-(i)}$ denotes the variance–covariance matrix without row and column i. Finally, in a stationary time series of multivariate normal data (e.g., a Lag-1 VAR model), the temporal closeness above becomes:

$$I\left(y_{t,j}; \boldsymbol{y}_{t+1}\right) = \frac{1}{2}\log_2\left(\frac{\sigma_{jj}\mid\boldsymbol{\Sigma}\mid}{\mid\boldsymbol{\Sigma}_{j,+}^{\mathrm{TP}}\mid}\right),$$

in which $\boldsymbol{\Sigma}_{j,+}^{\mathrm{TP}}$ denotes a subsetted Toeplitz matrix:

$$\boldsymbol{\Sigma}_{j,+}^{\mathrm{TP}} = \mathrm{Var}\left(Y_{ti}, \boldsymbol{Y}_{t+1}\right),$$

which can be obtained from a VAR analysis.

Applying these metrics to the networks shown in Figure 12.1 leads to strikingly different interpretations. In Panel A, Nodes A and C are now shown to be independent; thus, Node B does not have a problematic interpretation of having a high betweenness. Now in Panel C,[3] Node A is shown to have slightly more information over the next time point than Node B, even though Node B has more temporal connections. Information theory is a promising gateway to analyzing the network models obtained. However, future researchers must thoroughly test and validated these metrics on psychological data.

The Importance of Intercepts

The network models outlined in this dissertation are all models of second-order moments. That is, they model variances and covariances but not expected values. The parameters that do model the expected value in the GGM, Ising model, and VAR model are the *intercepts*. When drawing a network, these are ignored. As such, when using the network structure to compute centrality, for example, intercepts are not taken into account. The problem with this approach is that links are formed between variables that may have largely different intercepts. Particularly in models of binary variables, links may be formed between variables that never have high entropy at the same time. For example, whenever a person is in danger of suffering from suicidal ideation (high entropy), we might expect that person to always experience sadness (low entropy). If we apply a virus-spreading analogy (Borsboom et al., 2011), such nodes would never "infect" each other; the link would never be used.

Figure 12.2, Panel A, shows an example of a network we might estimate on educational data. This network is a fully connected Ising model, also called a Curie-Weiss model, which is known to be equivalent to the IRT model shown in Panel B (Marsman et al., 2015; see also Chapter 8). As such, IRT models are often used and work well on educational data, the Ising model of Panel A is not unreasonable. We can see a link between the items "1 + 1" and "0.07692 + 0.3409." This link represents a very plausible predictive relationship. Knowing someone can answer "0.07692 + 0.3409" tells us that person can also answer "1 +

---

[3]Panel B only shows half the information needed to characterize the full likelihood; see Chapter 6.
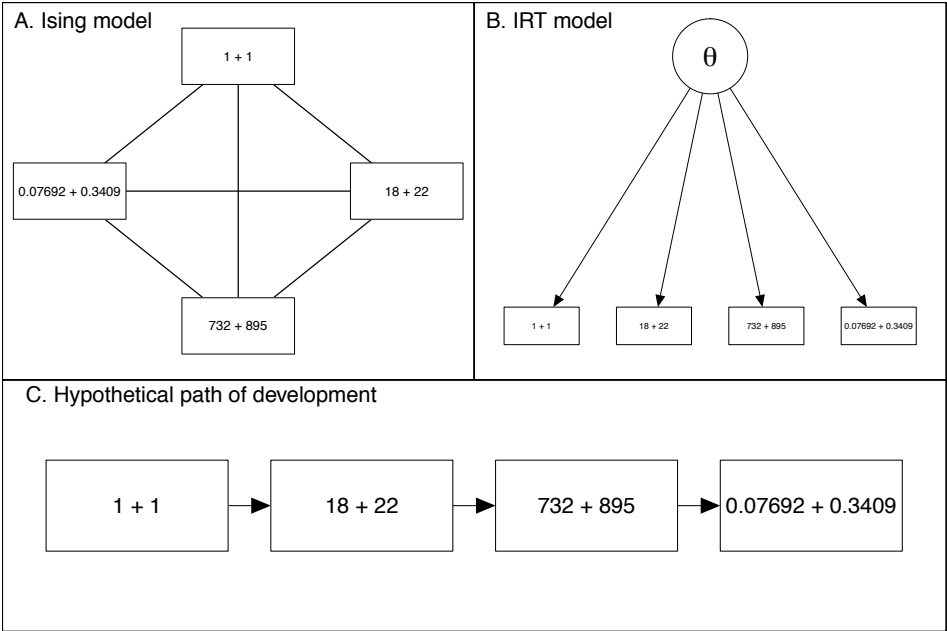
Figure 12.2: Three potential network structures on educational data. Panels A and B are equivalent models that do not show the difficulty of an item. In Panel C, items are ordered according to their difficulty, pointing from the easier item to the more difficult item.

1" (because the person is able to answer the much harder question as well), and knowing someone cannot answer "1 + 1" also tells us that person cannot answer "0.07692 + 0.3409" (because that person cannot answer the simpler question). These items likely do not have high entropy at the same time in any person's life. This means that when a person is struggling with "1 + 1," and may or may not answer this item correctly (high entropy), it is highly likely that this person will never be able to answer "0.07692 + 0.3409" correctly (always an incorrect response; low entropy). Conversely, whenever a person correctly answers the item "0.07692 + 0.3409," that person will likely always answer the item "1 + 1" correctly.

The network perspective would lead to the following interpretation of Panel A, that one could influence the probability of correctly answering "0.07692 + 0.3409" by training someone on the question "1 + 1." However, this seems unlikely. Teaching a person the techniques needed to answer "1 + 1" would not prepare that person to answer "0.07692 + 0.3409," which requires knowledge of decimal points, counting over tens, and so forth. The latent variable model in Panel B would implicate that training someone on one of the items would never help that person answer other items correctly. Although I do not wish to argue against mathematical ability, I do think that such an assumption might also be too strict. Children learn by making items, and learning how to make one item helps a child

make another item. This learning, however, does not jump wildly as would be expected from Panel A, rather it follows a straight path. Learning "1 + 1" helps to answer the item "2 + 2," which helps to answer the item "3 + 3", and so forth.

Panel C shows a network that is based only on intercepts rather than covariances. Here, each item points to the first harder item. This very different network structure shows that people first learn "1 + 1," then "18 + 22," and so forth. I term such a network structure a *path of development*. The network shown in Panel C is merely a hypothetical example of what a network that also takes intercepts into account could look like. In perfect unidimensional cases, it might look like Panel C, whereas in multidimensional cases, one could envision, for example, parallel paths or the path splitting. I do not seek to propose a new modeling framework in this section but merely wish to highlight that taking intercepts into account could lead to different ways of investigating the phenomena of interest.

Complexity

The network models, as outlined in this dissertation, are but one of the many consequences that may come from a more general *hypothesis of complexity*. Psychological behavior plausibly is the result of emergent behavior in a complex system of interacting psychological, biological, and sociological components. Simply stated, psychology is complex. People's behavior is dictated by their brains, which consist of billions of neurons, formed by many years of development. As such, every person is a mini universe of complexity. These universes, in turn, interact with one another in complicated social networks. Perhaps, psychology is one of the hardest fields to tackle. It is, in my opinion, only logical that many behaviors have no simple explanation.

The network model is but one attempt at grasping this complexity; we should not get sidetracked by believing it is the only possible attempt. The hypothesis of complexity is not limited to the expectation that data are generated due to an underlying (sparse) network model of, at most, second-order moments. This hypothesis reaches further, with many more implications. This point of view can take psychological research in many different directions—rather than merely the estimation of network models. For example, long-term predictions can be made on the effects of interventions, without understanding the true underlying causal mechanisms. Also of particular importance is the work done by van de Leemput et al. (2014) and Wichers et al. (2016) on identifying early warning signals for phase transitions in psychology, such as the onset of depression. I think that the hypothesis of complexity has much to offer in the years to come and will change psychological research in ways we cannot imagine now.

12.3   Conclusion

In this discussion, I outlined various topics for future research which can be tackled in network psychometrics: improving centrality measures, handling missing and ordinal data, quantifying evidence for sparsity in the network, and incorporating intercepts in inference on these models. This is just a highlight of several future directions; many more can be conceived, such as tackling heterogeneity, improving

multilevel estimation of contemporaneous effects, handling a mixture of observational and experimental data, and extending networks to nonlinear dynamics. Finally, I noted a far more general field of research—complexity in psychology—of which network modeling is merely a small part. The network models proposed in this dissertation add much to the toolbox of psychological and psychometric researchers. Network psychometrics, however, is still a young field of research with many unanswered questions. The full utility of these methods and their place in psychological research and psychometrics will be determined on the road ahead.

Part V

Appendices

References

Aan het Rot, M., Hogenelst, K., & Schoevers, R. A. (2012). Mood disorders in everyday life: A systematic review of experience sampling and ecological momentary assessment studies. *Clinical Psychology Review*, *32*(6), 510–523. doi: 10.1016/j.cpr.2012.05.007 72

Abegaz, F., & Wit, E. (2013). Sparse time series chain graphical models for reconstructing genetic networks. *Biostatistics*, kxt005. 31, 73, 79, 98, 107, 112

Abegaz, F., & Wit, E. (2015). SparseTSCGM: Sparse time series chain graphical models [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=SparseTSCGM` (R package version 2.2) 31, 98

Aggen, S. H., Neale, M. C., & Kendler, K. S. (2005). DSM criteria for major depression: evaluating symptom patterns using latent-trait item response models. *Psychological Medicine*, *35*(04), 475–487. 143, 144

Agresti, A. (1990). *Categorical data analysis*. New York, NY: John Wiley & Sons Inc. 65, 153, 158, 159

American Psychiatric Association. (2000). *Diagnostic and statistical manual of mental disorders: DSM-IV-TR* (4th ed., text rev. ed.). Washington, DC, USA: Author. 44

American Psychiatric Association. (2013). *Diagnostic and statistical manual of mental disorders: DSM-V* (5th ed. ed.). Arlington, VA, USA: Author. 61

Anderson, C. J., Li, Z., & Vermunt, J. (2007). Estimation of models in the rasch family for polytomous items and multiple latent variables. *Journal of Statistical Software*, *20*(6), 1–36. 157

Anderson, C. J., & Vermunt, J. K. (2000). Log-multiplicative association models as latent variable models for nominal and/or ordinal data. *Sociological Methodology*, *30*(1), 81–121. 157

Anderson, C. J., & Yu, H.-T. (2007). Log-multiplicative association models as item response models. *Psychometrika*, *72*(1), 5–23. 157

Arbuckle, J. L. (2010). *Amos 19 user's guide*. Chicago, IL: Amos Development Corporation. 222, 234

Arbuckle, J. L., Marcoulides, G. A., & Schumacker, R. E. (1996). Full information estimation in the presence of incomplete data. *Advanced structural equation modeling: Issues and techniques*, 243-277. 141

Armour, C., Fried, E. I., Deserno, M. K., Tsai, J., Southwick, S. M., & Pietrzak, R. H. (2016). A Network Analysis of DSM-5 posttraumatic stress disorder symptoms? and clinically relevant correlates in a national sample of U.S. military veterans. *PsyArXiv preprint*. Retrieved from `https://osf.io/p69m7/` 18, 20

Ashton, M. C., & Lee, K. (2005). A defence of the lexical approach to the study of personality structure. *European journal of personality*, *19*(1), 5–24. 197, 219

Ashton, M. C., & Lee, K. (2007). Empirical, theoretical, and practical advantages of the hexaco model of personality structure. *Personality and Social Psychology Review*, *11*(2), 150–166. 199, 219

Ashton, M. C., & Lee, K. (2009). The hexaco–60: A short measure of the major dimensions of personality. *Journal of personality assessment*, *91*(4), 340–345. 199, 200

Bache, S. M., & Wickham, H. (2014). magrittr: A forward-pipe operator for R [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=magrittr` (R package version 1.5) 44

Barabási, A.-L. (2009). Scale-free networks: a decade and beyond. *science*, *325*(5939), 412–413. 220

Barabási, A.-L., & Albert, R. (1999). Emergence of scaling in random networks. *science*, *286*(5439), 509–512. 162

Barber, R. F., Drton, M., & Others. (2015). High-dimensional Ising model selection with Bayesian information criteria. *Electronic Journal of Statistics*, *9*(1), 567–607. 106

Barrat, A., Barthelemy, M., Pastor-Satorras, R., & Vespignani, A. (2004). The architecture of complex weighted networks. *Proceedings of the National Academy of Sciences of the United States of America*, *101*(11), 3747–3752. 206, 209

Barzel, B., & Biham, O. (2009). Quantifying the connectivity of a network: The network correlation function method. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *80*(4), 046104. doi: 10.1103/PhysRevE.80.046104 34

Bates, D., Mächler, M., Bolker, B., & Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, *67*(1), 1–48. doi: 10.18637/jss.v067.i01 104

Benet-Martinez, V., & John, O. (1998). Los Cinco Grandes across Cultures and Ethnic Groups: Multitrait Multimethod Analyses of the Big Five in Spanish and English. *Journal of Personality and Social Psychology*, *75*, 729–750. 25, 138, 179

Bentler, P. M. (1990). Comparative fit indexes in structural models. *Psychological bulletin*, *107*(2), 238–346. 122

Bentler, P. M., & Wu, E. J. (1993). *EQS6 structural equations program manual.* Multivariate Software. 234

Besag, J. (1975). Statistical analysis of non-lattice data. *The statistician*, *24*, 179–195. 159

Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. In F. Lord & M. Novick (Eds.), *Statistical theories of mental test scores*. Reading, MA: Addison-Wesley. 154

Bland, J. M., & Altman, D. G. (1995). Multiple significance tests: the bonferroni method. *Bmj*, *310*(6973), 170. 42

Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., & Hwang, D. (2006). Complex networks: Structure and dynamics. *Physics reports*, *424*(4-5), 175–308. 178, 205, 220

Boker, S., Neale, M., Maes, H., Wilde, M., Spiegel, M., Brick, T., . . . others (2011). OpenMx: an open source extended structural equation modeling framework. *Psychometrika*, *76*(2), 306–317. 190, 222

Boker, S. M., McArdle, J., & Neale, M. (2002). An algorithm for the hierarchical organization of path diagrams and calculation of components of expected covariance. *Structural Equation Modeling*, *9*(2), 174–194. 221, 225, 231, 232

Bollen, K. (1989). *Structural equations with latent variables*. New York, NY, USA: Wiley. 190, 225

Bollen, K., & Lennox, R. (1991). Conventional Wisdom on Measurement: A Structural Equation Perspective. *Psychological Bulletin*, *110*(2), 305–314. 164, 166, 178, 188

Bollen, K. A., & Stine, R. A. (1992). Bootstrapping goodness-of-fit measures in structural equation models. *Sociological Methods & Research*, *21*(2), 205–229. 39

Bonacich, P. (1972). Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*, *2*(1), 113–120. 208

Bonacich, P. (1987). Power and centrality: A family of measures. *American journal of sociology*, 1170–1182. 208

Bonacich, P. (2007). Some unique properties of eigenvector centrality. *Social networks*, *29*(4), 555–564. 208

Bonacich, P., & Lloyd, P. (2001). Eigenvector-like measures of centrality for asymmetric relations. *Social networks*, *23*(3), 191–201. 208

Borgatti, S. P. (2005). Centrality and network flow. *Social networks*, *27*(1), 55–71. 206, 207

Borgatti, S. P., & Everett, M. G. (2006). A graph-theoretic perspective on centrality. *Social networks*, *28*(4), 466–484. 206

Borsboom, D. (2005). *Measuring the mind: Conceptual issues in contemporary psychometrics*. Cambridge, UK: Cambridge University Press. 164, 219

Borsboom, D. (2008). Psychometric Perspectives on Diagnostic Systems. *Journal of clinical psychology*, *64*(9), 1089–1108. 1, 116, 119, 120, 178

Borsboom, D. (in press). A network theory of mental disorders. *World Psychiatry*. 7, 72

Borsboom, D., & Cramer, A. O. J. (2013). Network analysis: an integrative approach to the structure of psychopathology. *Annual review of clinical psychology*, *9*, 91–121. 7, 11, 13, 34, 60, 72, 75, 86, 119, 143, 165, 169, 196, 209, 237

Borsboom, D., Cramer, A. O. J., Schmittmann, V. D., Epskamp, S., & Waldorp, L. J. (2011). The small world of psychopathology. *PloS one*, *6*(11), e27407. 1, 59, 60, 65, 75, 86, 119, 120, 152, 166, 217, 245

Borsboom, D., Mellenbergh, G. J., & Van Heerden, J. (2003). The theoretical status of latent variables. *Psychological review*, *110*(2), 203–219. 113, 164, 196, 197, 219

Boschloo, L., Van Borkulo, C. D., Rhemtulla, M., Keyes, K. M., Borsboom, D., & Schoevers, R. A. (2015). The network structure of symptoms of the diagnostic and statistical manual of mental disorders. *PLoS ONE*, *10*(9), e0137621. 34, 59, 61, 64, 67, 70, 75

Brandes, U. (2001). A faster algorithm for betweenness centrality. *Journal of mathematical sociology*, *25*(2), 163–177. 207

Brandes, U. (2008). On variants of shortest-path betweenness centrality and their generic computation. *Social Networks*, *30*(2), 136–145. 207, 208

Brandes, U., & Fleischer, D. (2005). Centrality measures based on current flow. In *Annual symposium on theoretical aspects of computer science* (pp. 533–544). 208

Bringmann, L. F., Lemmens, L. H., Huibers, M. J., Borsboom, D., & Tuerlinckx, F. (2015). Revealing the dynamic network structure of the beck depression inventory-ii. *Psychological medicine*, *45*(04), 747–757. 56, 72, 86

Bringmann, L. F., Vissers, N., Wichers, M., Geschwind, N., Kuppens, P., Peeters, F., . . . Tuerlinckx, F. (2013). A network approach to psychopathology: new insights into clinical longitudinal data. *PloS one*, *8*(4), e60188. 72, 73, 80, 86, 99, 100, 104, 208, 219, 220

Browne, M. W., & Cudeck, R. (1992). Alternative ways of assessing model fit. *Sociological Methods & Research*, *21*(2), 230–258. 122

Bühlmann, P. (2013). Statistical significance in high-dimensional linear models. *Bernoulli*, *19*(4), 1212–1242. 161

Bühlmann, P., & van de Geer, S. (2011). *Statistics for high-dimensional data: Methods, theory and applications*. New York, NY, USA: Springer. 161, 162

Bulteel, K., Tuerlinckx, F., Brose, A., & Ceulemans, E. (2016). Using raw var regression coefficients to build networks can be misleading. *Multivariate Behavioral Research*, *51*(2-3), 330-344. 79

Butts, C. T. (2008). Social network analysis: A methodological introduction. *Asian Journal of Social Psychology*, *11*(1), 13–41. 205

Butts, C. T. (2010). sna: Tools for social network analysis [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=sna` (R package version 2.2-0) 182, 206

Butts, C. T., Handcock, M. S., & Hunter, D. R. (2011). network: Classes for relational data [Computer software manual]. Retrieved from `http://statnet.org/` (R package version 1.7) 178

Butts, C. T., et al. (2008). Social network analysis with sna. *Journal of Statistical Software*, *24*(6), 1–51. 208

Campbell, D. T., & Fiske, D. W. (1959). Convergent and discriminant validation by the multitrait-multimethod matrix. *Psychological bulletin*, *56*(2), 81–105. 201

Cattell, R. B. (1988). The data box. In *Handbook of multivariate experimental psychology* (pp. 69–130). New York, NY, USA: Springer. 88

Cervone, D. (2005). Personality architecture: Within-person structures and processes. *Annual Review of Psychology*, *56*, 423–452. 219

Chalmers, R. P. (2012). mirt: A multidimensional item response theory package for the R environment. *Journal of Statistical Software*, *48*(6), 1–29. 162

Chandrasekaran, V., Parrilo, P. A., & Willsky, A. S. (2012). Latent variable graphical model selection via convex optimization (with discussion). *The Annals of Statistics*, *40*(4), 1935–1967. 122, 127, 170, 220

Chen, G., Glen, D. R., Saad, Z. S., Hamilton, J. P., Thomason, M. E., Gotlib, I. H., & Cox, R. W. (2011). Vector autoregression, structural equation modeling, and their synthesis in neuroimaging data analysis. *Computers in biology and medicine*, *41*(12), 1142–1155. 112

Chen, J., & Chen, Z. (2008). Extended bayesian information criteria for model selection with large model spaces. *Biometrika*, *95*(3), 759–771. 14, 38, 61, 92, 129, 160

Chernick, M. R. (2011). *Bootstrap methods: A guide for practitioners and researchers* (Vol. 619). New York, NY, USA: John Wiley & Sons. 41

Chin, W. W. (2001). PLS-Graph user's guide [Computer software manual]. CT Bauer College of Business, University of Houston. 222

Cohen, J. (1977). *Statistical power analysis for the behavioral sciences*. New York, NY, USA: Academic Press. 41, 56

Cohen, J. (1994). The earth is round (p<0.05). *American Psychologist*, *49*(12), 997–1003. Retrieved from `http://doi.apa.org/getdoi.cfm?doi=10.1037/0003-066X.49.12.997` 42

Costa, P. T., & McCrae, R. R. (1992). *Professional manual: revised neo personality inventory (NEO-PI-R) and NEO five-factor inventory (NEO-FFI)*. Odessa, FL, USA: Psychological Assessment Resources. 179, 201

Costantini, G., Epskamp, S., Borsboom, D., Perugini, M., Mõttus, R., Waldorp, L. J., & Cramer, A. O. J. (2015). State of the aRt personality research: A tutorial on network analysis of personality data in R. *Journal of Research in Personality*, *54*, 13–29. 13, 24, 34, 37, 38, 77, 116, 169

Costantini, G., & Perugini, M. (2014). Generalization of clustering coefficients to signed correlation networks. *PloS one*, *9*(2), e88669. 205, 209, 213, 220

Costantini, G., Richetin, J., Borsboom, D., Fried, E. I., Rhemtulla, M., & Perugini, M. (2015). Development of indirect measures of conscientiousness: combining a facets approach and network analysis. *European Journal of Personality*, *29*(5), 548–567. 34

Costenbader, E., & Valente, T. W. (2003). The stability of centrality measures when networks are sampled. *Social networks*, *25*(4), 283–307. 41, 43, 57

Cover, T. M., & Thomas, J. A. (2012). *Elements of information theory*. Hoboken, NJ, USA: John Wiley & Sons. 243, 244

Cox, D. R. (1972). The analysis of multivariate binary data. *Applied statistics*, *21*, 113–120. 154

Cox, D. R., & Wermuth, N. (1993). Linear dependencies represented by chain graphs. *Statistical science*, 204–218. 60, 202

A. References

Cox, D. R., & Wermuth, N. (1994). A note on the quadratic exponential binary distribution. *Biometrika*, *81*(2), 403–408. 13, 154

Cramer, A. O. J., & Borsboom, D. (2015). Problems Attract Problems: A Network Perspective on Mental Disorders. *Emerging Trends in the Social and Behavioral Sciences: An Interdisciplinary, Searchable, and Linkable Resource*, 1–15. 72

Cramer, A. O. J., Borsboom, D., Aggen, S. H., & Kendler, K. S. (2012). The pathoplasticity of dysphoric episodes: differential impact of stressful life events on the pattern of depressive symptom inter-correlations. *Psychological medicine*, *42*(05), 957–965. 60

Cramer, A. O. J., Sluis, S., Noordhof, A., Wichers, M., Geschwind, N., Aggen, S. H., . . . Borsboom, D. (2012). Dimensions of normal personality as networks in search of equilibrium: You can't like parties if you don't like people. *European Journal of Personality*, *26*(4), 414–431. 7, 34, 86, 116, 119, 166, 169, 196, 197, 201, 237

Cramer, A. O. J., Waldorp, L., van der Maas, H., & Borsboom, D. (2010). Comorbidity: A Network Perspective. *Behavioral and Brain Sciences*, *33*(2-3), 137–150. 1, 7, 34, 60, 65, 72, 75, 86, 116, 119, 120, 143, 164, 166, 169, 178, 217, 218, 220, 237

Cressie, N., & Holland, P. W. (1983). Characterizing the manifest probabilities of latent trait models. *Psychometrika*, *48*(1), 129–141. 157

Cronbach, L. J., & Meehl, P. E. (1955). Construct validity in psychological tests. *Psychological bulletin*, *52*(4), 281–302. 102

Crucitti, P., Latora, V., Marchiori, M., & Rapisarda, A. (2004). Error and attack tolerance of complex networks. *Physica A: Statistical Mechanics and its Applications*, *340*(1), 388–394. 206

Csardi, G., & Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal*, *Complex Systems*, 1695. 162, 178, 182, 184, 206, 222, 232

Dalege, J., Borsboom, D., van Harreveld, F., van den Berg, H., Conner, M., & van der Maas, H. L. J. (2016). Toward a formalized account of attitudes: The Causal Attitude Network (CAN) model. *Psychological review*, *123*(1), 2–22. 6, 34, 61, 116

da Silva, A. R., Cecon, P. R., & Puiatti, M. (2015). Phenotypic correlation network analysis of garlic variables. *Multi-Science Journal*, *1*(3), 9–12. 6

de Haan-Rietdijk, S., Kuppens, P., & Hamaker, E. L. (2016). What's in a Day? A Guide to Decomposing the Variance in Intensive Longitudinal Data. *Frontiers in Psychology*, *7*, 891. Retrieved from `http://journal.frontiersin.org/article/10.3389/fpsyg.2016.00891` doi: 10.3389/fpsyg.2016.00891 81

De Nooy, W., Mrvar, A., & Batagelj, V. (2011). *Exploratory social network analysis with pajek* (Vol. 27). Cambridge, UK: Cambridge University Press. 195, 205

De Raad, B., Barelds, D. P., Timmerman, M. E., De Roover, K., Mlačić, B., & Church, A. T. (2014). Towards a pan-cultural personality structure: Input from 11 psycholexical studies. *European Journal of Personality*, *28*(5), 497–510. 219

Deserno, M. K., Borsboom, D., Begeer, S., & Geurts, H. M. (2016). Multicausal systems ask for multicausal approaches: A network perspective on subjective well-being in individuals with autism spectrum disorder. *Autism*. doi: 10 .1177/1362361316660309 12

Di Battista, G., Eades, P., Tamassia, R., & Tollis, I. (1994). Algorithms for Drawing Graphs: An Annotated Bibliography. *Computational Geometry-Theory and Application*, *4*(5), 235–282. 182

Digman, J. (1989). Five Robust Trait Dimensions: Development, Stability, and Utility. *Journal of Personality*, *57*(2), 195–214. 25, 138, 179

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische mathematik*, *1*(1), 269–271. 207

Dolan, C., Oort, F., Stoel, R., & Wicherts, J. (2009). Testing Measurement Invariance in the Target Rotated Multigroup Exploratory Factor Model. *Structural Equation Modeling: A Multidisciplinary Journal*, *16*(2), 295–314. 179, 201

Drton, M., & Perlman, M. D. (2004). Model selection for gaussian concentration graphs. *Biometrika*, *91*(3), 591–602. 13, 204

Dryden, I. L., Scarr, M. R., & Taylor, C. C. (2003). Bayesian texture segmentation of weed and crop images using reversible jump markov chain monte carlo methods. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *52*(1), 31–50. 158

Dziak, J. J., Coffman, D. L., Lanza, S. T., & Li, R. (2012). Sensitivity and specificity of information criteria. *The Methodology Center and Department of Statistics, Penn State, The Pennsylvania State University*. 16, 30, 140

Edwards, J., & Bagozzi, R. (2000). On the Nature and Direction of Relationships Between Constructs and Measures. *Psychological Methods*, *5*(2), 155–174. 164, 166, 178, 188

Efron, B. (1979). Bootstrap methods: another look at the jackknife. *The Annals of Statistics*, *7*(1), 1–26. 39

Eichler, M. (2007). Granger causality and path diagrams for multivariate time series. *Journal of Econometrics*, *137*(2), 334–353. 96

Ellis, J. L., & Junker, B. W. (1997). Tail-measurability in monotone latent variable models. *Psychometrika*, *62*(4), 495–523. 122, 165, 166, 169

Enders, C. K. (2001). A primer on maximum likelihood algorithms available for use with missing data. *Structural Equation Modeling*, *8*(1), 128–141. 239

Epskamp, S. (2013). lisrelToR: Import output from LISREL into R [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package= lisrelToR` (R package version 0.1.4) 222

Epskamp, S. (2014). IsingSampler: Sampling methods and distribution functions for the Ising model [Computer software manual]. Retrieved from `github .com/SachaEpskamp/IsingSampler` (R package version 0.1.1) 40, 65, 159

Epskamp, S. (2015). graphicalVAR: Graphical var for experience sampling data [Computer software manual]. Retrieved from `http://CRAN.R-project .org/package=graphicalVAR` (R package version 0.1.3) 31, 79, 98

Epskamp, S. (2016). elasticIsing: Ising network estimation using elastic net and k-fold cross-validation [Computer software manual]. Retrieved from `github .com/SachaEpskamp/elasticIsing` (R package version 0.2) 68, 161, 162

Epskamp, S., Borsboom, D., & Fried, E. I. (2016). Estimating psychological networks and their accuracy: a tutorial paper. *arXiv preprint*, arXiv:1604.08462. 2, 11, 238

Epskamp, S., Cramer, A., Waldorp, L., Schmittmann, V. D., & Borsboom, D. (2012). qgraph: Network visualizations of relationships in psychometric data. *Journal of Statistical Software*, *48*(1), 1–18. 12, 18, 61, 75, 93, 164, 196, 199, 201, 206, 218, 237

Epskamp, S., Deserno, M. K., & Bringmann, L. F. (2016). mlvar: Multi-level vector autoregression [Computer software manual]. (R package version 0.3.0) 106

Epskamp, S., Maris, G., Waldorp, L., & Borsboom, D. (in press). Network psychometrics. In P. Irwing, D. Hughes, & T. Booth (Eds.), *Handbook of psychometrics.* New York, NY, USA: Wiley. Retrieved from `http://arxiv.org/abs/1609.02818` 60, 70, 86, 116, 120

Epskamp, S., Rhemtulla, M., & Borsboom, D. (2016). Generalized network psychometrics: Combining network and latent variable models. *arXiv preprint*, arXiv:1112.5635. 238

Fan, J., Feng, Y., & Wu, Y. (2009). Network exploration via the adaptive lasso and scad penalties. *The annals of applied statistics*, *3*(2), 521–541. 204

Fitzmaurice, G. M., Laird, N. M., & Rotnitzky, A. G. (1993). Regression models for discrete longitudinal responses. *Statistical Science*, *8*, 284–299. 154

Fleeson, W. (2001). Toward a structure-and process-integrated view of personality: Traits as density distributions of states. *Journal of personality and social psychology*, *80*(6), 1011–1027. 96, 219

Foa, E. B., Riggs, D. S., Dancu, C. V., & Rothbaum, B. O. (1993). Reliability and validity of a brief instrument for assessing post-traumatic stress disorder. *Journal of traumatic stress*, *6*(4), 459–473. 45

Forbush, K., Siew, C., & Vitevitch, M. (2016). Application of network analysis to identify interactive systems of eating disorder psychopathology. *Psychological Medicine*, *46*(12), 2667–2677. 34

Fox, J. (2006). Structural Equation Modeling With the sem Package in R. *Structural Equation Modeling*, *13*(3), 465–486. 177, 178, 179, 190, 221, 222

Foygel, R., & Drton, M. (2010). Extended Bayesian information criteria for Gaussian graphical models. *Advances in Neural Information Processing Systems*, *23*, 2020–2028. 12, 14, 15, 16, 18, 24, 25, 38, 39, 45, 50, 61, 92, 107, 129, 160, 170, 238

Foygel, R., & Drton, M. (2011). Bayesian model choice and information criteria in sparse generalized linear models. *arXiv preprint*, arXiv:1112.5635. 204

Foygel Barber, R., & Drton, M. (2015). High-dimensional Ising model selection with bayesian information criteria. *Electronic Journal of Statistics*, *9*(1), 567–607. 14, 38, 61, 160

Franić, S., Borsboom, D., Dolan, C. V., & Boomsma, D. I. (2014). The big five personality traits: psychological entities or statistical constructs? *Behavior genetics*, *44*(6), 591–604. 201

Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social networks*, *1*(3), 215–239. 206, 207

Freeman, L. C., Borgatti, S. P., & White, D. R. (1991). Centrality in valued graphs: A measure of betweenness based on network flow. *Social networks*, *13*(2), 141–154. 208

Fried, E. I., Bockting, C., Arjadi, R., Borsboom, D., Amshoff, M., Cramer, O. J., ... Stroebe, M. (2015). From loss to loneliness: The relationship between bereavement and depressive symptoms. *Journal of abnormal psychology*, *124*(2), 256–265. 2, 34, 60, 76, 116

Fried, E. I., Epskamp, S., Nesse, R. M., Tuerlinckx, F., & Borsboom, D. (2016). What are 'good' depression symptoms? Comparing the centrality of DSM and non-DSM symptoms of depression in a network analysis. *Journal of Affective Disorders*, *189*, 314–320. 12, 35, 36, 57, 72, 86

Fried, E. I., Nesse, R. M., Zivin, K., Guille, C., & Sen, S. (2014). Depression is more than the sum score of its parts: individual DSM symptoms have different risk factors. *Psychological medicine*, *44*(10), 2067–2076. 167

Fried, E. I., & van Borkulo, C. (2016). Mental disorders as networks of problems: a review of recent insights. *PsyArXiv Preprint*. Retrieved from `osf.io/6n8cg` 116

Fried, E. I., van Borkulo, C. D., Epskamp, S., Schoevers, R. A., Tuerlinckx, F., & Borsboom, D. (2016). Measuring depression over time... or not? lack of unidimensionality and longitudinal measurement invariance in four common rating scales of depression. *Psychological Assessment*, Advance Online Publication. Retrieved from `http://psycnet.apa.org/psycinfo/2016-04481-001/` 239

Friedman, J. H., Hastie, T., & Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, *9*(3), 432–441. 12, 14, 39, 92, 98, 159, 170, 220

Friedman, J. H., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, *33*(1), 1–22. 159, 161

Friedman, J. H., Hastie, T., & Tibshirani, R. (2014). glasso: Graphical lasso-estimation of gaussian graphical models [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=glasso` (R package version 1.8) 14, 18, 39, 45, 93, 98

Frijda, N. H. (1988). The laws of emotion. *American psychologist*, *43*(5), 349–358. 76

Fruchterman, T., & Reingold, E. (1991). Graph drawing by force-directed placement. *Software: Practice and Experience*, *21*(11), 1129–1164. 6, 182, 234

Funder, D. C. (1991). Global traits: A neo-allportian approach to personality. *Psychological Science*, *2*(1), 31–39. 197

Gates, K. M., & Molenaar, P. C. (2012). Group search algorithm recovers effective connectivity maps for individuals in homogeneous and heterogeneous samples. *NeuroImage*, *63*(1), 310–319. 112, 124

Gates, K. M., Molenaar, P. C., Hillary, F. G., Ram, N., & Rovine, M. J. (2010). Automatic search for fMRI connectivity mapping: an alternative to granger causality testing using formal equivalences among SEM path modeling, VAR, and unified SEM. *Neuroimage*, *50*(3), 1118–1125. 112

Gelman, A., & Hill, J. (2006). *Data analysis using regression and multilevel/hierarchical models.* New York, NY, USA: Cambridge University Press. 103

Gelman, A., Jakulin, A., Pittau, M. G., & Su, Y.-S. (2008). A weakly informative default prior distribution for logistic and other regression models. *The Annals of Applied Statistics*, 1360–1383. 241

Gentry, J., Long, L., Gentleman, R., Falcon, S., Hahne, F., Sarkar, D., & Hansen, K. (2011). Rgraphviz: Provides plotting capabilities for r graph objects [Computer software manual]. (R package version 1.27.0) 179

Genz, A., Bretz, F., Miwa, T., Mi, X., Leisch, F., Scheipl, F., & Hothorn, T. (2008). mvtnorm: Multivariate normal and t distributions [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=mvtnorm` (R package version 0.9-2) 40

Goldberg, L. (1993). The Structure of Phenotypic Personality Traits. *American Psychologist*, *48*(1), 26–34. 25, 138, 179

Goldberg, L. R. (1990a). An alternative "description of personality": the big-five factor structure. *Journal of personality and social psychology*, *59*(6), 1216–1229. 25, 138, 179

Goldberg, L. R. (1990b). An alternative "description of personality": the big-five factor structure. *Journal of personality and social psychology*, *59*(6), 1216–1229. 219

Goold, C., Vas, J. B., Olsen, C., & Newberry, R. C. (2015). A network approach to understanding dog personality. *Journal of Veterinary Behavior: Clinical Applications and Research*, *10*(5), 446. 6

Granger, C. W. J. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: Journal of the Econometric Society*, 424–438. 96

Green, P. J., & Richardson, S. (2002). Hidden markov models and disease mapping. *Journal of the American statistical association*, *97*(460), 1055–1070. 158

Gsell, A. S., Özkundakci, D., Hébert, M.-P., & Adrian, R. (2016). Quantifying change in pelagic plankton network stability and topology based on empirical long-term data. *Ecological Indicators*, *65*, 76–88. 7

Haberman, S. J. (1972). Log-linear fit for contingency tables—algorithm AS51. *Applied Statistics*, *21*, 218–225. 158

Hallquist, M., & Wiley, J. (2013). MplusAutomation: Automating mplus model estimation and interpretation [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=MplusAutomation` (R package version 0.5-4) 222

Hamaker, E. L. (2012). Why researchers should think "within-person": A paradigmatic rationale. *Handbook of research methods for studying daily life*, 43–61. 72, 101, 103, 113

Hamaker, E. L., Dolan, C. V., & Molenaar, P. C. (2005). Statistical modeling of the individual: Rationale and application of multivariate stationary time series analysis. *Multivariate behavioral research*, *40*(2), 207–233. 219

Hamaker, E. L., & Grasman, R. P. (2014). To center or not to center? investigating inertia with a multilevel autoregressive model. *Frontiers in psychology*, *5*, 1492. 103, 104

Handcock, M., Hunter, D., Butts, C., Goodreau, S., & Morris, M. (2008). statnet: Software tools for the representation, visualization, analysis and simulation of network data. *Journal of Statistical Software*, *24*(1), 1548. 178

Harary, F. (1969). *Graph theory*. Reading, MA, USA: Addison-Wesley. 178

Haslbeck, J. M. B., & Waldorp, L. J. (2016a). mgm: Structure estimation for time-varying mixed graphical models in high-dimensional data. *arXiv preprint*, arXiv:1510.06871. 30, 39, 81, 238

Haslbeck, J. M. B., & Waldorp, L. J. (2016b). Structure estimation for mixed graphical models in high dimensional data. *arXiv preprint*, arXiv:1510.05677. 38

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The Elements of Statistical Learning*. New York, NY, USA: Springer New York Inc. 12, 62

Hastie, T., Tibshirani, R., & Wainwright, M. (2015). *Statistical learning with sparsity: the lasso and generalizations*. Boca Raton, FL, USA: CRC Press. 12, 40, 57

Hayduk, L. A. (1987). *Structural equation modeling with LISREL: Essentials and advances*. Baltimore, MD, USA: Johns Hopkins University Press. 118, 222, 230

Hien, D. A., Wells, E. A., Jiang, H., Suarez-Morales, L., Campbell, A. N. C., Cohen, L. R., ... Nunes, E. V. (2009). Multisite randomized trial of behavioral interventions for women with co-occurring ptsd and substance use disorders. *Journal of consulting and clinical psychology*, *77*(4), 607–619. 37, 44

Hoekstra, H., de Fruyt, F., & Ormel, J. (2003). Neo-Persoonlijkheidsvragenlijsten: NEO-PI-R, NEO-FFI [Neo personality questionnaires: NEO-PI-R, NEO-FFI]. *Lisse: Swets Test Services*. 179, 201

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, *12*(1), 55–67. 134

Hofmann, R. J. (1978). Complexity and simplicity as objective indices descriptive of factor solutions. *Multivariate Behavioral Research*, *13*(2), 247–250. 200, 210

Holland, P. W. (1990). The dutch identity: A new tool for the study of item response models. *Psychometrika*, *55*(1), 5–18. 157

Holland, P. W., & Rosenbaum, P. R. (1986). Conditional association and unidimensionality in monotone latent variable models. *The Annals of Statistics*, *14*, 1523–1543. 118, 122

Holst, K. K., & Budtz-Joergensen, E. (2013). Linear latent variable models: The lava-package. *Computational Statistics*, *28*, 1385–1452. doi: 10.1007/s00180-012-0344-y 234

Holzinger, K. J., & Swineford, F. (1939). A study in factor analysis: The stability of a bi-factor solution. *Supplementary Educational Monographs*, *48*. 224, 226

Howell, R. D., Breivik, E., & Wilcox, J. B. (2007). Reconsidering formative measurement. *Psychological methods*, *12*(2), 205–218. 164

Humphries, M. D., & Gurney, K. (2008). Network 'small-world-ness': a quantitative method for determining canonical network equivalence. *PloS one*, *3*(4), e0002051. 208, 210, 215

Ising, E. (1925). Beitrag zur theorie des ferromagnetismus. *Zeitschrift für Physik A Hadrons and Nuclei*, *31*(1), 253–258. 60, 91, 120, 148

Isvoranu, A. M., Borsboom, D., van Os, J., & Guloksuz, S. (2016). A Network Approach to Environmental Impact in Psychotic Disorders: Brief Theoretical Framework. *Schizophrenia Bulletin*, *42*(4), 870–873. 12, 34, 76, 116

Isvoranu, A. M., van Borkulo, C. D., Boyette, L., Wigman, J. T. W., Vinkers, C. H., Borsboom, D., & GROUP Investigators. (2016). A Network Approach to Psychosis: Pathways between Childhood Trauma and Psychotic Symptoms. *Schizophrenia Bulletin*. (Advance Access published May 10, 2016) 12, 34, 61, 75, 76, 86, 116

Jackson, J. J., Bogg, T., Walton, K. E., Wood, D., Harms, P. D., Lodi-Smith, J., . . . Roberts, B. W. (2009). Not all conscientiousness scales change alike: a multimethod, multisample study of age differences in the facets of conscientiousness. *Journal of personality and social psychology*, *96*(2), 446–459. 212

Jacobucci, R., Grimm, K. J., & McArdle, J. J. (2016). Regularized structural equation modeling. *Structural Equation Modeling: A Multidisciplinary Journal*, *23*(4), 555-566. 118, 127, 134

Jaya, E. S., Hillmann, T. E., Reininger, K. M., Gollwitzer, A., & Lincoln, T. M. (2016). Loneliness and psychotic symptoms: The mediating role of depression. *Cognitive Therapy and Research*. Retrieved from `http://dx.doi.org/10.1007/s10608-016-9799-4` 12

Jensen, A. R. (1998). *The g factor: The science of mental ability*. Westport, CT, USA: Praeger. 164, 165

Jeong, H., Mason, S. P., Barabási, A.-L., & Oltvai, Z. N. (2001). Lethality and centrality in protein networks. *Nature*, *411*(6833), 41–42. 206

Jolliffe, I. (2002). *Principal Component Analysis*. Wiley Online Library. 188

Jöreskog, K. G. (1967). A general approach to confirmatory maximum likelihood factor analysis. *ETS Research Bulletin Series*, *1967*(2), 183–202. 118

Jöreskog, K. G., & Sörbom, D. (1996). *LISREL 8: User's reference guide*. Scientific Software. 222, 232

Kac, M. (1966). *Mathematical mechanism of phase transition*. New York, NY, USA: Gordon & Breach. 63, 152, 171

Kalisch, M., & Bühlmann, P. (2007). Estimating high-dimensional directed acyclic graphs with the pc-algorithm. *Journal of Machine Learning Research*, *8*(Mar), 613–636. 112

Kalisch, M., Mächler, M., Colombo, D., Maathuis, M. H., & Bühlmann, P. (2012). Causal inference using graphical models with the R package pcalg. *Journal of Statistical Software*, *47*(11), 1–26. 3, 112

Kalisch, M., Maechler, M., & Colombo, D. (2010). pcalg: Estimation of cpdag/pag and causal inference using the ida algorithm [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=pcalg` (R package version 1.1-2) 179

Kalna, G., & Higham, D. J. (2007). A clustering coefficient for weighted networks, with application to gene expression data. *AI Communications*, *20*(4), 263–271. 209

Kaplan, D. (2000). *Structural equation modeling: Foundations and extensions.* Thousand Oaks, CA, USA: Sage. 116

Kass, R. E., & Raftery, A. E. (1995). Bayes factors. *Journal of the american statistical association*, *90*(430), 773–795. 241

Kessler, R. C., Chiu, W. T., Demler, O., & Walters, E. E. (2005). Prevalence, severity, and comorbidity of 12-month DSM-IV disorders in the National Comorbidity Survey Replication. *Archives of general psychiatry*, *62*(6), 617–627. 65

Kindermann, R., Snell, J. L., et al. (1980). *Markov random fields and their applications* (Vol. 1). Providence, RI, USA: American Mathematical Society. 202

Klaiber, J., Epskamp, S., & van der Maas, H. L. J. (2015). Estimating ising models on complete and incomplete psychometric data. *Unpublished master's thesis.* Retrieved from `http://www.scriptiesonline.uba.uva.nl/571364` 239

Knefel, M., Tran, U. S., & Lueger-Schuster, B. (2016). The association of posttraumatic stress disorder, complex posttraumatic stress disorder, and borderline personality disorder from a network analytical perspective. *Journal of Anxiety Disorders*, *43*, 70–78. 12

Kolaczyk, E. D. (2009). *Statistical analysis of network data.* New York, NY, USA: Springer. 5, 37, 126, 158, 205, 206, 207

Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: principles and techniques.* Cambridge, MA, USA: MIT press. 22, 37, 86, 126, 241

Kossakowski, J. J., Epskamp, S., Kieffer, J. M., van Borkulo, C. D., Rhemtulla, M., & Borsboom, D. (2016). The application of a network approach to health-related quality of life (HRQoL): Introducing a new method for assessing hrqol in healthy adults and cancer patient. *Quality of Life Research*, *25*, 781-92. 12, 34, 61, 86, 116, 119

Krämer, N., Schäfer, J., & Boulesteix, A.-L. (2009). Regularized estimation of large-scale gene association networks using graphical gaussian models. *BMC Bioinformatics*, *10*(1), 1–24. 14, 30, 39, 93, 204, 212, 214, 220

Kroeze, R., Van Veen, D. C., Servaas, M. N., Bastiaansen, J. A., Oude Voshaar, R., Borsboom, D., . . . Riese, H. (2016). Personalized feedback on symptom dynamics of psychopathology: A proof-of-principle study. *Manuscript under revision*. 76, 78

Kruis, J., & Maris, G. (2015). On the mathematical equivalence, biological plausibility, & sparsity assumptions of network models. *Unpublished master's thesis*. 70

Kruis, J., & Maris, G. (2016). Three representations of the ising model. *Scientific reports*, *6*(34175), 1–11. 70

Kunegis, J., Lommatzsch, A., & Bauckhage, C. (2009). The slashdot zoo: mining a social network with negative edges. In *Proceedings of the 18th international conference on world wide web* (pp. 741–750). 205, 209

Lane, S., Gates, K., Molenaar, P., Hallquist, M., & Pike, H. (2016). gimme: Group iterative multiple model estimation [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=gimme` (R package version 0.1-7) 112

Langfelder, P., & Horvath, S. (2012). Fast R functions for robust correlations and hierarchical clustering. *Journal of statistical software*, *46*(11). 206

Langley, D. J., Wijn, R., Epskamp, S., & Van Bork, R. (2015). Should I Get That Jab? Exploring Influence to Encourage Vaccination via Online Social Media. *ECIS 2015 Research-in-Progress Papers*, Paper 64. 12, 61

Latora, V., Nicosia, V., & Panzarasa, P. (2013). Social cohesion, structural holes, and a tale of two measures. *Journal of Statistical Physics*, *151*(3-4), 745–764. 208

Lauritzen, S. L. (1996). *Graphical models.* Oxford, UK: Clarendon Press. 2, 12, 13, 25, 34, 37, 38, 60, 62, 75, 77, 86, 89, 91, 105, 116, 121, 126, 145, 170, 202, 204, 241

Lawley, D. N. (1940). VI.—the estimation of factor loadings by the method of maximum likelihood. *Proceedings of the Royal Society of Edinburgh*, *60*(01), 64–82. 118

Lee, J. J. (2012). Correlation and causation in the study of personality. *European Journal of Personality*, *26*(4), 372–390. 197

Lee, S.-I., Lee, H., Abbeel, P., & Ng, A. Y. (2006). Efficient ℓ_1 regularized logistic regression. In *Proceedings of the national conference on artificial intelligence* (Vol. 21, p. 401). 160

Leskovec, J., Huttenlocher, D., & Kleinberg, J. (2010). Signed networks in social media. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 1361–1370). 205

Levine, S. Z., & Leucht, S. (2016). Identifying a system of predominant negative symptoms: Network analysis of three randomized clinical trials. *Schizophrenia Research*. Retrieved from `http://dx.doi.org/10.1016/j.schres.2016.09.002` 12

Li, Z., & Hong, F. (2014). plRasch: Log linear by linear association models and rasch family models by pseudolikelihood estimation [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=plRasch` (R package version 1.0) 157

Liepmann, D., Beauducel, A., Brocke, B., & Amthauer, R. (2010). *Intelligentie Structuur Test.* Amsterdam, NL: Hogrefe Uitgevers B.V. (Dutch Translation by H.C.M. Vorst) 193

Lin, K.-Y. (1992). Spontaneous magnetization of the Ising model. *Chinese Journal of Physics*, *30*(3), 287–319. 152

Little, J. (2013). Multilevel confirmatory ordinal factor analysis of the life skills profile-16. *Psychological Assessment*, *25*(3), 810–825. 224

Liu, H., Han, F., Yuan, M., Lafferty, J. D., & Wasserman, L. (2012). High-dimensional semiparametric gaussian copula graphical models. *The Annals of Statistics*, *40*(4), 2293–2326. 38, 170

Liu, H., Lafferty, J. D., & Wasserman, L. (2009). The nonparanormal: Semiparametric estimation of high dimensional undirected graphs. *The Journal of Machine Learning Research*, *10*, 2295–2328. 13, 170

Liu, Q., & Ihler, A. (2012). Distributed parameter estimation via pseudolikelihood. *Proceedings of the International Conference on Machine Learning (ICML)*. 159

Lord, F. M., Novick, M. R., & Birnbaum, A. (1968). *Statistical theories of mental test scores.* Oxford, UK: Addison-Wesley. 118

Ly, A., Verhagen, A., & Wagenmakers, E.-J. (2016). Harold Jeffreys's default Bayes factor hypothesis tests: Explanation, extension, and application in psychology. *Journal of Mathematical Psychology*, *72*, 19–32. doi: http://dx.doi.org/10.1016/j.jmp.2015.06.004 241

Lykken, D. T. (1968). Statistical significance in psychological research. *Psychological bulletin*, *70*(3p1), 151–159. 202

Lykken, D. T. (1991). What's wrong with psychology anyway. *Thinking clearly about psychology*, *1*, 3–39. 202, 219

MacCallum, R. C., Wegener, D. T., Uchino, B. N., & Fabrigar, L. R. (1993). The problem of equivalent models in applications of covariance structure analysis. *Psychological bulletin*, *114*(1), 185–199. 4, 37, 90, 117, 124

Mair, P., & Wu, E. (2012). REQS: R/EQS interface [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=REQS` (R package version 0.8-12) 234

Marchetti, G. M., Drton, M., & Sadeghi, K. (2015). ggm: Functions for graphical markov models [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=ggm` (R package version 2.3) 93

Marcoulides, G. A., & Papadopoulos, D. (1993). Lispath: a program for generating structural equation path diagrams. *Educational and psychological measurement*, *53*(3), 675–678. 222

Markus, K. A., & Borsboom, D. (2013a). *Frontiers of test validity theory: Measurement, causation, and meaning.* New York, NY, USA: Routledge. 113

Markus, K. A., & Borsboom, D. (2013b). Reflective measurement models, behavior domains, and common causes. *New Ideas in Psychology*, *31*(1), 54–64. 164, 165

Marsh, H. W., Morin, A. J., Parker, P. D., & Kaur, G. (2014). Exploratory structural equation modeling: An integration of the best features of exploratory and confirmatory factor analysis. *Annual Review of Clinical Psychology*, *10*, 85–110. 127

Marsman, M., Maris, G., Bechger, T., & Glas, C. (2015). Bayesian inference for low-rank ising networks. *Scientific reports*, *5*(9050), 1–7. 62, 63, 70, 155, 162, 245

McArdle, J. J., & McDonald, R. P. (1984). Some algebraic properties of the reticular action model for moment structures. *British Journal of Mathematical and Statistical Psychology*, *37*(2), 234–251. 222, 230

McCrae, R. R., & Costa, P. T. (1997). Personality Trait Structure as a Human Universal. *American Psychologist*, *52*(5), 509–516. 1, 25, 92, 138, 179

McCrae, R. R., & Costa, P. T. (2008). Empirical and theoretical status of the five-factor model of personality traits. *Sage handbook of personality theory and assessment*, *1*, 273–294. 164, 165, 197, 201

McCrae, R. R., & John, O. P. (1992). An introduction to the five-factor model and its applications. *Journal of personality*, *60*(2), 175–215. 106

McDonald, R. P. (2003). Behavior domains in theory and in practice. *Alberta Journal of Educational Research*, *49*, 212–230. 165, 169

McGill, R., Tukey, J. W., & Larsen, W. A. (1978). Variations of box plots. *The American Statistician*, *32*(1), 12–16. 130

McNally, R. J. (2016). Can network analysis transform psychopathology? *Behaviour Research and Therapy*, *86*, 95–104. 12

McNally, R. J., Robinaugh, D. J., Wu, G. W., Wang, L., Deserno, M. K., & Borsboom, D. (2015). Mental disorders as causal systems a network approach to posttraumatic stress disorder. *Clinical Psychological Science*, *3*(6), 836–849. 13, 34, 75, 86, 116

Meehl, P. E. (1990). Why summaries of research on psychological theories are often uninterpretable. *Psychological Reports*, *66*(1), 195–244. 202

Meinshausen, N., & Bühlmann, P. (2006). High-dimensional graphs and variable selection with the lasso. *The annals of statistics*, 1436–1462. 12, 91, 92, 105, 159, 170

Meinshausen, N., Meier, L., & Bühlmann, P. (2009). P-values for high-dimensional regression. *Journal of the American Statistical Association*, *104*(488), 1671–1681. 161

Mellenbergh, G. J. (1989). Item bias and item response theory. *International Journal of Educational Research*, *13*(2), 127–143. 167

Meredith, W. (1993). Measurement invariance, factor analysis and factorial invariance. *Psychometrika*, *58*(4), 525–543. 167, 226

Milgram, S. (1967). The small world problem. *Psychology today*, *2*(1), 60–67. 209

Mohammadi, A., & Wit, E. C. (2015). BDgraph: An R package for Bayesian structure learning in graphical models. *arXiv preprint*, arXiv:1501.05108. 112

Mohammadi, A., Wit, E. C., et al. (2015). Bayesian structure learning in sparse gaussian graphical models. *Bayesian Analysis*, *10*(1), 109–138. 220

Møller, J., Pettitt, A. N., Reeves, R., & Berthelsen, K. K. (2006). An efficient markov chain monte carlo method for distributions with intractable normalising constants. *Biometrika*, *93*(2), 451–458. 158

Monecke, A., & Leisch, F. (2012). semPLS: Structural equation modeling using partial least squares. *Journal of Statistical Software*, *48*(3), 1–32. 234

Mooij, J. M., Peters, J., Janzing, D., Zscheischler, J., & Schölkopf, B. (2016). Distinguishing cause from effect using observational data: Methods and benchmarks. *Journal of Machine Learning Research*, *17*(32), 1-102. Retrieved from `http://jmlr.org/papers/v17/14-518.html` 37, 38

Mõttus, R., Epskamp, S., & Francis, A. (2016). Within-and between individual variability of personality characteristics and physical exercise. *Journal of Research in Personality*. doi: 10.1016/j.jrp.2016.06.017 87, 106

Mõttus, R., Penke, L., Murray, A. L., Booth, T., & Allerhand, M. (2014). Personality differences without common-cause latent factors are possible and can explain key findings in personality psychology. *Manuscript submitted for publication*. 218, 220

Mulaik, S. A., & McDonald, R. P. (1978). The effect of additional variables on factor indeterminacy in models with a single common factor. *Psychometrika*, *43*(2), 177–192. 165

Mulder, J. (2014). Bayes factors for testing order-constrained hypotheses on correlations. *Journal of Mathematical Psychology*, *72*, 104–115. 241

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. Cambridge, MA, USA: MIT press. 2, 12, 86, 91, 121, 145, 146, 151

Murray, I. (2007). *Advances in markov chain monte carlo methods* (Unpublished doctoral dissertation). Gatsby Computational neuroscience unit, University College London. 158

Murray, I., Ghahramani, Z., & MacKay, D. J. C. (2006). MCMC for doubly-intractable distributions. In *Uncertainty in artificial intellegence (UAI)* (pp. 359–366). AUAI Press. 158

Musek, J. (2007). A general factor of personality: Evidence for the big one in the five-factor model. *Journal of Research in Personality*, *41*(6), 1213–1233. 202

Muthén, B. (1984). A general structural equation model with dichotomous, ordered categorical, and continuous latent variable indicators. *Psychometrika*, *49*(1), 115–132. 240

Muthén, B. O. (1998–2004). *Mplus technical appendices*. Los Angeles, CA, USA: Muthén & Muthén. 222, 230

Muthén, L. K., & Muthén, B. O. (1998–2012). *Mplus user's guide*. (Seventh Edition ed., Vol. 5). Los Angeles, CA, USA: Muthén & Muthén. 222, 229, 234

Myin-Germeys, I., Oorschot, M., Collip, D., Lataster, J., Delespaul, P., & van Os, J. (2009). Experience sampling research in psychopathology: opening the black box of daily life. *Psychological medicine*, *39*(09), 1533–1547. 72, 86

Neale, M. C., Hunter, M. D., Pritikin, J. N., Zahery, M., Brick, T. R., Kirkpatrick, R. M., ... Boker, S. M. (2016). Openmx 2.0: Extended structural equation and statistical modeling. *Psychometrika*, *81*(2), 535–549. 127

Newman, M. E. J. (2003). The structure and function of complex networks. *SIAM review*, *45*(2), 167–256. 209

Newman, M. E. J. (2004). Analysis of weighted networks. *Physical Review E*, *70*(5), 056131. 37, 206, 208

Newman, M. E. J. (2005). A measure of betweenness centrality based on random walks. *Social networks*, *27*(1), 39–54. 208

Newman, M. E. J. (2010). *Networks: an introduction*. Oxford, UK: Oxford University Press. 3, 5, 6, 34, 77, 195, 205, 208

Newman, M. E. J., & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical review E*, *69*(2), 026113. 208

Olkin, I., & Tate, R. F. (1961). Multivariate correlation models with mixed discrete and continuous variables. *The Annals of Mathematical Statistics*, *32*, 448–465. 157

Onnela, J.-P., Saramäki, J., Kertész, J., & Kaski, K. (2005). Intensity and coherence of motifs in weighted complex networks. *Physical Review E*, *71*(6), 065103. 209

Open Science Collaboration. (2015). Estimating the reproducibility of psychological science. *Science*, *349*(6251), aac4716–aac4716. 36

Opsahl, T., Agneessens, F., & Skvoretz, J. (2010). Node centrality in weighted networks: Generalizing degree and shortest paths. *Social Networks*, *32*(3), 245–251. 6, 34, 77, 178, 207, 208, 220, 241

Opsahl, T., & Panzarasa, P. (2009). Clustering in weighted networks. *Social networks*, *31*(2), 155–163. 209

Ouellette, D. V. (1981). Schur complements and statistics. *Linear Algebra and its Applications*, *36*, 187–295. 89

Park, T., & Casella, G. (2008). The Bayesian lasso. *Journal of the American Statistical Association*, *103*(482), 681–686. 241

Pe, M. L., Kircanski, K., Thompson, R. J., Bringmann, L. F., Tuerlinckx, F., Mestdagh, M., . . . Gotlib, I. H. (2015). Emotion-network density in major depressive disorder. *Clinical Psychological Science*, *3*(2), 292–300. 72

Pearl, J. (2000). *Causality: Models, Reasoning, and Inference.* Cambridge Univ Pr. 3, 13, 22, 37, 62, 76, 77, 90, 91, 102, 112, 124, 145, 178, 190, 240

Pearl, J., & Verma, T. S. (1995). A theory of inferred causation. *Studies in Logic and the Foundations of Mathematics*, *134*, 789–811. 198

Pettersson, E., & Turkheimer, E. (2010). Item selection, evaluation, and simple structure in personality data. *Journal of research in personality*, *44*(4), 407–420. 200

Plate, T. (2009). RSVGTipsDevice: An R svg graphics device with dynamic tips and hyperlinks [Computer software manual]. (R package version 1.0-1) 184, 187

Pornprasertmanit, S., Miller, P., Schoemann, A., & Rosseel, Y. (2013). sem-Tools: Useful tools for structural equation modeling. [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=semTools` (R package version 0.4-0) 226

Pötscher, B. M., & Leeb, H. (2009). On the distribution of penalized maximum likelihood estimators: The LASSO, SCAD, and thresholding. *Journal of Multivariate Analysis*, *100*(9), 2065–2082. 42

Pourahmadi, M. (2011). Covariance estimation: The GLM and regularization perspectives. *Statistical Science*, 369–387. 91, 202

Quax, R., Apolloni, A., & Sloot, P. M. (2013). The diminishing role of hubs in dynamical processes on complex networks. *Journal of The Royal Society Interface*, *10*(88), 20130568. 243

Quax, R., Kandhai, D., & Sloot, P. M. (2013). Information dissipation as an early-warning signal for the lehman brothers collapse in financial time series. *Scientific reports*, *3*(1898). 243

R Core Team. (2016). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from `https://www.R-project.org/` 12, 14, 36, 93, 158, 179, 188, 196, 221

Rasch, G. (1960). *Probabilistic models for some intelligence and attainment tests.* Copenhagen, DM: Danish Institute for Educational Research. 154

Ravikumar, P., Wainwright, M. J., & Lafferty, J. D. (2010). High-dimensional Ising model selection using ℓ_1-regularized logistic regression. *The Annals of Statistics*, *38*(3), 1287–1319. 159, 160

Reckase, M. D. (2009). *Multidimensional item response theory.* New York, NY, USA: Springer. 62, 64, 120, 154

Reichenbach, H. (1991). *The direction of time* (Vol. 65). Berkeley, CA, USA: University of California Press. 164

Reingold, E. M., & Tilford, J. S. (1981). Tidier drawings of trees. *Software Engineering, IEEE Transactions on*, *SE-7*(2), 223–228. 232

Reise, S. P., & Waller, N. G. (2009). Item response theory and clinical measurement. *Annual review of clinical psychology*, *5*, 27–48. 165

Revelle, W. (2010). psych: Procedures for psychological, psychometric, and personality research [Computer software manual]. Evanston, Illinois. Retrieved from `http://personality-project.org/r/psych.manual.pdf` (R package version 1.0-93) 1, 25, 92, 117, 138, 179, 188, 199

Rhemtulla, M., Brosseau-Liard, P. E., & Savalei, V. (2012). When can categorical variables be treated as continuous? a comparison of robust continuous and categorical sem estimation methods under suboptimal conditions. *Psychological methods*, *17*(3), 354–373. 239

Rhemtulla, M., Fried, E. I., Aggen, S. H., Tuerlinckx, F., Kendler, K. S., & Borsboom, D. (2016). Network analysis of substance abuse and dependence symptoms. *Drug and alcohol dependence*, *161*, 230–237. 24, 61

Rosa, M., Friston, K., & Penny, W. (2012). Post-hoc selection of dynamic causal models. *Journal of neuroscience methods*, *208*(1), 66–78. 124

Rosseel, Y. (2012). lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, *48*(2), 1–36. 18, 26, 45, 177, 178, 179, 190, 221, 222

Rothman, A. J., Levina, E., & Zhu, J. (2010). Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics*, *19*(4), 947–962. 31, 79, 98

RStudio, & Inc. (2013). shiny: Web application framework for R [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=shiny` (R package version 0.5.0) 234

Ruzzano, L., Borsboom, D., & Geurts, H. M. (2015). Repetitive Behaviors in Autism and Obsessive–Compulsive Disorder: New Perspectives from a Network Analysis. *Journal of Autism and Developmental Disorders*, *45*(1), 192–202. 75

Sabidussi, G. (1966). The centrality index of a graph. *Psychometrika*, *31*(4), 581–603. 207

Saramäki, J., Kivelä, M., Onnela, J.-P., Kaski, K., & Kertesz, J. (2007). Generalizations of the clustering coefficient to weighted complex networks. *Physical Review E*, *75*(2), 027105. 209

Saucier, G., Thalmayer, A. G., Payne, D. L., Carlson, R., Sanogo, L., Ole-Kotikash, L., ... Zhou, X. (2014). A basic bivariate structure of personality attributes evident across nine languages. *Journal of Personality*, *82*(1), 1–14. 219

Schäfer, J., Opgen-Rhein, R., Zuber, V., Ahdesmäki, M., Silva, A. P. D., & Strimmer., K. (2015). corpcor: Efficient estimation of covariance and (partial) correlation [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=corpcor` (R package version 1.6.8) 93

Scheffer, M., Bascompte, J., Brock, W. A., Brovkin, V., Carpenter, S. R., Dakos, V., ... Sugihara, G. (2009). Early-warning signals for critical transitions. *Nature*, *461*(7260), 53–59. 168

Schlegel, K., Grandjean, D., & Scherer, K. R. (2013). Constructs of social and emotional effectiveness: Different labels, same content? *Journal of Research in personality*, *47*(4), 249–253. 201

Schmittmann, V. D., Cramer, A. O. J., Waldorp, L. J., Epskamp, S., Kievit, R. A., & Borsboom, D. (2013). Deconstructing the construct: A network perspective on psychological phenomena. *New Ideas in Psychology*, *31*(1), 43–53. 2, 11, 24, 59, 62, 86, 116, 119, 120, 124, 178, 196, 237

Schuurman, N. K., Grasman, R. P. P. P., & Hamaker, E. L. (2016). A comparison of inverse-wishart prior specifications for covariance matrices in multilevel autoregressive models. *Multivariate Behavioral Research*, *51*(2-3), 185-206. 103

Scutari, M. (2010). Learning Bayesian Networks with the bnlearn R Package. *Journal of Statistical Software*, *35*(3), 1–22. 3, 4, 112

Sebastiani, G., & Sørbye, S. H. (2002). A bayesian method for multispectral image data classification. *Journal of Nonparametric Statistics*, *14*(1-2), 169–180. 158

Selig, J. P., & Little, T. D. (2012). Autoregressive and Cross- Lagged Panel Analysis for Longitudinal Data. *Handbook of Developmental Research Methods.*(Chapter 12), 265–278. 73

Servaas, M. N., Riese, H., Ormel, J., & Aleman, A. (2014). The neural correlates of worry in association with individual differences in neuroticism. *Human brain mapping*, *35*(9), 4303–4315. 208

Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., . . . Ideker, T. (2003). Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome research*, *13*(11), 2498–2504. 222

Sharpsteen, C., & Bracken, C. (2010). tikzDevice: A device for R graphics output in pgf/tikz format [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=tikzDevice` (R package version 0.5.3) 184

Sherman, R. A., Nave, C. S., & Funder, D. C. (2010). Situational similarity and personality predict behavioral consistency. *Journal of personality and social psychology*, *99*(2), 330–343. 212

Silva, A. R. d., Rêgo, E. R. d., Pessoa, A. M. d. S., & Rêgo, M. M. d. (2016). Correlation network analysis between phenotypic and genotypic traits of chili pepper. *Pesquisa Agropecuária Brasileira*, *51*(4), 372–377. 6, 7

Spearman, C. (1904). "general intelligence," objectively determined and measured. *The American Journal of Psychology*, *15*(2), 201–292. 164

Spirtes, P., Glymour, C., & Scheines, R. (2000). *Causation, prediction, and Search.* Cambridge, MA, USA: The MIT Press. 178

Sporns, O., Chialvo, D. R., Kaiser, M., & Hilgetag, C. C. (2004). Organization, development and function of complex brain networks. *Trends in cognitive sciences*, *8*(9), 418–425. 37

Stevens, J. (1996). *Applied Multivariate Statistics for the Social Sciences.* Hillsdale, NJ, USA: Lawrence Erlbaum Associates. 188, 190

Stevens, S. S. (1946). On the theory of scales of measurement. *Science, New Series*, *103*(2684), 677–680. 239

Strimmer., K. (2011). fdrtool: Estimation and control of (local) false discovery rates [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=fdrtool` (R package version 1.2.7) 187

Sytema, S., & Van der Krieke, L. (2013). Routine outcome monitoring: A tool to improve the quality of mental health care? In G. Thornicroft, M. Ruggeri, & D. Goldberg (Eds.), *Improving mental health care: The global challenge* (pp. 246–263). Chichester, UK: John Wiley & Sons. 78

Tett, R. P., & Guterman, H. A. (2000). Situation trait relevance, trait expression, and cross-situational consistency: Testing a principle of trait activation. *Journal of Research in Personality*, *34*(4), 397–423. 212

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, *58*, 267–288. 13, 38, 61, 92, 117, 134, 159

Valente, T. W. (2012). Network interventions. *Science*, *337*(6090), 49–53. 206

Van Bork, R. (2015). Latent variable and network model implications for partial correlation structures. In *80th annual meeting of the Psychometric Society (IMPS)*. 24

Van Borkulo, C., Boschloo, L., Kossakowski, J., Tio, P., Schoevers, R., Borsboom, D., & Waldorp, L. (2016). Comparing network structures on three aspects: A permutation test. *Manuscript submitted for publication*. 24

van Borkulo, C. D. (2016). NetworkComparisonTest: Statistical comparison of two networks based on three invariance measures [Computer software manual]. Retrieved from `github.com/cvborkulo/NetworkComparisonTest` (R package version 2.0.0) 238

van Borkulo, C. D., Borsboom, D., Epskamp, S., Blanken, T. F., Boschloo, L., Schoevers, R. A., & Waldorp, L. J. (2014). A new method for constructing networks from binary data. *Scientific reports*, *4*(5918), 1–10. 14, 26, 30, 37, 38, 39, 61, 62, 67, 69, 75, 86, 92, 107, 109, 128, 129, 140, 160, 161, 238

van Borkulo, C. D., Boschloo, L., Borsboom, D., Penninx, B. W. J. H., Waldorp, L. J., & Schoevers, R. A. (2015). Association of Symptom Network Structure With the Course of Depression. *JAMA psychiatry*, *72*(12), 1219–1226. 12, 34, 61, 76, 86, 116

van Borkulo, C. D., & Epskamp, S. (2014). IsingFit: Fitting Ising models using the elasso method [Computer software manual]. Retrieved from `http://CRAN.R-project.org/package=IsingFit` (R package version 0.2.0) 30, 61, 63, 161

van de Geer, S., Bühlmann, P., & Ritov, Y. (2013). On asymptotically optimal confidence regions and tests for high-dimensional models. *arXiv preprint*, arXiv:1303.0518. 161, 162

van de Leemput, I. A., Wichers, M., Cramer, A. O. J., Borsboom, D., Tuerlinckx, F., Kuppens, P., . . . Scheffer, M. (2014). Critical slowing down as early warning for the onset and termination of depression. *Proceedings of the National Academy of Sciences*, *111*(1), 87-92. 60, 168, 247

van der Krieke, L., Emerencia, A. C., Bos, E. H., Rosmalen, J. G., Riese, H., Aiello, M., . . . de Jonge, P. (2015). Ecological Momentary Assessments and Automated Time Series Analysis to Promote Tailored Health Care: A Proof-of-Principle Study. *JMIR research protocols*, *4*(3), e100. 72

Van Der Maas, H. L., Dolan, C. V., Grasman, R. P., Wicherts, J. M., Huizenga, H. M., & Raijmakers, M. E. (2006). A dynamical model of general intelligence: the positive manifold of intelligence by mutualism. *Psychological review*, *113*(4), 842–861. 1, 86, 120, 143, 144, 164, 166, 169, 218

von Oertzen, T., Brandmaier, A. M., & Tsang, S. (2013). Onyx user guide [Computer software manual]. Retrieved from `http://onyx.brandmaier.de/` 222

Wagenmakers, E.-J. (2007). A practical solution to the pervasive problems ofp values. *Psychonomic bulletin & review*, *14*(5), 779–804. 16, 42, 240

Wainwright, M. J., & Jordan, M. I. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, *1*(1-2), 1–305. 151

Ware Jr, J. E., & Sherbourne, C. D. (1992). The MOS 36-item short-form health survey (SF-36): I. Conceptual framework and item selection. *Medical Care*, *30*, 473–483. 119

Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications.* Cambridge, UK: Cambridge University Press. 12, 34

Watts, D., & Strogatz, S. (1998). Collective dynamics of small-world networks. *Nature*, *393*(6684), 440–442. 5, 34, 50, 205, 208, 209

Wetzels, R., & Wagenmakers, E.-J. (2012). A default bayesian hypothesis test for correlations and partial correlations. *Psychonomic Bulletin & Review*, *19*(6), 1057–1064. 241

Whittaker, J. (1990). *Graphical models in applied multivariate statistics.* Chichester, UK: John Wiley & Sons. 159

Wichers, M., Groot, P. C., Psychosystems, ESM Group, & EWS Group. (2016). Critical Slowing Down as a Personalized Early Warning Signal for Depression. *Psychotherapy and psychosomatics*, *85*(2), 114–116. 60, 81, 247

Wichers, M., Lothmann, C., Simons, C. J., Nicolson, N. A., & Peeters, F. (2012). The dynamic interplay between negative and positive emotions in daily life predicts response to treatment in depression: a momentary assessment study. *British Journal of Clinical Psychology*, *51*(2), 206–222. 72

Wickens, T. D. (1989). *Multiway contingency tables analysis for the social sciences.* Hillsdale, NJ, USA: Lawrence Erlbaum Associates. 153

Wigman, J., van Os, J., Borsboom, D., Wardenaar, K., Epskamp, S., Klippel, A., ... Wichers, M. (2015). Exploring the underlying structure of mental disorders: cross-diagnostic differences and similarities from a network perspective using both a top-down and a bottom-up approach. *Psychological medicine*, *45*(11), 2375–2387. 72, 86

Wild, B., Eichler, M., Friederich, H.-C., Hartmann, M., Zipfel, S., & Herzog, W. (2010). A graphical vector autoregressive modelling approach to the analysis of electronic diary data. *BMC medical research methodology*, *10*(1), 28. 31, 73, 79, 97

Wilkins, M. R., Shizuka, D., Joseph, M. B., Hubbard, J. K., & Safran, R. J. (2015). Multimodal signalling in the north american barn swallow: a phenotype network approach. *Proceedings of the Royal Society B*, *282*(1816), 20151574. 7

Witten, D. M., Friedman, J. H., & Simon, N. (2011). New insights and faster computations for the graphical lasso. *Journal of Computational and Graphical*

Statistics, *20*(4), 892–900. 92

Woodward, J. (2005). *Making things happen: A theory of causal explanation.* Oxford, UK: Oxford University Press. 102

Wright, S. (1921). Correlation and causation. *Journal of agricultural research*, *20*(7), 557–585. 116

Wright, S. (1934). The method of path coefficients. *The Annals of Mathematical Statistics*, *5*(3), 161–215. 118

Yin, J., & Li, H. (2011). A sparse conditional gaussian graphical model for analysis of genetical genomics data. *The annals of applied statistics*, *5*(4), 2630–2650. 114, 128

Yuan, M. (2012). Discussion: Latent variable graphical model selection via convex optimization. *The Annals of Statistics*, *40*, 1968–1972. 220

Yuan, M., & Lin, Y. (2007). Model selection and estimation in the gaussian graphical model. *Biometrika*, *94*(1), 19–35. 92

Zhang, B., Horvath, S., et al. (2005). A general framework for weighted gene co-expression network analysis. *Statistical applications in genetics and molecular biology*, *4*(1), 1128. 209, 213

Zhao, L. P., & Prentice, R. L. (1990). Correlated binary regression using a quadratic exponential model. *Biometrika*, *77*(3), 642–648. 154

Zhao, P., & Yu, B. (2006). On model selection consistency of lasso. *The Journal of Machine Learning Research*, *7*, 2541–2563. 14

Zhao, T., Li, X., Liu, H., Roeder, K., Lafferty, J., & Wasserman, L. (2015). huge: High-dimensional undirected graph estimation [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=huge` (R package version 1.2.7) 14, 30, 39, 93

Ziegler, M., Booth, T., & Bensch, D. (2013). Getting entangled in the nomological net. *European Journal of Psychological Assessment*, 157–161. 6, 201

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, *101*(476), 1418–1429. 204

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *67*(2), 301–320. 134, 161

Zou, H., Hastie, T., Tibshirani, R., et al. (2007). On the "degrees of freedom" of the lasso. *The Annals of Statistics*, *35*(5), 2173–2192. 135

Appendix B

Contributed Work

This appendix contains a list of the contributed publications and software packages until the completion of this PhD project. The PDF version of the dissertation contains links to all publications and software repositories, which can also be found on my website at:

<div align="center">

www.sachaepskamp.com

</div>

B.1 Publications

This section lists all manuscripts that are submitted or accepted for publication at the time of submission of this dissertation. As such, some of the titles and journals of yet unpublished work might change before publications as a result of the review process.

Main Author Publications

- **Epskamp, S.**, and Fried, E.I. A Tutorial on Regularized Partial Correlation Networks.

    - Chapter 2
    - Under review at *Psychological Methods*

- **Epskamp, S.** , Regularized Gaussian Psychological Networks: Brief Report on the Performance of Extended BIC Model Selection.

    - Section 2.6

- **Epskamp, S.**, Borsboom, D., and Fried, E.I. (in press). Estimating Psychological Networks and their Accuracy: A Tutorial Paper. *Behavior Research Methods.*

    - Chapter 3 and Chapter 1

- **Epskamp, S.**, Kruis, J., and Marsman, M. (in press). Estimating psychopathological networks: be careful what you wish for. *PlosOne*.

    – Chapter 4

- **Epskamp, S.**, van Borkulo, C.D., van der Veen, D.C., Servaas, M.N., Isvoranu, A.M., Riese, H., and Cramer, A.O.J. Personalized Network Modeling in Psychopathology: The Importance of Contemporaneous and Temporal Connections.

    – Chapter 5

    – Submitted to *Psychological Medicine*

- **Epskamp, S.**, Waldorp, L.J., Mõttus, R., and Borsboom, D. Discovering Psychological Dynamics: The Gaussian Graphical Model in Cross-sectional and Time-series Data.

    – Chapter 6

    – Under review at *Multivariate Behavioral Research*

- **Epskamp, S.**, Rhemtulla, M., and Borsboom, D. (in press). Generalized Network Psychometrics: Combining Network and Latent Variable models. *Psychometrika*.

    – Chapter 7

- **Epskamp, S.**, Maris, G., Waldorp, L. J., and Borsboom, D. (in press). Network Psychometrics. To appear in: Irwing, P., Hughes, D., and Booth, T. (Eds.), *Handbook of Psychometrics*. New York: Wiley.

    – Chapter 8

- **Epskamp, S.**, Cramer, A.O.J., Waldorp, L.J., Schmittmann, V.D. and Borsboom, D. (2012) qgraph: Network Visualizations of Relationships in Psychometric Data. *Journal of Statistical Software, 48(4)*, 1-18.

    – Chapter 9

- Costantini, G., **Epskamp, S.** (the first two authors contributed equally to this work), Borsboom, D., Perugini, M., Mõttus, R., Waldorp, L. J., and Cramer, A. O. (2014). State of the aRt personality research: A tutorial on network analysis of personality data in R. *Journal of Research in Personality, 54*, 13-29.

    – Chapter 10

- **Epskamp, S.** (2015). semPlot: Unified visualizations of Structural Equation Models. *Structural Equation Modeling, 22*, 747-757.

    – Chapter 11

- Van Bork, R., **Epskamp, S.** (the first two authors contributed equally to this work), Rhemtulla, M., Borsboom, D., and Van der Maas, H.L.J. What is P? Some Risks of General Factor Modeling.

    - Under review at *Theory and Psychology*

Collaborations

- Golino, H., **Epskamp, S.** (in press). Exploratory graph analysis: a new approach for estimating the number of dimensions in psychological research. *PlosOne.*

- Wagenmakers, E-J., Marsman, M., Jamil T., Ly, A., Verhagen, J., Love, J., Selker, R., Gronau, Q.F., Smirna, M., **Epskamp, S.**, Matzke, D., Rouder, J.N., Morey, R.D. Bayesian Inference for Psychology. Part I: Theoretical Advantages and Practical Ramifications

    - Under review at *Psychonomic Bulletin & Review*

- Mõttus, R., **Epskamp, S.**, and Francis, A. (2016). Within-and between individual variability of personality characteristics and physical exercise. *Journal of Research in Personality.*

- Tio, P., **Epskamp, S.**, Noordhof, A., and Borsboom, D. (2016). Mapping the manuals of madness: Comparing the ICD-10 and DSM-IV-TR using a network approach. *International Journal of Methods in Psychiatric Research, 25 (4)*, 267-276.

- Fried, E. I., van Borkulo, C. D., **Epskamp, S.**, Schoevers, R. A., Tuerlinckx, F., and Borsboom, D. (2016). Measuring Depression Over Time. . . or not? Lack of Unidimensionality and Longitudinal Measurement Invariance in Four Common Rating Scales of Depression. *Psychological Assessment, 28 (11)*, 1354-1367.

- **Open Science Collaboration**. (2015). Estimating the reproducibility of psychological science. *Science, 349 (6251).*

    - I setup and coordinated the "analysis audit" in which a team of students worked on replicating all analyses done in R.

- Wichers, M., Groot, P.C., **Psychosystems**, ESM Group, EWS Group (2016). *Psychotherapy and Psychosomatics, 85*: 114-116.

- Nuijten, M.B., Hartgerink, C. H. J., van Assen, M. A. L. M., **Epskamp, S.**, and Wicherts, J. M. (2016). The prevalence of statistical reporting errors in psychology (1985-2013). *Behavior Research Methods, 48*: 1205.

- Fried, E. I., **Epskamp, S.**, Nesse, R.M., Tuerlinckx, F., and Borsboom, D. (2016). What are 'good' depression symptoms? Comparing the centrality of DSM and non-DSM symptoms of depression in a network analysis. *Journal of Affective Disorders, 189*: 314-320.

- Kossakowski, J. J., **Epskamp, S.**, Kieffer, J. M., van Borkulo, C. D., Rhemtulla, M., and Borsboom, D. (2015). The application of a network approach to health-related quality of life (HRQoL): Introducing a new method for assessing HRQOL in healthy adults and cancer patient. *Quality of Life Research, 0962-9343*, 1-12.

- Langley, D. J., Wijn, R., **Epskamp, S.**, and Van Bork, R. (2015). Should I Get That Jab? *ECIS 2015 Research-in-Progress Papers*, Paper 64.

- Wigman, J. T. W., Van Os, J., Borsboom, D., Wardenaar, K. J., **Epskamp, S.**, Klippel, A, MERGE, Viechtbauer, W., Myin-Germeys, I., and Wichers, M. (2015). Exploring the underlying structure of mental disorders: cross-diagnostic differences and similarities from a network perspective using both a top-down and a bottom-up approach. *Psychological Medicine, 45*, 2375-2387.

- Fried, E. I., Bockting, C., Arjadi, R., Borsboom, D., Amshoff, M., Cramer, A. O. J., **Epskamp, S.**, Tuerlinckx, F., Carr, D., and Stroebe, M. (2015). From loss to loneliness: The relationship between depressive symptoms and bereavement. *Journal of Abnormal Psychology, 124*, 256-65.

- van Borkulo, C. D., Borsboom, D., **Epskamp, S.**, Blanken, T. F., Boschloo, L., Schoevers, R. A., and Waldorp, L. J. (2014). A new method for constructing networks from binary data. *Scientific Reports, 4*, 5918.

- Borsboom, D., Cramer, A.O.J., Schmittmann, V.D., **Epskamp, S.** and Waldorp L.J. (2011) The Small World of Psychopathology. *PLoS ONE 6*, e27407.

- Borsboom, D., **Epskamp, S.**, Kievit, R.A., Cramer, A.O.J., and Schmittmann,V.D. (2011). Transdiagnostic networks. *Perspectives on Psychological Science, 6*, 610-614.

- Schmittmann, V.D., Cramer, A.O.J., Waldorp, L.J., **Epskamp, S.**, Kievit, R.A. and Borsboom, D. (2011). Deconstructing the construct: Anetwork perspective on psychological phenomena. *New Ideas in Psychology, 31*, 43-53.

B.2 Software

Stable R packages

- **qgraph**
  - Network drawing, construction and estimation and network-based data visualization (Link to CRAN repository)

- **semPlot**
  - Path diagrams and visual analysis of various SEM packages' output (Link to CRAN repository)

- **IsingSampler**

  - Sampling methods and distribution functions for the Ising model (Link to CRAN repository)

- **lisrelToR**

  - Import output from LISREL into R (Link to CRAN repository)

- **graphicalVAR**

  - Estimate temporal and contemporaneous effects on $N = 1$ longitudinal data (Link to CRAN repository)

- **mlVAR**

  - Multi-level vector autoregression (Link to CRAN repository)

- **bootnet**

  - General robustness tests and plots for network models (Link to CRAN repository)

- **elasticIsing**

  - Ising model estimation using elastic net regularization (Link to CRAN repository)

- **lvnet**

  - Latent variable network modeling (Link to CRAN repository)

Collaborations

- **IsingFit**

  - Fitting Ising models using the eLasso method (Link to CRAN repository)

- **statcheck**

  - Fitting Ising models using the eLasso method (Link to CRAN repository)

- **JASP**

  - A low fat alternative to SPSS, a delicious alternative to R. (Link to website)
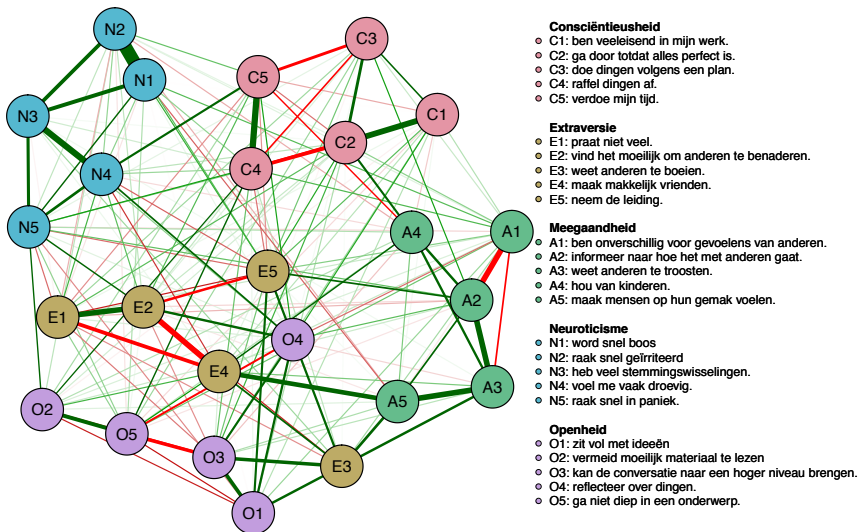
Appendix C

Nederlandse Samenvatting

C.1 Introductie: Psychologische netwerken

Dit proefschrift beschrijft het schatten van netwerkstructuren op psychologische data: *netwerkpsychometrie*. Dit onderzoeksveld is ontstaan uit een vraag om psychologische verschijnselen, zoals het gezamenlijk voorkomen van depressieve symptomen als moeheid en concentratieproblemen, niet te conceptualiseren als reflectief aan een onderliggende latente trek, zoals depressie, maar juist als een gevolg van directe interacties tussen de gemeten eigenschappen: moeheid leidt tot concentratieproblemen. De opkomst van bijvoorbeeld een depressieve episode kan dan gezien worden als emergent gedrag in een systeem van interacterende componenten. Om deze systemen in kaart te brengen wordt gebruik gemaakt van netwerkmodellen. Om verwarring te voorkomen met, bijvoorbeeld, sociale netwerken worden deze netwerken aangeduid als *psychologische netwerken*.

Figuur C.1 laat een voorbeeld zien van een dergelijk psychologisch netwerk. Dit netwerk is een ongericht netwerk waarin variabelen worden weergegeven als knopen. De verbinding tussen knopen geeft weer hoe sterk twee variabelen samenhangen nadat er is gecontroleerd op alle andere variabelen in de dataset. De introductie beschrijft hoe dergelijke verbindingen kunnen worden geïnterpreteerd; een verbinding geeft de unieke variantie weer tussen twee variabelen die niet verklaard kan worden door andere variabelen. Deze verbindingen kunnen gezien worden als indicatief voor mogelijke causale relaties tussen twee variabelen. Daartegenover: het ontbreken van een verbinding indiceert dat twee variabelen mogelijk conditioneel onafhankelijk zijn. De introductie beschrijft verder hoe methoden uit de grafentheorie gebruikt kunnen worden om de verkregen netwerkstructuren te analyseren. Zo kan de belangrijkheid van knopen bepaald worden met behulp van centraliteitsmaten.

De introductie van deze psychologische netwerken, en de in dit proefschrift beschreven software en technologische ontwikkelingen, hebben geleid tot een sterke navolging in (met name klinisch) psychologisch onderzoek. Deel 1 van dit proefschrift is derhalve gericht tot empirische onderzoekers in de psychologie, met een nadruk op klinische psychologie. In Deel 1 wordt het schatten van netwerkstructu-

Consciëntieusheid
- C1: ben veeleisend in mijn werk.
- C2: ga door totdat alles perfect is.
- C3: doe dingen volgens een plan.
- C4: raffel dingen af.
- C5: verdoe mijn tijd.

Extraversie
- E1: praat niet veel.
- E2: vind het moeilijk om anderen te benaderen.
- E3: weet anderen te boeien.
- E4: maak makkelijk vrienden.
- E5: neem de leiding.

Meegaandheid
- A1: ben onverschillig voor gevoelens van anderen.
- A2: informeer naar hoe het met anderen gaat.
- A3: weet anderen te troosten.
- A4: hou van kinderen.
- A5: maak mensen op hun gemak voelen.

Neuroticisme
- N1: word snel boos
- N2: raak snel geïrriteerd
- N3: heb veel stemmingswisselingen.
- N4: voel me vaak droevig.
- N5: raak snel in paniek.

Openheid
- O1: zit vol met ideeën
- O2: vermeid moeilijk materiaal te lezen
- O3: kan de conversatie naar een hoger niveau brengen.
- O4: reflecteer over dingen.
- O5: ga niet diep in een onderwerp.

Figuur C.1: Voorbeeld van een psychologisch netwerk. Knopen geven variabelen weer (in dit geval items van een persoonlijkheidsvragenlijst) en verbindingen tussen knopen geven partiële correlaties weer.

ren geïntroduceerd, en tevens een kritische noot gelegd om overinterpretatie tegen te gaan. Naast empirisch onderzoek bleken de netwerkmodellen ook een belangrijke bijdrage te kunnen leveren aan methodologisch en psychometrisch onderzoek; psychologische netwerken zijn niet alleen contrasterend ten opzichte van klassieke psychometrie, maar juist ook aanvullend. Een netwerkmodel is niet een tegenpool van het latente-variabelenmodel, maar kan juist nieuwe inzichten geven aan het latente-variabelenmodel. Derhalve richten Deel 2 en Deel 3 zich voornamelijk op de overeenkomsten tussen het netwerkmodel en de klassieke psychometrie. Zo blijkt dat het netwerkmodel als formeel psychometrisch model kan worden gevormd en dat netwerkmodellen en latente-variabelenmodellen zelfs equivalent kunnen zijn. Tevens biedt de netwerkpsychometrie sterke, nieuwe visualisatietechnieken voor psychometrisch onderzoek.

C.2 Deel I: Netwerkpsychometrie voor de empirische wetenschapper

### Hoofdstuk 2: Geregulariseerde netwerken van partiële correlaties

Dit hoofdstuk introduceert het meest gebruikte netwerkmodel voor psychologische netwerken: het *netwerk van partiële correlaties* (later in het proefschrift ook

een Gaussisch grafisch model genoemd). Dit is een netwerk waarin geobserveerde variabelen worden weergegeven als knopen. Verbindingen tussen knopen zijn gebaseerd op partiële correlaties tussen twee variabelen na conditioneren op alle andere variabelen. Deze netwerken moeten worden geschat op basis van data. Bij het schatten van een netwerk moet ook gekeken worden naar de *structuur* van het netwerk: welke knopen zijn met elkaar verbonden? Dit betekent dat het schatten van een netwerk gepaard gaat met een modelselectieprobleem. Doordat een netwerk een hoog-dimensioneel model is, is zowel het schatten van parameters als het uitvoeren van modelselectie niet triviaal. Er moet rekening gehouden worden met het feit dat de parameters niet 'overfitten' (te veel gebaseerd zijn op ruis) en dat de modelzoekruimte uitzonderlijk groot kan zijn. Een veel gebruikte oplossing is om gebruik te maken van een statistische techniek genaamd *regularisatie*. In regularisatie wordt doorgaans gebruik gemaakt van bestrafte grootste-aannemelijkheidsschatting (*penalized maximum likelihood estimation*) om overfitten tegen te gaan. Met name de variant genaamd de LASSO is veelbelovend voor het schatten van netwerken. De LASSO bestraft de som van absolute parameterwaardes, waardoor veel parameters in de schatting krimpen naar 0. Voor netwerken kan gebruik gemaakt worden van een variant van de LASSO genaamd de 'grafische LASSO' (glasso).

Om glasso te gebruiken moet eerst een schatting worden verkregen van de variantie–covariantie matrix. Wanneer de data continu zijn, kan deze worden verkregen door Pearson correlatiecoëfficiënten uit te rekenen. Voor ordinale data kan gebruik gemaakt worden van polychorische correlatiecoëfficiënten. Vervolgens berekent glasso een spaarzaam netwerk van partiële correlaties. De glasso berekent echter niet één netwerk, maar een *reeks* aan netwerken: van netwerken met veel verbindingen tot een netwerk met geen verbindingen. Om vervolgens een enkel netwerk te selecteren kan gebruik gemaakt worden van modelselectie. Met name het uitgebreide Bayesiaans informatie criterium (*extended Bayesian information criterion*; EBIC) werkt goed in het selecteren van de juiste netwerkstructuur. De EBIC maakt gebruik van een instelparameter, γ, die tussen 0 en 1 kan worden gezet. Bij $\gamma = 0$ prefereert de EBIC complexere modellen en bij $\gamma = 1$ prefereert de EBIC simpelere modellen. Gebruikelijk zijn 0.25 en 0.5 goede waardes voor deze instelparameter. Vervolgens wordt het netwerk met de laagste EBIC geselecteerd. Figuur C.2 geeft dit selectieproces grafisch weer. Deze methode is geïmplementeerd in het R pakket *qgraph*.[1]

Het hoofdstuk presenteert een lijst van mogelijke problemen die zich kunnen voordoen bij het schatten van netwerken van partiële correlaties. Met name wanneer polychorische correlatiecoëfficiënten worden gebruikt kunnen problemen ontstaan wanneer er te weinig data zijn (bijv. de correlatiematrix kan niet positief semi-definiet zijn). Het hoofdstuk concludeert met een simulatiestudie die laat zien dat de beschreven methode goed werkt. De schatting van een Ising model voor binaire variabelen en een gemengd grafisch model voor een mix van categorische en continue variabelen volgt ruwweg hetzelfde proces als het hier beschreven.

[1] https://github.com/SachaEpskamp/qgraph
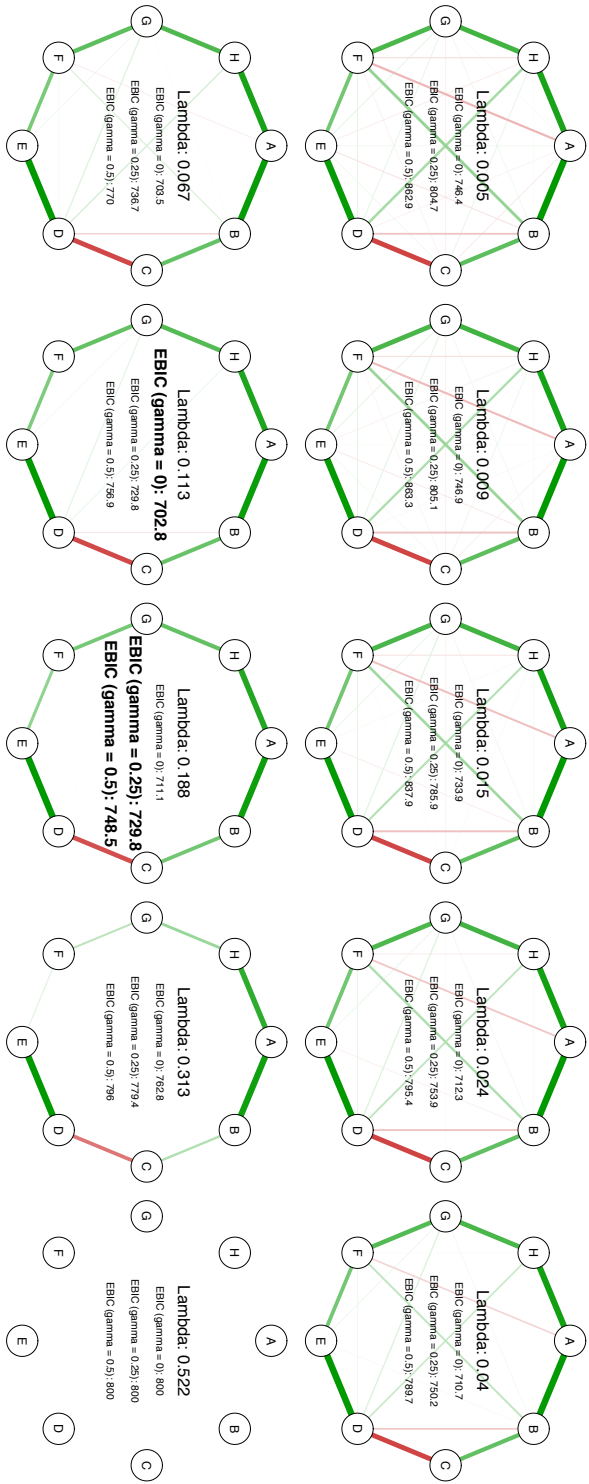
Figuur C.2: Tien verschillende geschatte netwerken van partiële correlaties met behulp van LASSO regularisatie. Een reeks van netwerken kan worden verkregen: van een netwerk met veel verbindingen tot een netwerk met geen verbindingen. Hoe goed elk model past wordt beoordeeld met de EBIC, waarbij de instelparameter γ op 0, 0.25 of 0.5 wordt gezet. De dik-gedrukte waarde is de laagste EBIC gegeven een bepaalde waarde van γ: dit netwerk wordt gekozen als beste model.

Hoofdstuk 3: Accuraatheid van psychologische netwerken

Dit hoofdstuk beargumenteert dat bij het analyseren van netwerken rekening moet worden gehouden met het feit dat deze zijn gebaseerd op een *steekproef*. Zoals beschreven in Hoofdstuk 2 moet een netwerk geschat worden op basis van data. Aangezien deze data een steekproef behelzen, zijn zelfs de beste schatters onderhevig aan ruis en niet perfect. Als hier geen rekening mee wordt gehouden, kan er een verkeerde conclusie worden getrokken op basis van een verkregen netwerkstructuur. Dit is onder meer belangrijk wanneer er gekeken wordt naar de *centraliteit* van knopen in het netwerk. Figuur C.3 laat een voorbeeld zien waarin knopen niet meer of minder centraal zijn van elkaar in het ware netwerk (Paneel A). In een netwerk dat geschat is op basis van een steekproef (Paneel C) verschillen knopen echter wel in centraliteit. Deze verschillen komen echter puur door ruis, en een substantiële interpretatie van centraliteit in dit netwerk zou niet gepast zijn. Het hoofdstuk beargumenteert dat door middel van *bootstrap* methoden inzicht kan worden verkregen in deze onzekerheid. In bootstrap methoden worden observaties (rijen in de dataset) willekeurig getrokken om nieuwe datasets te genereren. Op elk van deze datasets wordt vervolgens een netwerk geschat; de parameterwaardes van deze netwerken geven inzicht in de variabiliteit van de schatter. Deze methoden zijn geïmplementeerd in het R pakket *bootnet*.[2]

Het hoofdstuk beschrijft een driestappenplan om de mate van interpreteerbaarheid van gevonden verschillen te beoordelen. Deze stappen worden gepresenteerd in een afnemende mate van belangrijkheid. In stap A wordt de *non-parametrische bootstrap* (trekken van een zelfde aantal rijen als in de oorspronkelijke data met teruglegging) gebruikt om betrouwbaarheidsintervallen te schatten voor de netwerkverbindingen. Als deze betrouwbaarheidsintervallen groot zijn, is de sterkte van een verbinding in het netwerk moeilijk te interpreteren. Dergelijke betrouwbaarheidsintervallen kunnen echter niet gevormd worden voor centraliteitsmaten. In stap B wordt de interpreteerbaarheid van centraliteitsmaten gekwantificeerd door te kijken naar de stabiliteit van centraliteitsmaten met minder observaties. Dat wil zeggen, blijft de interpretatie van centraliteit hetzelfde op basis van een subset van de data. Om deze stabiliteit te bekijken, wordt gebruik gemaakt van de *subset bootstrap* (trekken van minder aantal rijen als in de oorspronkelijke data zonder teruglegging). Om dit te kwantificeren stelt het hoofdstuk de correlatiestabiliteit-coëfficiënt (CS-coëfficiënt) voor: $CS(\text{cor} = 0.7) = X$ betekent dat maximaal een proportie van X observaties uit de data kan worden gehaald om met 95% zekerheid te stellen dat de correlatie tussen de oorspronkelijke centraliteit en de centraliteit op basis van een willekeurige subset boven 0.7 blijft. Als deze maat onder 0.25 is, is de centraliteit niet te interpreteren. Idealiter is deze maat boven 0.5. In stap C kan een statistische test (*bootstrapped difference test*) worden uitgevoerd om te toetsen of verschillen tussen verbindingssterktes en centraliteit van knopen significant zijn. Gebruikmakend van deze nieuwe methoden blijkt dat de verschillen in centraliteit in Figuur C.3 zowel niet significant als niet stabiel zijn. Het hoofdstuk concludeert met stap-voor-stap instructies over het gebruik van *bootnet* en drie simulatiestudies over de CS-coëfficiënt en significantietoetsen.
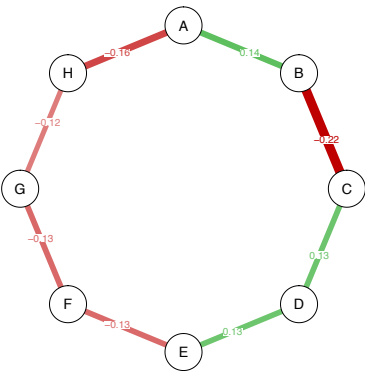

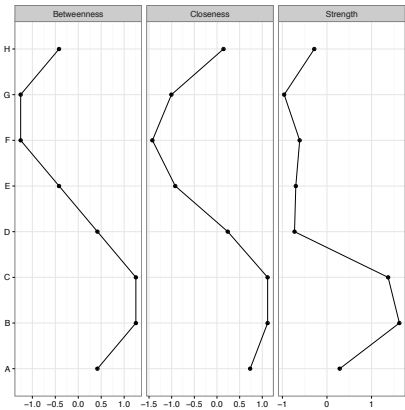
[2]https://github.com/SachaEpskamp/bootnet
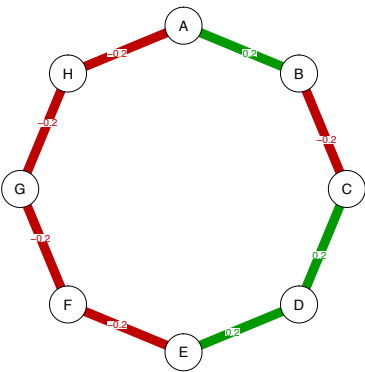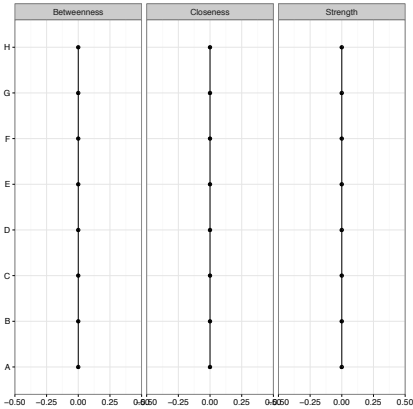
A. Ware netwerk

B. Ware centraliteit

C. Geschatte netwerk

D. Geschatte centraliteit

Figuur C.3: Paneel A laat een netwerk van partiële correlaties zien waaronder data zijn gesimuleerd. Paneel B laat de centraliteit zien (gestandaardiseerd als z-scores) volgens dit 'ware' netwerk. Het netwerk in Paneel A is geconstrueerd zodat geen enkele knoop meer of minder centraal is dan een andere knoop. In Paneel C is een netwerk te zien dat is geschat op gesimuleerde data ($N = 500$) onder het netwerk van Paneel A. Hoewel het geschatte netwerk van Paneel C goed lijkt op het ware netwerk van Paneel A is de schatting niet exact. Hierdoor zijn in het geschatte netwerk *wel* verschillen in centraliteit (Paneel D).

## Hoofdstuk 4: De schattingsmethode van netwerken en spaarzaamheid

Dit hoofdstuk waarschuwt tegen het overinterpreteren van geschatte netwerken op psychologische data. Aangezien netwerkmodellen zeer nauw verbonden zijn aan latente-variabelenmodellen (zie Hoofdstuk 7 en Hoofdstuk 8) betekent het schatten van een netwerkmodel niet dat het datagenererende mechanisme niet een latente-variabelenmodel had kunnen zijn. Een latente-variabelenmodel leidt tot een netwerk dat niet spaarzaam is: alle knopen zijn met elkaar verbonden. Echter, als wel een spaarzaam netwerk wordt gevonden (er ontbreken meerdere mogelijke verbindingen in het netwerk), bijvoorbeeld door LASSO schatting te gebruiken, is dat geen evidentie dat een latente-variabelenmodel niet aan de data ten grondslag kan liggen. Dit komt doordat de schattingsmethode bepaalde assumpties doet over de ware netwerkstructuur die niet kunnen worden genegeerd. De LASSO neemt aan dat het ware netwerk spaarzaam is, en zal zodoende vaak een spaarzaam netwerk schatten. Een andere methode is om een netwerk te schatten dat een lage rang heeft. Deze methode kan goed een netwerk terugschatten dat overeenkomt met een latente-variabelenmodel, maar heeft juist weer als keerzijde dat nooit een spaarzaam netwerk zal worden geschat.

Figuur C.4 laat een voorbeeld zien van het effect van de schattingsmethode. Data waren gegenereerd volgens het latente-variabelenmodel in Paneel A, een Multidimensionaal IRT model (MIRT). Dit model is een hypothetisch model gebaseerd op de symptomen van dysthemie en gegeneraliseerde angststoornis. Omdat deze symptomen deels overlappen (bijv. slapeloosheid is een symptoom van beide stoornissen) en deels elkaar uitsluiten (bijv. slapeloosheid en hypersomnie) zijn residuele factoren toegevoegd. De symptomen laden alle even sterk op de latente factoren, en zijn dus *uitwisselbaar*. Het MIRT model is sterk verbonden aan het Ising model (zie Hoofdstuk 7). Paneel B laat het Ising model zien dat zou verwacht worden gegeven het ware MIRT model (geschat op 10 miljoen gesimuleerde observaties). Paneel C laat een Ising model zien dat zonder regularisatie is geschat,[3] wat veel verbindingen oplevert (vooral negatieve) die niet overeenkomen met het ware model. Paneel D laat een Ising model zien geschat met LASSO regularisatie.[4] Dit netwerk is spaarzaam en levert te weinig verbindingen op. De clustering wordt niet volledig teruggeschat en knopen hebben een verschillend aantal verbindingen terwijl deze uitwisselbare symptomen representeren. Paneel E, ten slotte, laat een lage rang benadering zien die juist niet in staat is de brug-symptomen terug te schatten.

Zonder te weten dat het latente-variabelenmodel het ware model is, had een onderzoeker mogelijk alleen het spaarzame netwerk van Paneel D bekeken, en vervolgens geconcludeerd dat het ware model niet een latente-variabelenmodel kan zijn. Deze conclusie zou echter niet gegrond zijn, aangezien de LASSO zorgt voor een spaarzaam netwerk. Het onderscheiden van dergelijke modellen is nog een groot open vraagstuk in de netwerkpsychometrie, en kan mogelijk niet gedaan worden zonder experimentele manipulatie. Onderzoekers moeten daarom rekening

---

[3]Gebruikmakend van het R pakket *IsingSampler*; https://github.com/SachaEpskamp/IsingSampler

[4]Gebruikmakend van het R pakket *IsingFit*; https://github.com/cvborkulo/IsingFit
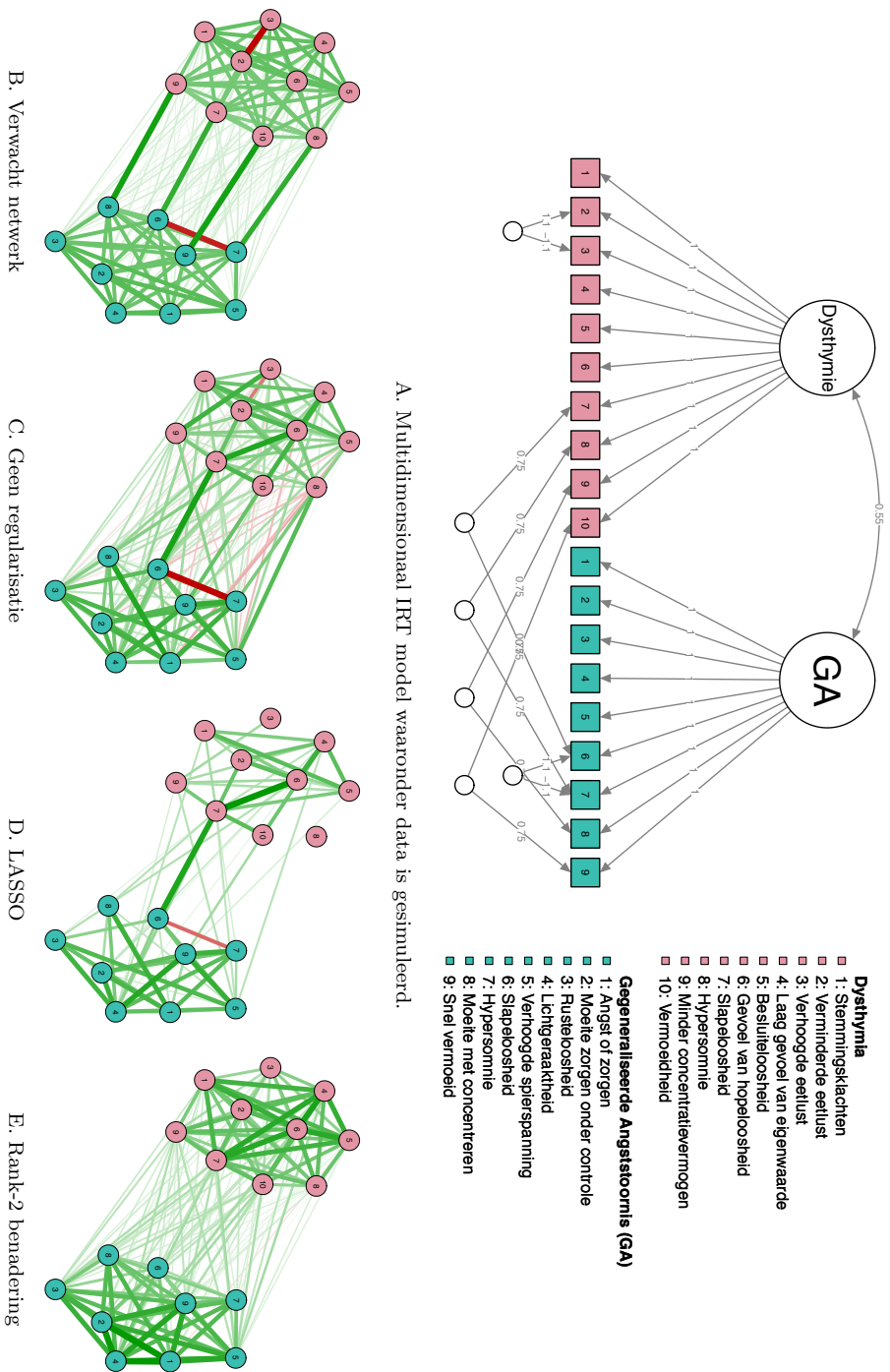
Figuur C.4: Verschillende geschatte netwerken gegeven dat het ware model een latente-variabelenmodel is (Paneel A).
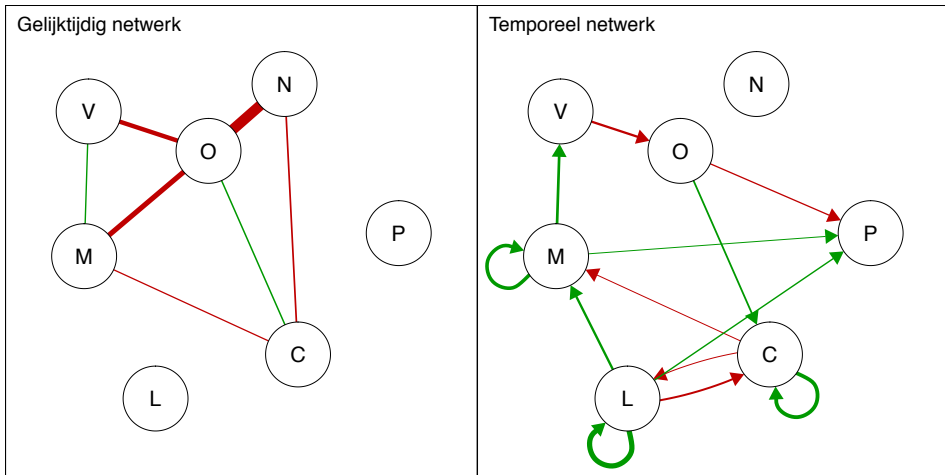
Figuur C.5: Temporeel (links) en gelijktijdig (rechts) netwerken gebaseerd op data van een klinische patiënt die 52 keer is gemeten over een periode van twee weken. Legenda van knopen: O = 'ontspannen'; V = 'verdrietig'; N = 'nerveus'; C = 'concentratie'; M = 'moe'; P = 'piekeren'; L = 'lichamelijk ongemak'.

houden met het feit dat de schattingsmethode de netwerkstructuur beïnvloedt, waardoor sommige conclusies niet getrokken kunnen worden.

Hoofdstuk 5: Gepersonaliseerde netwerkmodellen in de psychopathologie

Dit hoofdstuk beschrijft hoe netwerkmodellen gebruikt kunnen worden in de klinische praktijk. Een patiënt kan meerdere keren per dag worden gemeten over een periode van enkele weken. Vervolgens kunnen statistische methoden gebruikt worden om de dynamiek van deze patiënt in kaart te brengen. Het meest eenvoudige model dat kan worden gebruikt is het vector-autoregressieve model (*vector-autoregression*; VAR). In VAR wordt doorgaans een *temporeel netwerk* berekend, dat de samenhang tussen variabelen laat zien tussen opeenvolgende tijdsblokken. Een voorbeeld van een temporeel netwerk is te zien in het linker paneel van Figuur C.5. Een verbinding in een dergelijk temporeel netwerk indiceert dat een gemeten variabele in een bepaald tijdsblok (bijv. moe) een andere variabele in het volgende tijdsblok (bijv. verdrietig) voorspelt. Een dergelijke relatie kan voorkomen als iemands moeheid van invloed is op iemands gevoel van verdriet.

De VAR-analyse levert naast het temporele netwerk ook een tweede netwerk op: het *gelijktijdige netwerk*. Een voorbeeld van een dergelijk netwerk is te zien in het rechter paneel van Figuur C.5. Dit netwerk is een netwerk van partiële correlaties tussen de residuen van de VAR-analyse, en laat relaties zien die niet verklaard kunnen worden door de temporele verbanden. Relaties die sneller zijn dan de intensiteit van meten en zich afspelen in het zelfde tijdsblok komen terecht in dit gelijktijdige netwerk. Dit hoofdstuk beargumenteert dat het gelijktijdige netwerk

naast het temporele netwerk ook waardevolle informatie kan bevatten, omdat relaties in de psychopathologie plausibel snel kunnen optreden. Een voorbeeld is de relatie:

$$\text{hartkloppingen} \rightarrow \text{bang voor paniekaanval.}$$

Een persoon die lijdt onder een paniekstoornis kan bang worden dat hij of zij een paniekaanval krijgt omdat hij of zij hartkloppingen voelt. Een dergelijke relatie speelt zich waarschijnlijk zeer snel af en zal niet terecht komen in het temporele netwerk. Deze relatie zal echter mogelijk wel te zien zijn in het gelijktijdige netwerk; in tijdsblokken wanneer deze persoon hartkloppingen voelde was deze persoon mogelijk ook vaker bang voor een paniekaanval.

Het hoofdstuk beschrijft hoe een ongericht gelijktijdig netwerk, net als het gerichte temporele netwerk, indicatief kan zijn voor mogelijke relaties in een persoon, en zodoende als hypothese-genererend gebruikt kan worden in klinische praktijk. De methoden om een temporeel en gelijktijdig netwerk te schatten op een gering aantal observaties is onder andere geïmplementeerd in het R pakket *graphicalVAR*.[5] Het hoofdstuk beschrijft twee voorbeelden waarin deze software is gebruikt op data van patiënten, en beschrijft welke klinische inzichten verkregen kunnen worden uit een dergelijke analyse.

C.3 Deel II: Technologische ontwikkelingen in de netwerkpsychometrie

Hoofdstuk 6: Ontdekking van dynamische relaties in psychologische data

In dit hoofdstuk wordt het meest gebruikte netwerkmodel voor continue variabelen, het Gaussisch grafisch model (*Gaussian graphical model*; GGM) geïntroduceerd in de context van het ontdekken van dynamische relaties in psychologische data, wanneer kan worden aangenomen dat de data multivariabel normaal verdeeld zijn. Het GGM is een model voor de inverse van een variantie–covariantie matrix. Het hoofdstuk laat drie equivalenties zien tussen het GGM en andere statistische modellen. Ten eerste kunnen de elementen van de inverse variantie–covariantie matrix gestandaardiseerd worden tot *partiële correlatiecoëfficiënten*. Een netwerk wordt gebruikelijk getekend met partiële correlatiecoëfficiënten als verbindingen. Een partiële correlatie van nul betekent dat twee variabelen conditioneel onafhankelijk zijn en dus elkaar niet direct beïnvloeden. Een partiële correlatie die niet nul is – een verbinding in het netwerk – kan daardoor indicatief zijn van een mogelijk causale relatie. Ten tweede: deze partiële correlatiecoëfficiënten zijn proportioneel aan regressiecoëfficiënten verkregen uit de multipele regressie van één variabele op alle andere variabelen. Dit betekent dat de verbindingen in een GGM netwerk opgevat kunnen worden als predictief, en paden in het netwerken (bijvoorbeeld $A — B — C$) indicatief zijn voor predictieve mediatie. Zodoende brengt het GGM de multicollineariteit van een regressieanalyse in kaart. Ten derde heeft

---

[5] `https://github.com/SachaEpskamp/graphicalVAR`

het GGM dezelfde vorm als het Ising model, doordat beide lid zijn van een familie van modellen, genaamd Markov willekeurige velden (*Markov random fields*). Deze modellen worden in andere velden van de wetenschap veelvuldig gebruikt om complexe systemen te modelleren (zie hoofdstuk 8).

Wanneer de data niet transversaal zijn (*cross-sectional*; meerdere personen eenmaal gemeten) maar juist bestaan uit meerdere metingen van één of meerdere personen, wordt een algemene assumptie van de statistiek aannemelijk geschonden: opeenvolgende metingen zijn niet onafhankelijk. Zodoende kan het GGM niet zomaar berekend worden op dergelijke data. De tijdserie levert echter ook nieuwe informatie op: hoe variabelen zich tot elkaar verhouden over tijd. In tijdserie-data van een enkel persoon kan rekening gehouden worden met deze schending van onafhankelijkheid door te corrigeren voor het vorige meetmoment. Onder de aanname van multivariabele normaliteit heeft deze correctie de vorm van het vector-autoregressieve model (*vector-autoregression*: VAR). Dit model levert twee netwerken op: (1) een *temporeel netwerk*, een gericht netwerk waarin de voorspellende kracht van een variabele op een andere variabele over de tijd wordt weergegeven, en (2) een *gelijktijdig netwerk*, een GGM berekend op de residuen van de VAR analyse. Beargumenteerd wordt, zoals eerder in Hoofdstuk 5, dat beide netwerken relaties vertonen die interessant kunnen zijn voor de onderzoeker. Het temporele netwerk wordt al veelvuldig gebruikt in psychologisch onderzoek omdat het relaties weergeeft die zich over de tijd ontvouwen. Het gelijktijdige netwerk laat daarentegen relaties zien die zich afspelen op een tijdschaal die korter is dan de afstand tussen de metingen.

Wanneer tijdseriedata van meerdere personen beschikbaar zijn, kan er gebruik worden gemaakt van multiniveau (*multi-level*) VAR analyse. In deze analyse kan de grootte van individuele verschillen in kaart worden gebracht. Verder kan onderzocht worden welke relaties *in* een persoon plaatsvinden en welke relaties *tussen* personen plaatsvinden. Dit is een belangrijk verschil met resultaten die verkregen kunnen worden in transversale data. Een transversale relatie tussen bijvoorbeeld moeheid en concentratieproblemen kan voorkomen doordat mensen die over het algemeen moe zijn ook over het algemeen meer concentratieproblemen ervaren (een relatie tussen personen), of omdat wanneer een persoon buitensporig moe is deze persoon ook meer concentratieproblemen ervaart dan zijn of haar gemiddelde (een relatie in een persoon). De multiniveau VAR analyse levert zodoende naast een temporeel en gelijktijdig netwerk per persoon – en schattingen van de populatiegemiddelden: vaste effecten (*fixed effects*) – ook een GGM netwerk op dat de relaties beschrijft tussen de gemiddelde scores van personen: (3) een *tussen-persoons netwerk*. Beargumenteerd wordt dat dit tussen-persoons netwerk ook dynamische relaties kan weergeven.

De beschreven multiniveau VAR analyses zijn geïmplementeerd in het R pakket *mlVAR*.[6] Het hoofdstuk beschrijft simulatieresultaten die laten zien dat *mlVAR* goed de netwerkstructuren kan terugschatten. Figuur C.6 laat een voorbeeld zien van de drie netwerken die verkregen kunnen worden met multiniveau VAR. In deze netwerken zijn variabelen meegenomen die de persoonlijkheidstrek 'extraversie' meten en of een persoon inspanning heeft verricht. De resultaten laten zien dat

---

[6]https://github.com/SachaEpskamp/mlVAR
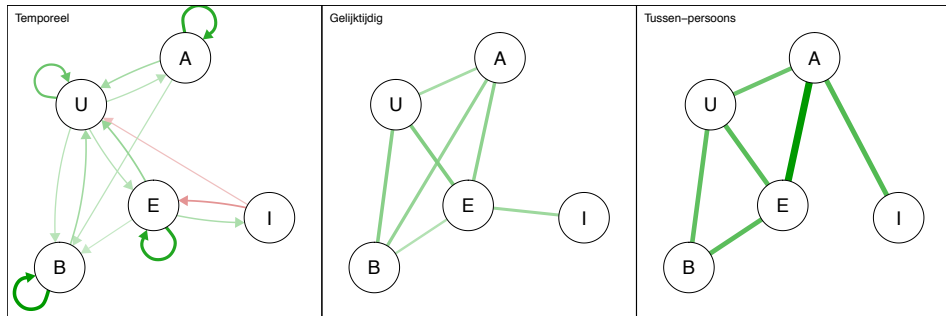
Figuur C.6: Geschatte gemiddelde netwerkstructuren die verkrijgbaar zijn in multiniveau vector-autoregressie. Het model is geschat op basis van 88 mensen die drie tot vijf keer per dag waren gemeten via een smartphone-applicatie; in totaal waren er 3516 observaties. Niet-significante verbindingen zijn verwijderd. Voor de ongerichte netwerken is gebruik gemaakt van de 'of'-regel voor significantie: een verbinding is behouden als één van de twee regressiecoëfficiënten waarop deze verbinding is gebaseerd significant was. Legenda van knopen: U = 'uitgaand'; E = 'energiek'; A = 'avontuurlijk'; B = 'blij'; I = 'inspanning'.

in hetzelfde tijdsblok dat een persoon inspanning verricht (bijv. sporten) deze persoon zich ook gemiddeld meer energiek voelt. In het tijdsblok nadat een persoon inspanning heeft verricht voelt deze persoon zich juist minder energiek. Verder is te zien dat mensen minder uitgaand waren na het verrichten van inspanning. In het tussen-persoons netwerk worden deze relaties niet gevonden. Het tussen-persoons netwerk laat echter een sterke verbinding zien tussen inspanning verrichten en zich avontuurlijk voelen: mensen die zich, over het algemeen, meer avontuurlijk voelen verrichten, over het algemeen, ook meer inspanning. Deze relatie was niet te zien in het temporele netwerk en het gelijktijdige netwerk.

Hoofdstuk 7: Gegeneraliseerde netwerkpsychometrie

In dit hoofdstuk wordt het GGM geïntroduceerd als een formeel psychometrisch model. In een netwerkmodel (Paneel A van Figuur C.7) wordt de samenhang van variabelen gezien als het gevolg van paarsgewijze interacties direct tussen de geobserveerde variabelen. Dit staat in sterk contrast met standaard psychometrie, waarin de samenhang van variabelen juist verklaard wordt door één of meerdere niet geobserveerde (latente) variabelen (Paneel B van Figuur C.7). Dit hoofdstuk beschrijft de relatie tussen het GGM en de meest gebruikte methode voor latente variabelen onderliggend aan continue data: structurele vergelijkingsmodellen (*Structural Equation Modeling*; SEM). Het hoofdstuk laat zien dat netwerkmodelleren de andere kant is van dezelfde munt: waar in SEM de variantie–covariantie wordt gemodelleerd, wordt in het GGM juist de inverse van de variantie–covariantie matrix gemodelleerd. Hierdoor is het GGM direct te implementeren in SEM, en kan van een GGM model passingsmaten worden verkregen of een GGM model vergeleken worden met een SEM model. Door de combinatie

A. Structurele vergelijkingsmodellen

B. Netwerkmodellen

C. Latente network modellen

D. Residuele netwerkmodellen
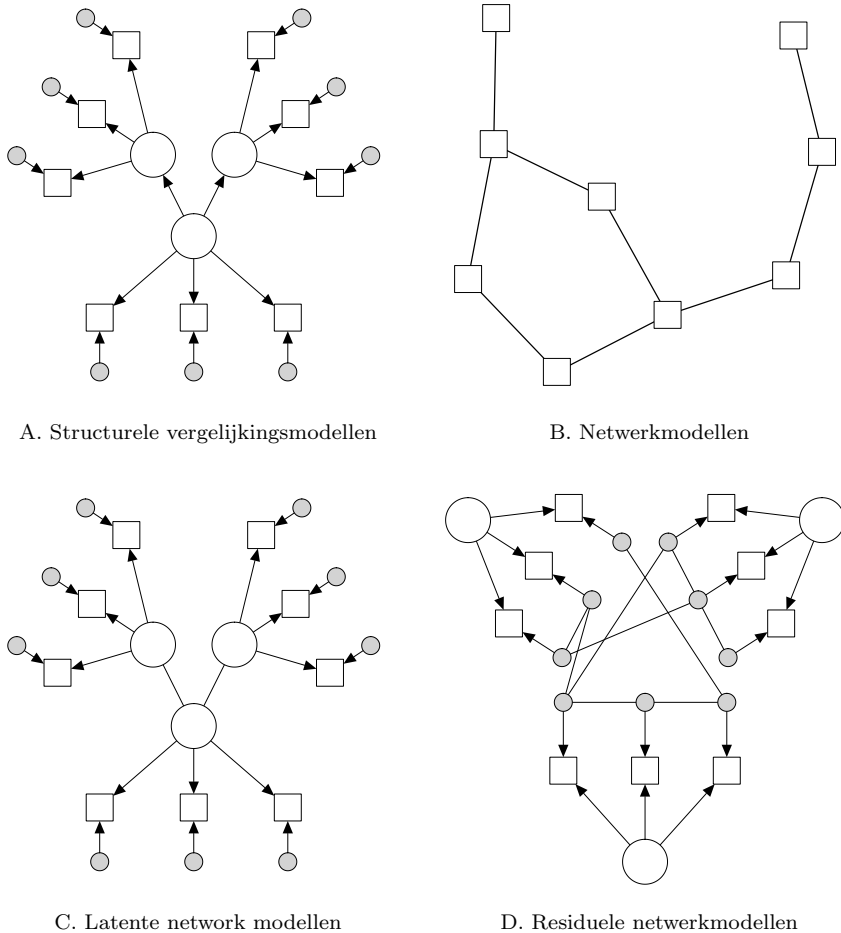
Figuur C.7: Voorbeelden van verschillende mogelijke modellen onder de vier beschreven modelspecificaties. Ronde knopen laten latente variabelen zien, vierkante knopen geobserveerde variabelen en grijze knopen de residuen. Gerichte verbindingen representeren factor-ladingen of regressieparameters en ongerichte verbindingen representeren paarsgewijze interacties. Deze verbindingen zijn *niet* marginale covarianties, maar juist *conditionele*.

van SEM en het GGM ontstaan ook twee nieuwe modellen: latent netwerkmodelleren (LNM) en residueel netwerkmodelleren (RNM).

Figuur C.7 laat voorbeelden van de twee nieuwe modelspecificaties zien. In Paneel C is een voorbeeld van het LNM model te zien waarin negen geobserveerde variabelen worden verklaard door drie latente variabelen. Op het latente niveau is een ongericht netwerk gemodelleerd in plaats van een gericht netwerk zoals typisch in SEM. Dit netwerk encodeert dat twee latente variabelen onafhankelijk zijn na conditioneren op een derde. Het model in Paneel C is statistisch equivalent aan

het SEM model in Paneel A en heeft geen equivalente modellen (zonder meer latente variabelen toe te voegen). Doordat LNM dergelijke conditionele onafhankelijkheden kan ontdekken, is deze modelspecificatie veelbelovend in exploratief onderzoek naar relaties tussen latente variabelen. Paneel D laat een RNM model zien: een SEM model met een GGM netwerk tussen de residuen. In RNM worden conditionele covarianties gemodelleerd in plaats van marginale covarianties. Hierdoor kan dit model op een spaarzame manier (positief aantal vrijheidsgraden) een residuele variantie–covariantie matrix modelleren waarin geen enkel element nul is. Zodoende wordt in het model van Paneel D geen lokale onafhankelijkheid aangenomen (alle residuen kunnen correleren met elkaar). Deze modelspecificatie is veelbelovend om een confirmatief factor-model passend te maken zonder dat kruisladingen gebruikt moeten worden.

Het hoofdstuk concludeert door te laten zien dat met een bestrafte grootste-aannemelijkheidsschatter, de LASSO, beide modellen kunnen worden geschat. Deze schattingsmethode volgt grofweg de modelselectiemethode beschreven in Hoofdstuk 2. Het gebruik van deze schattingsmethode wordt onderbouwd met twee simulatiestudies. Deze simulatiestudies laten zien dat de methode in combinatie met EBIC modelselectie de ware netwerkstructuur goed kan terugschatten, mits er genoeg observaties zijn. De methoden om deze modellen te schatten zijn geïmplementeerd in het R pakket *lvnet*.[7]

Hoofdstuk 8: Het Ising model in de psychometrie

Dit hoofdstuk introduceert netwerkmodellen voor psychometrici, met een bijzondere nadruk op het netwerkmodel voor binaire data: het Ising model. Het Ising model wordt eerst geïntroduceerd vanuit de statistische natuurkunde, als een model voor magnetisme van deeltjes. Vervolgens laat het hoofdstuk zien dat het Ising model nauw verbonden is aan drie gebruikelijke modellen in de psychometrie: (1) het conditionele Ising model is equivalent aan een logistisch regressiemodel, (2) het Ising model is equivalent aan een loglinear model met hoogstens paarsgewijze interacties, en (3) het Ising model is equivalent aan een bepaald soort latente-variabelenmodel: het multidimensionale item respons theorie (MIRT) model met een conditionele normale verdeling voor de latente trekken.

Met name de laatste equivalentie wordt uitvoerig besproken. Een Ising model is equivalent aan een bepaald soort MIRT model; de rang van de netwerk-matrix komt overeen met het aantal latente variabelen. Zoals eerder beschreven in Hoofdstuk 4 heeft deze equivalentie grote gevolgen voor de interpretatie van netwerkmodellen. Voor elk Ising model bestaat een equivalent latente-variabelenmodel en vice versa. De equivalentie heeft ook gevolgen voor de psychometrie, aangezien het een nieuwe karakterisatie mogelijk maakt van de gezamenlijke verdeling van data gegeven het MIRT model. Het hoofdstuk beschrijft verder hoe het Ising model geschat kan worden en breidt uit over het onderscheiden van netwerkmodellen en latente-variabelenmodellen. Het hoofdstuk beschrijft ten slotte een voorbeeld van geschatte Ising modellen gebruikmakend van het R pakket *elasticIsing*.[8]

---

[7]https://github.com/SachaEpskamp/lvnet
[8]https://github.com/SachaEpskamp/elasticIsing

C.4 Deel III: Visualisaties in de psychometrie en persoonlijkheidsonderzoek

Hoofdstuk 9: Visualisering van psychometrische relaties

Dit hoofdstuk introduceert het *qgraph*[9] pakket voor R in de context van data-visualisatie in de psychometrie. Het *qgraph* pakket kan gebruikt worden voor het weergeven en analyseren van gewogen netwerken, en is de methode die gebruikt is voor de meeste visualisaties in dit proefschrift. Naast de eerder beschreven netwerkmodellen kan *qgraph* gebruikt worden om een netwerk weer te geven van marginale correlatiecoëfficiënten. Op deze manier kan een hoog-dimensionale correlatiematrix op een nieuwe manier worden weergegeven. Deze visualisatiemethode kan inzicht geven in de clustering van variabelen, alsmede in de algemene structuur van correlationele patronen in de data. Tevens kan met deze visualisatiemethode inzicht gekregen worden in hoe goed een SEM model wel of niet past op de data. Daarnaast kan *qgraph* ook gebruikt worden om andere statistische relaties weer te geven, zoals factor- en componentladingen. Figuur C.8 laat een grafische weergave zien van geschatte componentladingen in een principale-componenten-analyse. Een belangrijk voordeel van deze methode voor het weergeven van factor- of componentladingen is dat er direct inzicht wordt gekregen in zowel de sterkste ladingen als de belangrijkste kruisladingen.

Hoofdstuk 10: De netwerkbenadering in persoonlijkheidsonderzoek

Dit hoofdstuk beschrijft stap voor stap hoe netwerkmodellen in persoonlijkheids-onderzoek kunnen worden gebruikt. Hierbij wordt gebruik gemaakt van regulari-satie om een netwerk van partiële correlaties te schatten.[10] Vervolgens beschrijft dit hoofdstuk in detail hoe het verkregen netwerk kan worden geanalyseerd met behulp van centraliteitsmaten, clusteringcoëfficiënten en algemene netwerkeigen-schappen. De methode wordt verduidelijkt door data te analyseren van de veel-gebruikte HEXACO vragenlijst.

Hoofdstuk 11: Geünificeerde visualisaties van structurele vergelijkingsmodellen

Dit hoofdstuk introduceert het R pakket *semPlot*[11], waarmee pad diagrammen kunnen worden gegenereerd op basis van structurele vergelijkingsmodellen (*structural equation modeling*; SEM). Output van verschillende softwarepakketten (zoals *lavaan*, *sem*, *Mplus* en *LISREL*) kunnen worden ingelezen om zodoende een pad diagram te tekenen. Het pakket biedt meer functionaliteit dan alleen pad diagram-men weergeven, en kan tevens gebruikt worden als een brug tussen verschillende

---

[9]https://github.com/SachaEpskamp/qgraph

[10]De beschreven methode is vergelijkbaar met de methode die is beschreven in Hoofdstuk 2, die nog niet was geïmplementeerd ten tijde van het schrijven van Hoofdstuk 10

[11]https://github.com/SachaEpskamp/semPlot
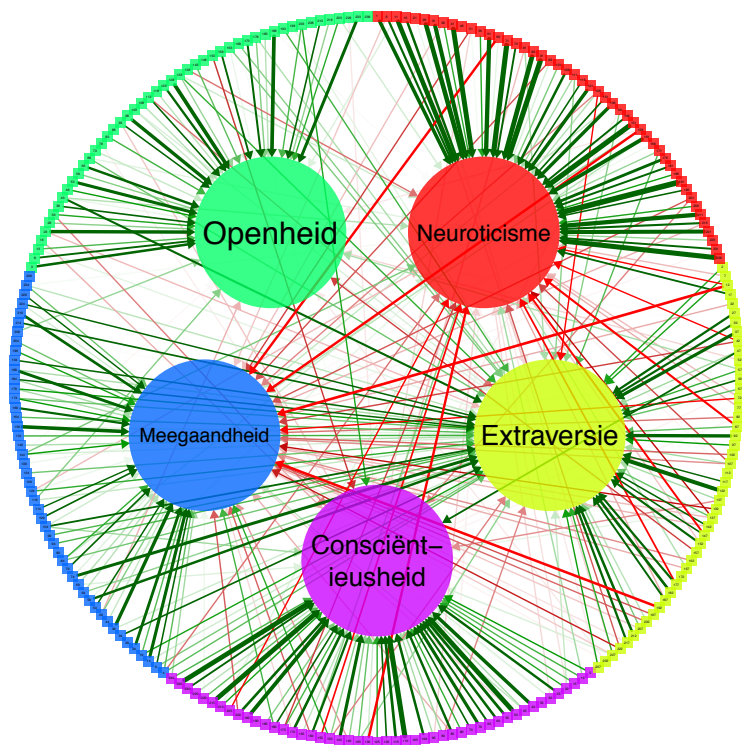
Figuur C.8: Visualisatie gemaakt met het *qgraph* pakket van de resultaten van een principale componentanalyse.

softwarepakketten en modelspecificaties. Input voor bijvoorbeeld *lavaan* kan gegenereerd worden op basis van output van bijvoorbeeld *Mplus*. Tevens kunnen de modelmatrices verkregen worden en gebruikt worden voor verdere berekeningen. Het hoofdstuk concludeert met een gedetailleerd overzicht van verschillende algoritmes om de knopen van het pad-diagram te plaatsen.

C.5 Discussie: open vraagstukken in de netwerkpsychometrie

In de tijd van dit promotietraject is de netwerkpsychometrie uitgegroeid tot een veld waarin meer en meer talentvolle onderzoekers hun intrede doen. Hoewel het gebruik van netwerken in de psychologie steeds beter begrepen is en steeds

meer methoden worden ontwikkeld, zijn er ook nog veel open vragen. Derhalve concludeert dit proefschrift met enkele richtingen voor vervolgonderzoek:

1. Er zijn op dit moment weinig manieren om om te gaan met ontbrekende data in het schatten van psychologische netwerken. Meestal worden alle observaties waarin minstens één response ontbreekt volledig weggehaald. Hierdoor worden echter niet alle data meegenomen in de analyse en wordt er een sterke aanname gedaan over de oorzaak waarom een response ontbreekt. De psychometrie kent een lange geschiedenis van omgaan met ontbrekende data, waar de literatuur van het schatten van netwerkmodellen veel van kan leren.

2. Veel data in de psychologie zijn gemeten op een ordinale schaal. Bijvoorbeeld, veel vragenlijsten in de psychopathologie meten de ernst van symptomen op een schaal van 0 (geen klachten) tot 3 (heel veel klachten). De psychometrie kent ook hier een lange geschiedenis van het modelleren van dergelijke data. Dit soort data komt echter niet voor in de velden waarin netwerkmodellen doorgaans worden gebruikt. Derhalve is het gebruik van ordinale data in het schatten van netwerkmodellen nog niet goed uitgewerkt. De doorgaans gebruikte methodes – het gebruik van polychorische correlaties voor het berekenen van een netwerk van partiële correlaties of het discreet maken van antwoorden om vervolgens een Ising model te berekenen – zijn niet zonder problemen. Het is dus van cruciaal belang dat het modelleren van ordinale data in netwerkmodellen verder wordt ontwikkeld.

3. Zoals beschreven in Hoofdstuk 4 is het momenteel niet mogelijk evidentie te vinden dat het ware netwerk spaarzaam is (niet alle knopen zijn verbonden). Dit terwijl het ontbreken van verbindingen een belangrijke interpretatie met zich mee brengt: conditionele onafhankelijkheid. Derhalve is het van belang dat methoden worden ontwikkeld die wel evidentie kunnen vergaren voor het ontbreken van verbindingen. Veelbelovend hierin is het gebruik van Bayesiaanse statistiek, waarin de Bayes-factor een steeds meer gebruikte maat is die evidentie voor een null-hypothese kan vergaren. Toekomstig onderzoek zou zich kunnen richten op het ontwikkelen van standaard Bayesiaanse testen voor netwerkverbindingen.

4. Vaak worden de verkregen netwerkstructuren geanalyseerd op een vergelijkbare manier als bijvoorbeeld wegen-netwerken of sociale netwerken. De modellen die voor dit proefschrift zijn gebruikt, zijn echter structureel anders. De knopen representeren variabelen met meerdere staten en de verbindingen representeren statistische relaties. Deze modellen worden *probabilistische grafische modellen* genoemd en encoderen de volledige gezamenlijke verdeling van een set variabelen. Het is de vraag of dergelijke modellen wel moeten worden geanalyseerd met maten uit de grafentheorie. De discussie laat zien dat met gebruik van informatietheorie op een andere manier gekeken kan worden naar de verkregen resultaten. Meer onderzoek is nodig om deze maten te valideren.

5. De netwerkmodellen die zijn gepresenteerd in dit proefschrift modelleren alleen de samenhang van variabelen. Er wordt in deze netwerken niet gekeken

naar de drempelwaarde van een variabele (bijv. dat het symptoom 'suïcidale gedachten' minder vaak voorkomt dan het symptoom 'droevige stemming'). Deze eigenschap wordt gemodelleerd met een intercept. Doordat niet gekeken wordt naar intercepten kan het zijn dat twee variabelen aan elkaar worden verbonden zonder dat deze vergelijkbaar zijn in drempelwaarde. Dit kan problematisch zijn. Mogelijk kan vervolgonderzoek kijken naar het construeren van netwerken waarbij ook het intercept een rol speelt.

Het proefschrift concludeert met de notie dat de beschreven netwerkmodellen, hoewel veelbelovend, lang niet de enige methode vormen om om te gaan met de meer algemenere hypothese dat psychologie complex is. De toekomst zal uitwijzen welke plek deze methoden in psychologisch onderzoek zullen innemen.

Acknowledgements — Dankwoord

Beste Denny, dit dankwoord kan niet anders beginnen dan een woord aan jou. Zonder jouw inspirerende colleges in het eerste jaar was ik misschien wel nooit geïnteresseerd geraakt in methodologie. Bedankt voor de unieke kans in mijn tweede jaar om voor je te werken. Ik realiseerde me jaren later pas hoe uitzonderlijk deze kans was. Ik had me geen betere studie kunnen voorstellen dan in jouw lab mee te draaien voor het begin van mijn promotie, en ik had tijdens mijn promotie geen betere begeleider kunnen bedenken. Jij hebt me keer op keer laten zien wat een ware wetenschapper hoort te doen. Ik ben heel blij om 7.5 jaar deze ontdekkingsreis samen met jou te hebben kunnen maken en kijk uit naar onze verdere samenwerkingen. Lourens, bedankt voor de vele dingen die ik van jou heb geleerd. Veel ideeën in dit proefschrift komen direct of indirect van jou, en daarnaast bedankt dat ik altijd met vragen terecht kon.

Mam, bedankt voor dat je me geleerd hebt om werk zonder zuchten en maren te doen voor een deadline, je onverslagen geloof in mij, de Reiki en iets te veel tempels en Boeddas. Pap, bedankt voor dat ik altijd bij je terecht kan, dat je er altijd voor me bent, heerlijk eetentjes, wijze levenslessen en eindeloze ritten door de Outback. Bro, bedankt voor dat je me de sportschool in sleep, de lol die ik altijd met je , dat je al twee decennia keurig boven mijn Moo Moo Farm record blijft, dat ik je mocht railen bij poker en hearthstone toernooien en dat je me introduceerde in de wondere wereld van het show worstelen. Ik kan me geen betere broer voorstellen. Amy, bedankt voor de duizende schattige foto's die ik langs zag komen de laatste jaren, ik kijk uit naar je te zien opgroeien. Oma, bedankt voor dat je dat je de meest coole oma bent die een persoon kan hebben. Verdere familie, bedankt voor de blues, bierkennis, klus hulp en nog veel meer!

Renze, ik ken je al meer dan 20 jaar en had dat voelt nog steeds te kort. Ik kan me geen betere vriend voorstellen. Bedankt voor dat ik altijd bij je terecht kan, en het vertrouwen dat ik dat altijd zal kunnen. Ik kijk uit naar de komende 20 jaar als vrienden, en naar je te zien als vader. Dear meisje Adela, thank you for standing by me during the time leading up to finishing this dissertation, joining me countless times working in the Coffee Company, eating the food I make, drinking (ginger) beers and Sambuca with me and crushing me at pool and other games.

I look forward to seeing you continue being a totally awesome rock-star scientist and person in the coming years!

I especially want to thank my dear Paranymphs. Froomy Sanja, thank you for starting this PhD off the right way by putting a roof over my head and a jacuzzi under my feet. Thank you for your friendship and your honesty, which make up for the crippling vanilla latte addiction you made me develop. I look forward to more tile-hunting adventures. Alexander, bedankt voor onze wekelijkse therapie sessies op de cross-trainer, voor het organiseren van bijzondere feesten en koningsdagen en voor de vele levenslessen. Gracia, thank you for being the latent puppet-master paranymph and all the amazing curry, chilli crab, laksa and spring rolls!

There are so many wonderful people I have met throughout the years that I can hardly list them all. Boris en Cesar, bedankt voor dat jullie de meest coole huisgenoten waren. Ceren, thank you for the postcards, coffee breaks and beers. Jonathon, thank you for the Friday drinks and being such a wonderful and open person. Ravi, bedankt voor de inspirerende Torino's pizzas. Colleagues from the Pyschosystems group, Claudia, Riet, Jolanda, Eiko, Pia, Lisa, Tessa, Joost, Marie and both Jonasses, thank you for the amazing atmosphere in which we keep discovering new things. Mijke and Angelique, who in my view still belong to the previous list, thank you for all the collaboration and many chats. We all miss you a lot! Abe and Alexander, bedankt voor dat jullie hele coole kamergenoten waren tijdens deze promotie. Han, bedankt dat je de afdeling zo een warme plek maakt om te werken. Eric-Jan, Johnny, Maarten, Helen, Dora, Quentin, Frans and Bruno, thank you for adopting me in your lab from time to time, the drinks and bittergarnituren. Andere medewerkers aan de UvA, inclusief Dylan, Gunter, Harrie, Lousie, Maria, Claire, Raoul en Ineke, bedankt voor dat jullie de afdeling zo leuk maken. Medewerkers van Oefenweb, onder andere Marthe, Daniel, Marin en Vera, bedankt voor dat ik een klein deel mocht uitmaken van de opgroei van jullie fantastische bedrijf. Harriette, Date, Michelle en Jos, bedankt voor onze leerzame Skype-afspraken en leuke bezoeken aan Groningen. Bedankt IOPS en haar fantastische studenten en staff leden—waaronder Michele, Mattis, Tanja, Robert, Laura, Noemi en vele anderen—voor talloze methodologische discussies, enerverende conferenties en iets te veel biertjes. Also thanks to the wonderful people I have met during my travels, such as Emilanio, Sorina, Weng Kin, Suzanne, Vivek and Bee Wee, as well as the inspiring people I have gotten to interact or work with, among who but not limited to Yves Rosseel, Richard McNally, Don Robinaugh, Rene Mottus, Tom Booth, Ellen Hamaker, Peter Sloot and Rick Quax. Finally, thanks to Maien, Thomas, Max, Ale, Sjoerd, Charlotte, Mattis, Anja, Jason, and all other friends I fail to mention.

This dissertation ends as it started, in memory of the two friends whose lives were cut much too short. Dear Zara, I will always be shocked by your sudden departure from this world. I regret not having got to know you better than I did and spend more time with you. You were an amazingly colorful person and the world is much poorer without you. Lieve Janneke, toen deze promotie begon stond een ding al vast: jij zou hoe dan ook mijn paranimf zijn. Ik had niets liever gewild dat jij op deze dag naast me had kunnen staan. Bedankt voor je vriendschap, onze gesprekken en je goede adviezen, je kooktips, je eerlijkheid en nog veel meer dan ik hier kan noemen. Ik mis je.