Modeling Moderators in Psychological Networks

Trevor James Swanson

B.A. Psychology, Texas Christian University, 2014

M.A. Psychology, University of Kansas 2014

Submitted to the graduate degree program in Department of Psychology and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Mark Landau, Chairperson

Michael Vitevitch

Committee members

Tim Pleskac

Nancy Hamilton

Ben Sherwood

Date defended: June 5, 2020

The Dissertation Committee for Trevor James Swanson certifies
that this is the approved version of the following dissertation :

Modeling Moderators in Psychological Networks

Mark Landau, Chairperson

Date approved: June 5, 2020

Abstract

It is an important goal for psychologists to develop and improve upon methods for describing multivariate relationships among observed variables. Psychological network models represent one class of methods for studying such relationships, and are being applied widely throughout psychological science. While these models have been shown to have a variety of diverse applications, they are limited by the fact that they currently only consider pairwise relationships among sets of variables. Specifically, they don't take into consideration more complex relational structures such as those characterized by moderation effects and higher-order interactions. Moderation analysis, which focuses on these types of effects, is a common technique used within psychological research to help reveal the *contexts* and *conditions* under which different relationships may emerge or be observed. Thus, the goal of this research is to extend the psychological network framework to include moderator variables, as well as provide statistical tools and software to facilitate testing such models with psychological data.

Acknowledgements

Thank you to my loving parents, family, and friends. I am forever grateful for your unwavering encouragement and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, network representations of complex systems have been applied widely throughout many scientific disciplines, contributing statistical tools designed to aid us in understanding intricate causal structures that drive the behavior of multivariate processes (c.f. Borgatti et al., 2009; Barabási et al., 2016). For instance, network models—i.e., *probabilistic graphical models* (Lauritzen, 1996)—have been used by psychologists in studying such diverse phenomena as the dynamics of psychopathology (e.g., Cramer et al., 2010; Epskamp et al., 2018b), personality development across the lifespan (e.g., Read et al., 2010; Costantini et al., 2019), the associative structure of words in memory (e.g., Hills et al., 2010; De Deyne & Storms, 2008), the role of probabilistic phonotactics in spoken word recognition (e.g., Vitevitch & Luce, 2016), and connections between social networks and personal well-being (e.g., Kindermann & Snell, 1980; Galaskiewicz & Wasserman, 1993). Researchers working within these domains, among others, are often interested in characterizing the underlying structure that relates variables within a complex system, as well as the dynamic, interdependent connections that make up important links between psychological, sociological, and behavioral variables as they interact and change over time.

Although they have been applied to a wide array of topics, the general goal of psychological network models is to characterize the unique associations among variables that are theorized to be interrelated with regard to some psychological process or construct. These models have been applied within both cross-sectional and longitudinal contexts, where in the former the objective is often to describe which variables have conditional relationships across individuals, and in the latter—i.e., *temporal networks*—the goal is often to describe the evolution and development of those relationships within a single individual over time. Multilevel implementations of longitudinal

models have also been employed in the study of whether those within-person processes hold across groups of individuals (Epskamp et al., 2018c,b).

While psychological networks have been shown to have a variety of diverse applications, they are limited by the fact that at present they are only used to consider pairwise relationships among sets of variables. Specifically, they don't take into consideration more complex relational structures such as those characterized by moderation effects and higher-order interactions. Moderation analysis, which focuses on these types of effects, is a common technique used within psychological research to help reveal the *contexts* and *conditions* under which different relationships emerge or are observed between variables. For instance, we may be interested in whether the relationship between anxiety and insomnia is dependent on the degree to which an individual is experiencing situational stress. It is well-known that moderators can be crucial for understanding conditional interdependencies between variables, and that failing to investigate such variables when they are present can lead to a distorted understanding of the underlying phenomenon. Thus, my goal for this research is to extend the psychological network framework to include moderator variables, as well as to provide both statistical and computational tools that facilitate testing such models with psychological data.

1.1 Overview

In this paper, I introduce a basic framework for constructing *moderated network models* (MNMs), as well as showcase how these have been implemented in a software package I developed for R called `modnets`. Overall, the goals of this project were to: (1) Develop a framework for constructing and estimating MNMs in both cross-sectional and longitudinal data; (2) create open-source software that allows researchers to apply these models in a variety of settings; and (3) present flexible options for model selection and other analyses of MNMs, including methods for model comparison and post-hoc stability assessments.

In the current chapter, I begin by describing different types of psychological network models, as well as how they are commonly interpreted and applied in the literature. I then provide some

background on moderated regression analysis, as well as how we can integrate this method into the network framework. Chapter 2 covers the statistical foundations of MNMs, along with a discussion of general analysis procedures, and presents a cross-sectional example by applying these to empirical data. In Chapter 3 I discuss model selection techniques for MNMs, which differ from methods commonly used with psychological networks as they must adapted to incorporate some additional considerations. Then, in Chapter 4 I present a large-scale simulation study aimed at evaluating the performance of MNMs across different contexts, as well as to determine how they stack up against alternative approaches used in the literature. Chapter 5 changes the focus to temporal MNMs, presenting the differences between these and the cross-sectional models through another simulation study accompanied by an application to empirical data. Finally, Chapter 6 provides a brief discussion of some limitations and future directions regarding the development of these models, and concludes with a summary of the contributions of this framework to psychologists' methodological toolkit.

1.2 Psychological Networks

It is an important goal for psychologists to develop and improve upon methods for describing complex relationships among observed variables. The dominant approach toward this end has long been *latent variable modeling*, where the objective is to connect a set of observations with some unobserved latent construct(s) using methods such as confirmatory factor analysis and structural equation modeling (e.g., Little et al., 2006). In general, these methods aim at highlighting the shared variance among variables while downplaying their conditional relationships. The *network perspective* in psychology, however, offers a powerful counterpoint to this approach by framing psychological constructs as systems of interacting variables rather than manifestations of unobserved factors. Thus, while latent variable models are designed to sift through observed variation in search of underlying commonalities, network models focus on the direct associations between variables in an effort to identify their possible causal structure. For instance, rather than conceptualizing cognitive abilities, behavioral dispositions, and mood disorders as indicators of un-

derlying common causes—such as general intelligence, personality traits, and psychopathology—psychological networks frame these phenomena as emergent characteristics of networks of interrelated components.

The primary goal behind the network perspective is to characterize the causal structure of psychological and behavioral systems using concepts from graph theory, a field of study that was developed to describe information processing in computer networks and transportation systems (West, 1996). Network models provide both statistical and theoretical representations of psychological constructs, wherein a set of variables (such as mood states, attitudes, symptoms, etc.) can be represented as components, or 'nodes', whose unique associations are depicted as links, or 'edges'. One domain in which these models have been applied is the *network approach to psychopathology*, where mental disorders are viewed as arising from networks of directly interacting and mutually reinforcing symptoms (Cramer et al., 2010; Fried et al., 2017; Bringmann et al., 2013; Cramer et al., 2016). From this perspective, symptoms of mental disorders such as Major Depression (MD) co-occur because of their direct causal relationships, rather than their connections to a latent disease entity. For example, instead of treating symptoms such as insomnia, fatigue, and dysphoria merely as consequences of a neurochemical imbalance or brain disorder, the network approach to psychopathology considers distinct causal pathways such as insomnia $\rightarrow$ fatigue $\rightarrow$ dysphoria as key contributors to the experience of MD.

1.2.1 Interpreting Network Models

Despite being built on methods from graph theory, psychological networks are often constructed in ways that are vastly different from traditional graph-theoretic models (Epskamp et al., 2018a). Specifically, the structure of these networks is typically unknown, rather than observed, and so must be estimated from data. This leads to questions about the best way to construct such models, especially with regards to defining the nature of the edges—i.e., associations between variables. Any type of statistical association can be used, including marginal associations (such as zero-order correlations; e.g., Forbush et al., 2016; Siew et al., 2019), however the most common way to define

edges is in terms of *conditional* associations, or partial correlations between variables (Epskamp & Fried, 2018b; Williams et al., 2019). This is the standard approach for estimating probabilistic graphical models (Lauritzen, 1996), and has been widely utilized in psychological research.

1.2.1.1 Conditional associations

The utility of representing edges with partial correlations lies in the fact that this allows us to understand network models as encoding *conditional dependencies* between nodes. That is, when an edge is drawn between nodes A and B in a given network, we can interpret this as indicating that the variables associated with A and B share unique variation after taking into account all other nodes in the network. Conversely, when no edge is drawn between two nodes, this indicates that the corresponding variables are *conditionally independent*, meaning that they do not have any direct association after conditioning on their connections with other nodes in the network. These interpretations are useful in the sense that they help to reveal the core structure of a network in terms of the direct effects between any given pair of variables (van Borkulo et al., 2015; Haslbeck & Fried, 2017).

Given that one of the primary objectives of network modeling is to identify causal relationships between variables, estimating conditional (in)dependencies has the potential of revealing the "causal skeleton" of the network, even when estimated from observational data (McNally et al., 2015; Boschloo et al., 2016). That is, when two variables are determined to be conditionally independent given the remaining nodes in the network, it makes it highly unlikely that they have a direct causal relationship. Thus, researchers often set out to model a *sparse* network structure, wherein the fewest number of edges as possible are estimated. Techniques such as significance thresholding, model selection, and regularization are common approaches to estimating a sparse network structure, though each of these comes with their own potential drawbacks (Epskamp et al., 2018a).

Lastly, when the presence of an edge is interpreted as encoding the conditional dependence between two nodes, this preserves the possibility that the two corresponding variables do indeed have

a causal relationship. This is certainly not guaranteed when the network has been estimated from observational, cross-sectional data, but it nevertheless provides an exploratory way for researchers to begin generating causal hypotheses and perhaps conducting experiments to assess particular aspects of a given network's structure.

1.2.1.2 Types of networks

There are two primary types of network models: (a) undirected, and (b) directed networks. *Undirected networks* are the most common type used in psychological research, as these are estimated from cross-sectional data where the causal relationships between variables cannot be directly known. Thus, the edges in these networks are simply represented as lines connecting nodes, indicating that there are no directed causal associations that define the relationships between them. These are the types of model where partial correlations are most commonly employed, as they make up the vast majority of investigations within the psychological network literature. Importantly, inferences about causality are not possible for these types of models (see Ryan et al., 2019), although researchers often use them for exploratory purposes and as hypothesis-generating tools. In any event, determining whether some set of nodes are conditionally independent can help to narrow down the possibilities of what the true causal structure of a network might be.

In contrast with undirected network models, *directed networks* explicitly encode directed associations in the sense that edges are represented with one-sided arrows to signify that one variable is the predictor of another. These networks are frequently estimated from time-series or repeated-measures data, and are often interpreted as temporal networks, where edges are identified as regression coefficients with the direction of the arrow representing the relationship from the predictor to the outcome. I will discuss these models in greater detail in Chapter 5, and for now restrict our focus on cross-sectional models to provide the basic foundations of what will later be extended into the temporal context.

1.2.2 Extending Network Models

In all of the cases described above, one important aspect of network models is that they only encode *pairwise relationships* between variables, and do not take into account how some relationships may vary in accordance with changes in other variables. An implicit assumption of these models is that pairwise relationships between variables will be the same across all values of both variables (as well as across others). Additionally, no software currently exists (or at least is not readily available) for psychologists to test higher-order interactions in network models of the type described in this paper.

Although in some circumstances it may be a reasonable assumption that the relationships between variables are restricted to pairwise associations, researchers have explicitly called for the need to test alternative models in past work (e.g., Fried & Cramer, 2017). Moreover, this assumption seems particularly unreasonable in the case of temporal network models, wherein the associations between variables are estimated over time, while the influences of important situational and contextual variables (e.g., situational stress, environmental factors) may be overlooked entirely due to the lack of an available framework with which to test for moderator and interaction effects. Thus, an important goal for extending the framework of network modeling is to afford researchers the ability to test *interaction hypotheses* as well as use exploratory techniques to investigate multivariate causal structures in psychological data. In the next section, I will provide a more detailed background on moderation effects as well as motivate reasons why it is important to investigate these types of relationships in psychological networks.

1.3 Basics of Moderation

In the most basic sense, moderation implies that the relationship between two variables depends upon values of a third variable. This third variable is referred to as a *moderator*, in that it modifies "the direction and/or strength of the relation between an independent or predictor variable and a dependent or criterion variable" (Baron & Kenny, 1986, p.1174). Moderators can be either qualitative

(e.g., gender) or quantitative (e.g., situational stress) variables, and so are interpreted differently depending upon the context. For instance, researchers examining a qualitative moderator may be interested in *subgroup effects*, where a predictor has an inconsistent relationship to an outcome for different types of people (Hall & Sammons, 2013), or perhaps only has influence within specific subpopulations (Brambor et al., 2006). A quantitative moderator may instead be seen as an *effect modifier* (Hinshaw, 2002), where its variation is expected to correspond with variation in the degree of some observed relationship. Thus, moderation is central to evaluating *conditional hypotheses*, or when a researcher is interested in investigating the conditions under which, or for whom, a purported cause produces some expected effect(s).

Testing conditional hypotheses is ubiquitous in psychological research, as it can afford a more detailed understanding of both how and when different independent and dependent variables relate (Baron & Kenny, 1986). Perhaps it is theorized that a particular medication will only be effective for people with specific neurological characteristics, or that emotional reactions to negative feedback will be stronger among individuals who are higher in certain personality traits. In cases such as these, explicitly testing for the presence of moderation is crucial to obtaining a complete picture of the phenomenon at hand.

From a statistical standpoint, moderation effects are typically investigated via the inclusion of *multiplicative interaction terms* in an analysis of variance (ANOVA) or multiple regression model. For example, imagine that we wish to determine whether or not the effect of some independent variable X on some dependent variable Y depends on the value of some third variable Z. In the context of regression, we can assess this possibility by estimating

$$Y = \beta_0 + \beta_1 X + \beta_2 Z + \beta_3 XZ + \varepsilon, \tag{1.1}$$

where the multiplicative term XZ and its corresponding coefficient $\hat{\beta}_3$ serve to represent the moderation effect under investigation. We can then assess the significance of this effect by employing standard hypothesis tests such as t-tests or F-tests (depending upon the context), and thereby de-

termine whether or not there is evidence of moderation. That is, if we find that the interaction term XZ has a significant relationship with Y via our evaluation of $\hat{\beta}_3$, we would conclude the effect of X on Y is indeed dependent on levels of Z.

1.3.1 Interpreting Multiplicative Terms

The first reported use of the term "moderation" was by Saunders (1955), in which the term was employed as a synonym for what is commonly called an "interaction effect". However, although testing multiplicative terms is still the standard approach for assessing moderation effects, it is important to note that many authors make a conceptual distinction between the two constructs. An *interaction effect*, for instance, is taken to be a more general description of the conditional relationships between an outcome and some set of interacting predictors, while a *moderation effect* refers to a more particular type of causal relationship (Hall & Sammons, 2013). Specifically, moderation requires that the researcher identify which of the two constituent terms is the relevant causal variable (X, in the example here) that directly affects Y, as well as which term is the modifier of their relationship (here, Z). In sum, moderation requires that we identify one variable as the *explanatory* variable, and the other as the moderator of its effect on the outcome. Still, this distinction is purely theoretical. It simply guides how we approach analyzing and interpreting the results of a multiplicative interaction model.

From a mathematical perspective, both types of effect are assessed through the same means: the statistical significance of a multiplicative term within a regression or ANOVA model. The interaction term itself is agnostic about which variable is which—it simply provides empirical evidence as to whether the nonlinear product of the two variables reveals significant variation in each variable's slope over the range of observed values for the other. The causally-agnostic interpretation of a multiplicative term would then be to simply characterize it as an interaction effect, wherein the researcher does not (or cannot) specify which constituent variable is in fact the moderator, and which is the independent variable. This is especially relevant in observational research, where none of the measured variables are experimentally manipulated. Yet, even in such

cases it may be possible to identify one variable as the moderator based on theory—for instance, an individual's social context, personality traits, or average number of stressful events encountered in a day may reasonably be interpreted as moderators even in the absence of experimental design. Thus, it is up to the researcher to determine which interpretation of the multiplicative interaction term and its constituents best applies in a given situation.

Either way we choose to interpret the roles of these two variables, however, we are still left with the task of interpreting the model parameters. In standard regression models without interaction terms we interpret the coefficient estimate of a given predictor as the expected change in Y given a unit change in that variable, assuming all other variables are held constant. This is referred to as the *unconditional marginal effect* of that predictor on the outcome Y (Berry et al., 2016; Brambor et al., 2006; Braumoeller, 2004). These effects are 'unconditional' in the sense that they remain constant regardless of the values of all other predictors. This is only true for additive models, wherein no multiplicative interaction terms are included. But whenever interaction terms are included, as in equation 1.1, this interpretation no longer holds. In these situations, the marginal effect of a predictor that is also part of an interaction term now depends on values of the other variable(s) it interacts with. Continuing with the example in equation 1.1, we must now interpret X as having *conditional marginal effects* on Y, meaning that its contribution to the variance of Y is dependent upon values of Z.

1.3.2 Conditional Marginal Effects

The key to interpreting slope coefficients in moderated regression models is to consider the conditional marginal effects (or simply 'conditional effects') of the independent variable on the dependent variable across substantively meaningful values of the moderator (Wright Jr, 1976; Friedrich, 1982). In the case of a moderation hypotheses, we not only expect that Y depends on values of both X and Z, but that the effect of X on Y is itself dependent upon values of Z. And while the converse of this will also be true (i.e., that the slope relating Z to Y is dependent upon X), with moderation analysis we are typically only interested in how the explanatory variable affects the

outcome, as well as how their relationship varies across levels of the moderator. Nevertheless, everything presented here about interpreting conditional effects applies equally to both variables that are constitutive of an interaction effect.

To start thinking about the conditional effects of X on Y across values of Z, we can re-write equation 1.1 as follows:

$$\begin{aligned} Y &= \beta_0 + \beta_1 X + \beta_2 Z + \beta_3 XZ + \varepsilon \\ &= \beta_0 + (\beta_1 + \beta_3 Z)X + \beta_2 Z + \varepsilon. \end{aligned} \tag{1.2}$$

Upon rearranging the terms of our model, we can more clearly see how to interpret the conditional marginal effects of X on Y. Considering regression models in general, the marginal effect of a predictor on the outcome is equal to the partial derivative of that predictor with respect to the outcome, where $\frac{\partial Y}{\partial X} = \beta_1$ would be the marginal effect of X on Y in an additive model that *excludes* the interaction term XZ (Brambor et al., 2006; Friedrich, 1982). In the present case, however, the marginal effect will instead be equal to

$$\frac{\partial Y}{\partial X} = \beta_1 + \beta_3 Z, \tag{1.3}$$

which, as can be seen in equation 1.2, is simply the quantity that we multiply with X to obtain its marginal effect on Y. This equation clearly shows that even when we hold Z constant, the effect of X on Y directly depends on its value (in addition to the estimate $\hat{\beta}_3$). Additionally, we can see that when $Z = 0$, the conditional effect of X on Y is equal to β_1. This is the meaning of a 'main effect' in a regression model that includes an interaction: the 'main effect' or 'simple slope' of a predictor that is part of an interaction is *not* the 'average effect' of that variable across values of the moderator, as it may sometimes be perceived (c.f. Friedrich, 1982), but is rather the effect of that variable only when the moderator is equal to 0.

Importantly, this value is still a *conditional* marginal effect. That is, in this example $\hat{\beta}_1$ is the estimated marginal effect of X on Y conditional on $Z = 0$. At any other value of Z, the marginal

effect of X will be equal to $\hat{\beta}_1 + \hat{\beta}_3 Z$. The reasoning here applies equally to the standard error of the estimate: the standard error associated with $\hat{\beta}_1$, as returned by any standard statistical software, is also conditional on $Z = 0$. Just as the marginal effect of X on Y varies over the range of Z, so too does its standard error. We can obtain the standard errors associated with the conditional effects by computing

$$\hat{\sigma}_{\frac{\partial Y}{\partial X}} = \sqrt{var(\hat{\beta}_1) + Z^2 var(\hat{\beta}_3) + 2Zcov(\hat{\beta}_1, \hat{\beta}_3)} \tag{1.4}$$

across values of Z.

1.3.3 Visualizing Moderation Effects

With the capacity to obtain both conditional slopes (Eq. 1.3) and their associated standard errors (Eq. 1.4) for values of Z, we can now clearly visualize the nature of a given interaction effect, as well as conduct hypothesis tests at meaningful values of the moderator (e.g., $+/-1$ *SD* around the mean) to determine the conditions under which the independent variable has a significant relationship with the outcome. Relatedly, the availability of standard errors affords us the ability to plot confidence intervals along with the estimated marginal effects conditioned on values of Z. The choice of which values of the moderator to estimate conditional effects for will be up to the researcher and question at hand. However, with a quantitative moderator it is common to simply plot conditional effects across the full range of observed values (Hainmueller et al., 2019).

1.3.3.1 (Conditional) marginal effects plot

To provide an example of how to visualize conditional marginal effects, I've simulated data with the same structure as the present example. Specifically, two variables $\{X, Z\}$ with 500 observations were sampled from $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ and used to construct the outcome

$$Y = 0.5 + 0.6X - 0.4Z + 0.25XZ + \varepsilon, \quad \text{where} \quad \varepsilon \sim \mathcal{N}(0, 1). \tag{1.5}$$

Regressing Y onto $\{X, Z, XZ\}$ and estimating the marginal effects of X on Y across the observed range of Z produced the following conditional effects plot.

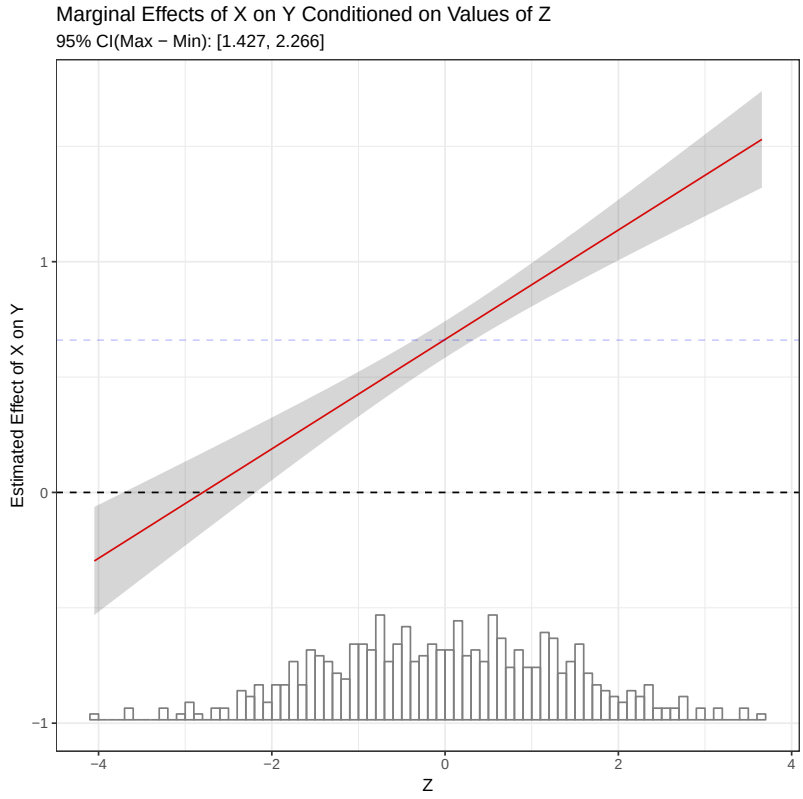

Figure 1.1: Conditional marginal effects of X on Y across values of Z. The red line represents the estimated marginal effects, with the gray bands being 95% confidence intervals. The faint blue line at $Y = .66$ shows the estimated marginal effect when the interaction term is not included in the model. The 95%CI at the top of the plot reflects the coverage interval of the interaction effect.

Figure 1.1 helps us gain a more comprehensive understanding of how Z moderates the effect of X on Y. First, as per equations 1.1 and 1.2, the unstandardized estimate of the interaction effect is equal to the slope of the line observed in the plot ($\hat{\beta}_3 = 0.24$). Additionally, the marginal effect conditioned on $Z = 0$ is equal to the unstandardized main effect of X on Y ($\hat{\beta}_1 = 0.66$). The gray bands surrounding the marginal effects represent 95% confidence intervals based on the standard errors estimated from equation 1.4. We can see that the intervals do not include 0 over most the range of Z, and reveal that X is expected to have a significant *negative* effect on Y at low values of Z, but a significant *positive* effect across the majority of Z's range. Moreover, the histogram at the bottom of the plot displays the distribution of Z, and the 95% CI at the top represents the

coverage interval of the difference between effects estimated at its maximum and minimum using data simulated from the posterior distribution of $(\beta_1 + \beta_3 Z)$ with the `sim` function from the `arm` package in R (Gelman & Su, 2018).

1.3.3.2 Discussion

The purpose of this example is to demonstrate how a researcher might go about probing an interaction effect by examining the conditional effects of an independent variable across the range of a continuous moderator. Importantly, this plot illustrates the necessity of testing interaction effects when one is investigating a conditional hypothesis. Although this is a simulated example, we can clearly see that without modeling the interaction between X and Z, one's understanding of the relationship between X and Y would be greatly misrepresented. Had the interaction effect not been included, the estimated marginal effect of X on Y would be captured by the dashed blue line where the y-axis equals 0.66. This is the main effect of X on Y in the non-interaction model, where Z is still included as a covariate. The bottom line is that visualizing the character of how two variables interact is crucial for understanding how they might relate with an outcome of interest. Especially for multivariate models such as psychological networks, failing to include relevant interaction terms can lead to misspecified models that improperly characterize the relationships among a set of variables. This plotting function is included as part of the `modnets` package for assessing interaction effects within psychological network models; thus, it is a tool that researchers studying moderated networks in psychology will be able to use as part of their analyses and profitably employ to explore the nature of interaction effects in their models.

1.3.4 Summary

The goal of the present section was to provide a background describing moderation analysis, interaction effects, and how rich information about multivariate relationships can be gained from studying the conditional effects of a given predictor on an outcome after considering values of a moderator. The central point here was to showcase a simple way of extracting this type of informa-

tion from a moderated regression model to better understand the nature of the relationships among variables.

In the next section, I will present my approach to analyzing *moderated network models*, which are essentially a multivariate generalization of the basic template presented here. Combined with techniques from graphical modeling, adding interaction terms to network models contributes a potentially powerful framework for investigating more complex models and more specific hypotheses (namely, those dealing with moderation and interaction effects). Furthermore, generalizing the concept of plotting conditional effects across values of a moderator, I will present a new contribution to psychological network modeling which I term *conditional networks*. As in the case of plotting marginal effects at specific values of a moderator, I will show how we can construct networks after conditioning on specific values of a moderator. This offers a way of gaining added information about moderated networks and visualizing the results in easily interpretable plots. I present the basic ideas behind moderated networks in the next section.

Chapter 2

Moderated Network Models

The goal of this section is to provide an overview of Moderated Network Models (MNMs) along with some examples of how they can be used to analyze psychological data. I limit the present discussion to *cross-sectional* MNMs, where the objective is to estimate an undirected network from a sample of $n > 1$ subjects measured on p variables at a single time point. Temporal formulations of these models (i.e., MNMs applied to time series data; both in idiographic and multi-subject settings) will be examined in a separate paper.

Currently, the MNM framework I present here and implement in the `modnets` package supports the analysis of both continuous and binary variables, where each type can serve as either a predictor or an outcome. The simplest case, however, is based on the Gaussian Graphical Model (GGM), wherein all outcomes (i.e., nodes in the network) are assumed to be continuous variables with a multivariate normal distribution. An undirected network consisting of all binary variables is commonly referred to as an *Ising Model*, and has been studied at great length in previous work (see Lauritzen, 1996; van Borkulo et al., 2014; Marsman et al., 2018). Moreover, Mixed Graphical Models—where outcome variables may be associated with different univariate distributions—have also received treatment within the literature (e.g., Yang et al., 2014; Chen et al., 2014). Importantly, all of these models can be formulated as MNMs using the same basic approach, although the results will be subject to different interpretations and are not necessarily associated with a normalizable joint distribution (e.g., Yang et al., 2014; Haslbeck et al., 2018). In such cases, this precludes us from computing the likelihood of the model when taken as a whole, thereby preventing us from performing global goodness-of-fit analyses along with model comparison tests. Thus, to outline the basic foundation of MNMs here, I focus on the scenario where all outcome variables are assumed

to be continuous, Gaussian variables, as these have the most straightforward interpretation along with a normalizable joint distribution.

I distinguish between two different types of MNMs: (a) the *exogenous* MNM, and (b) the *endogenous* MNM. The qualifier in each case is meant to refer to the nature of the moderator variable(s) included in the model. For an *exogenous* MNM, we essentially treat the moderator variable(s) as covariates in the model; that is, they are not treated as outcomes which are affected by the other variables, and therefore are not represented as nodes in the resulting network. Instead, they are treated as external components of the process or system under investigation. Identification of moderators as exogenous variables should be based on theory, or based on a reasonable assumption that the variable in question is not influenced by the other variables under study (such as ethnicity, experimental condition, gender, etc.). With *endogenous* moderators, however, these variables will be identified as nodes in the network and treated as outcomes in the construction of the model. This latter case will be most useful in exploratory settings, and will be discussed in a separate paper. Thus, in the remainder of this section I will present the foundation for estimating and interpreting MNMs wherein all outcomes are Gaussian variables and there is only one moderator which is treated as an exogenous variable.

2.1 Estimation

The standard way of constructing a GGM is through joint estimation of model parameters via the standardized inverse covariance matrix—i.e., the *partial correlation matrix*. Let $\mathbf{X}$ be an $n \times p$ matrix, where each column represents a different variable $(j = 1, \ldots, p)$, and each row contains the response vector for a single subject $(i = 1, \ldots, n)$. For the standard GGM, we assume that the response vectors are multivariate normal:

$$\mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \; \forall i \in \{1, \ldots, n\}, \tag{2.1}$$

where $\boldsymbol{\mu}$ is the $p \times 1$ vector of means, and $\boldsymbol{\Sigma}$ is the $p \times p$ variance-covariance matrix. For simplicity, we will also assume that all the variables have been mean-centered, such that $\boldsymbol{\mu} = \mathbf{0}$. Thus, our only objective is to estimate the partial correlation matrix $\boldsymbol{\Omega}$, which is a $p \times p$ symmetric matrix with zeros in the diagonal and partial correlations between each pair of variables in the off-diagonal. This can be obtained by taking the maximum likelihood estimate (MLE) of $\boldsymbol{\Sigma}$ and standardizing its inverse (i.e., the precision matrix), such that

$$\hat{\boldsymbol{\Sigma}} = \frac{\mathbf{X}^{\mathsf{T}}\mathbf{X}}{n}, \quad \text{and} \quad \hat{\boldsymbol{\Omega}} = \mathbf{I} - \boldsymbol{\Delta}\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{\Delta}, \tag{2.2}$$

where $\mathbf{I}$ is a $p \times p$ identity matrix, and $\boldsymbol{\Delta}$ is a diagonal scaling matrix with zeros in the off-diagonal and $\text{diag}(\boldsymbol{\Delta}) = \text{diag}(\hat{\boldsymbol{\Sigma}}^{-1})^{-\frac{1}{2}}$. The resulting estimate of $\boldsymbol{\Omega}$ therefore encodes the conditional dependence structure of the data, and is used to define the undirected network.

When moderators, covariates, or any higher-order interaction terms are included in the model, however, this approach is no longer possible. Instead, we can use an approach called *nodewise regression* (Haslbeck & Waldorp, 2015; Epskamp et al., 2018c). This method is based on a graph-theoretic approach to structure learning called neighborhood selection (Meinshausen & Bühlmann, 2006), and entails estimating the network structure via a series of univariate regression models. That is, a separate regression model is constructed for each variable represented as a node in the network, and then the coefficients are aggregated across models to define the final network structure.

For example, imagine that we wanted to model the conditional dependence of three random variables: X_1, X_2, and X_3. Taking the nodewise regression approach, we would begin by constructing three separate univariate regressions:

$$\begin{aligned} X_1 &= \beta_{10} + \beta_{12}X_2 + \beta_{13}X_3 + \varepsilon_1 \\ X_2 &= \beta_{20} + \beta_{21}X_1 + \beta_{23}X_3 + \varepsilon_2 \\ X_3 &= \beta_{30} + \beta_{31}X_1 + \beta_{32}X_2 + \varepsilon_3. \end{aligned} \tag{2.3}$$

Given that the slope parameters represent directed conditional relationships—e.g., in this example, β_{12} is the slope predicting X_1 from X_2 after conditioning on X_3—we can combine relevant pairs of coefficients to obtain the *undirected* conditional dependencies (i.e., partial correlations) between variables. For instance, here the partial correlation between X_1 and X_2 can be defined as $\omega_{1,2} = \text{sign}(\beta_{12})\sqrt{\beta_{12}\beta_{21}}$. It is important to note that the signs of β_{12} and β_{21} will always be the same if and only if the two slope parameters are conditioned on the same covariates (e.g., X_3 serves as a covariate in both models), and $n > p$ (Williams et al., 2019).

While this equivalence holds when exogenous covariates are included as predictors in the nodewise regressions, it no longer holds when interaction terms are included. That is, imagine that we included a potential moderator Z in the current example. We would now have the following equations:

$$\begin{aligned} X_1 &= \beta_{10} + (\beta_{12}X_2 + \beta_{13}X_3 + \beta_{1Z}Z) + (\delta_{1,2Z}X_2Z + \delta_{1,3Z}X_3Z) + \varepsilon_1 \\ X_2 &= \beta_{20} + (\beta_{21}X_1 + \beta_{23}X_3 + \beta_{2Z}Z) + (\delta_{2,1Z}X_1Z + \delta_{2,3Z}X_3Z) + \varepsilon_2 \\ X_3 &= \beta_{30} + (\beta_{31}X_1 + \beta_{32}X_2 + \beta_{3Z}Z) + (\delta_{3,1Z}X_1Z + \delta_{3,2Z}X_2Z) + \varepsilon_3. \end{aligned} \tag{2.4}$$

Here I use parentheses to separate main effect terms from interaction terms, where parameters associated with main effects are represented by β and those associated with interactions are represented by δ—this distinction is purely for notational clarity.

In these cases, we cannot directly aggregate main effect parameters and transform them into partial correlations. However, we can still approximate the conditional dependencies between variables by averaging over relevant pairs of coefficients; this is a common approach taken in the psychological network literature, even when interactions are not included and exact partial correlations could otherwise be computed (e.g., Haslbeck & Waldorp, 2015; Epskamp et al., 2018c). While this approach seems simple enough, the examples I've provided so far exemplify cases where each nodewise regression is saturated with all possible terms. In practice, however, model selection procedures are often implemented to determine whether or not all terms should be included in the final models.

For example, perhaps we find that X_2 strongly predicts X_1, and so we estimate β_{12}. But, based on theory or some model selection procedure we conclude that the converse is not true (i.e., X_1 is not an important predictor of X_2), and so opt to exclude β_{21} from the final set of models. This means that we implicitly assume $\beta_{21} = 0$. Now we must choose whether to draw an edge between X_1 and X_2 in the resulting network. It is common to make this decision by employing either the 'AND' or 'OR' rule: with the 'AND' rule, we only draw an edge between X_1 and X_2 if both β_{12} and β_{21} are estimated to be non-zero, whereas with the 'OR' rule we draw an edge between the two nodes if at least one of the two relevant parameters is non-zero. In the current example, the edge between X_1 and X_2 would be calculated as $(\beta_{12} + 0)/2$.

2.2 Analyzing Moderated Networks

It will be helpful to use an example that illustrates the process of analyzing moderated networks. For simplicity, I will use a subset of the `msq` dataset from the R package `psych` (Revelle, 2018). The data consist of $n = 3896$ responses to 75 items on the Motivational State Questionnaire, which measures a variety of mood states on scales ranging from 0 to 3. While these variables are clearly ordinal rather than continuous, I chose these data because they resemble a common scenario in psychological research. For this example I will only focus on 5 items from the questionnaire: *hostile*, *lonely*, *nervous*, *sleepy*, and *depressed*. Here, we will treat *depressed* as an exogenous moderator variable, and will investigate whether the relationships among the other 4 mood states vary across levels of depression.

2.2.1 Visualization

First, we begin by mean-centering the data. Although given that 0 is a value on the scales, it could be argued that mean-centering is not necessary in this case. Nevertheless, centering allows us to assess the relationships between variables at their mean levels, and also reduces the correlations between multiplicative terms and their constituents (Lance, 1988; Brambor et al., 2006). The next

step is to fit 4 nodewise regressions, one for each of the 4 outcomes: *hostile*, *lonely*, *nervous*, and *sleepy*. Each regression therefore contains 7 predictors, with 4 main effects and 3 interaction terms. Upon aggregating the coefficients we obtain the following plot.

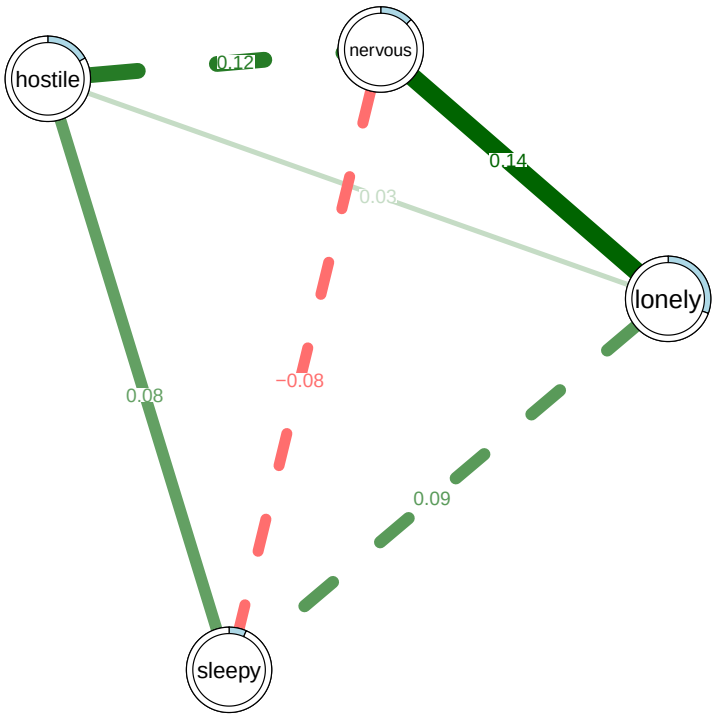

Figure 2.1: Moderated network model at *depressed* $= 0$, plotted with the AND rule and all interaction terms included in each nodewise regression.

In Figure 2.1, green lines indicate a positive relationship between two variables, red lines indicate negative relationships, and dashed lines indicate that the relationship between two nodes is moderated by the *depressed* variable at $p < .05$. Here the coefficients have been aggregated using the AND rule. Given that all models are saturated, applying either the AND or OR rule produces the same aggregated parameters. However, the rule also determines whether a dashed line is used to indicate the presence of moderation effects—specifically, when using the AND rule I only use a dashed line if *both* relevant interaction terms pass a threshold for significance, while for the OR rule I would draw a dashed line if *at least one* of the two relevant interactions pass the threshold.

To elaborate, here the interaction term *nervous* $\times$ *depressed* is a significant predictor of *hostile*,

and the interaction *hostile* $\times$ *depressed* is a significant predictor of *nervous*. Thus, a dashed line is drawn between the two corresponding nodes in the network. Conversely, the interaction *sleepy* $\times$ *depressed* is a significant predictor of *hostile*, but *hostile* $\times$ *depressed* is not a significant predictor of *sleepy*. So, using the AND rule, the edge between *hostile* and *sleepy* is represented with a solid line, given that only one of the two relevant interaction terms are significant—this edge would be drawn as a dashed line if we chose to apply the OR rule instead.

2.2.2 Visualizing the exogenous moderator

The *depressed* variable is not represented in the overall network because it is not being considered as an outcome. However, this variable still serves as a predictor in all 4 regression equations. Thus, we can choose to visualize this variable as an exogenous node in the network as well. In Figure 2.2, I use a square to represent *depressed* along with arrows denoting its links to the other variables to indicate that this variable only serves as a predictor and is therefore conceptualized as being external to the network. This same approach is taken when other exogenous covariates are included in the model, even if they don't serve as potential moderators.

2.2.3 Conditional networks

Given that there are interaction effects in these models, we cannot interpret the network as representing average or overall conditional dependencies between variables. Instead, these values denote the conditional dependencies between nodes when *depressed* $= 0$; which, since all of the variables have been mean-centered in this example, is equal to the sample mean of *depressed*. The dashed lines in Figures 2.1 & 2.2 then show us which relationships we should expect to vary across levels of the moderator, while solid lines show us which relationships we should expect to remain constant. Thus it is important to investigate how the network changes across different levels of the moderator, in order to gain more detailed insights into the nature of the interaction effects.

For this example, we will look at how the network changes at increasing levels of depression. That is, using equations 1.3 & 1.4 we can compute marginal values of the slope parameters (along

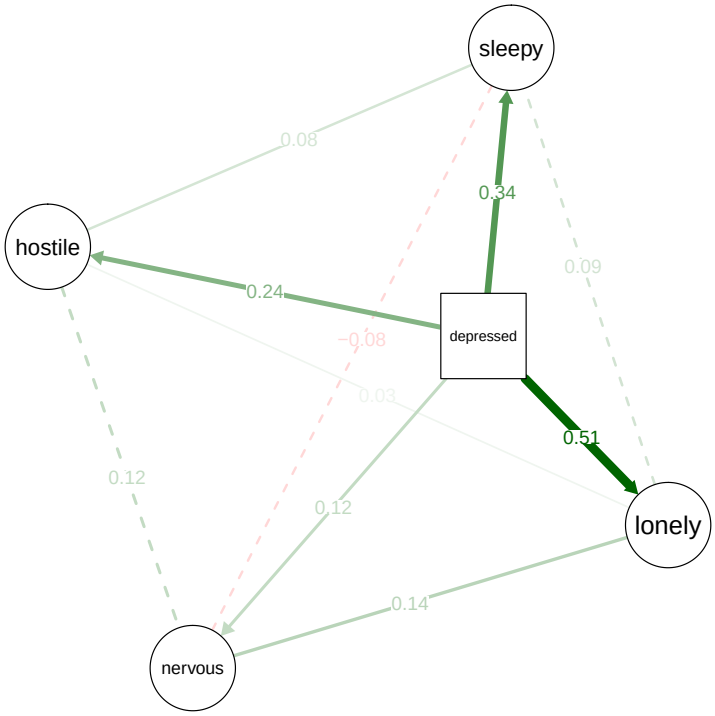

Figure 2.2: Moderated network model with the exogenous moderator plotted. The color saturation of the edges is scaled against the largest edge weight, which is represented by the beta weight of *depressed* on *lonely*.

with their corresponding standard errors) after conditioning on fixed values of the *depressed* variable (i.e., conditional effects). This is the same approach taken in standard moderation analysis when, given a continuous moderator, researchers often visualize slopes at mean levels of the moderator as well as at $+/-1$ *SD*. Given that the *depressed* variable is positively skewed in this case, however, we will instead compute marginal slopes for the values *depressed* $\in \{0, 1, 2\}$.

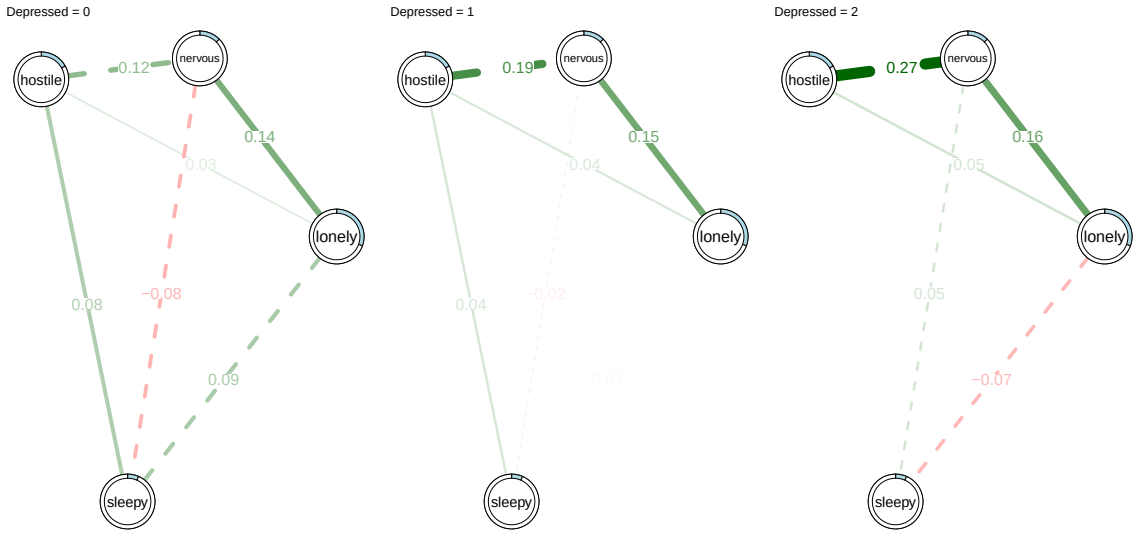

Figure 2.3: Conditional network models at the values of *depressed* $\in \{0, 1, 2\}$. Plotted with the AND rule and no significance thresholding.

I refer to the plots in Figure 2.3 as *conditional networks* to indicate that they exhibit relationships between variables after conditioning on fixed values of the moderator. In fact, any time we include interaction terms in these models we will always have a conditional network. You'll notice that the first network here is identical to the one previously shown; this is because including interaction terms implicitly means that all constituent main effects represent the slopes when the moderator equals 0. The only difference between the two visualizations lies in the width and saturation of the edges, as those in the conditional networks plotted in Figure 2.3 are scaled against the largest edge weight, which is the edge connecting *hostile* and *nervous* in the network where *depressed* $= 2$.

Lastly, while thus far the AND rule has only determined whether or not edges are drawn with a dashed or solid line (based on the presence of significant interaction terms), we can similarly

apply the thresholding rule to all model parameters. That is, we can determine whether to draw an edge between two nodes based on whether the main effect terms are significant at some threshold (here, $p < .05$). Using the AND rule, we determine that an edge is drawn between two nodes if both relevant main effects are significant at this cutoff. Thus, in Figure 2.4, we have a set of networks that only represent the most important relationships between variables, in addition to the most important interaction effects.

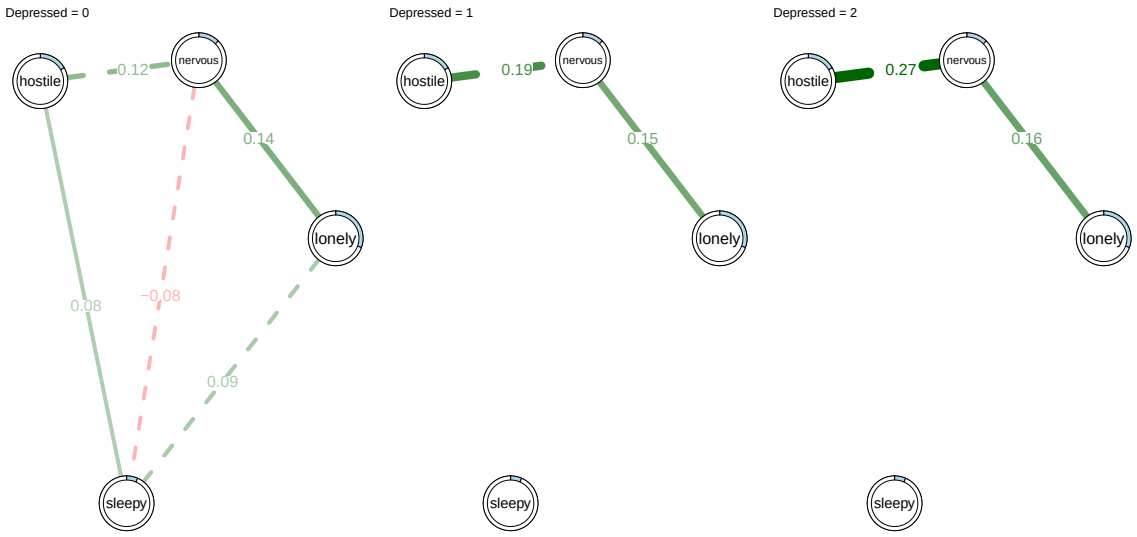

Figure 2.4: Conditional network models at the values of *depressed* $\in \{0, 1, 2\}$. Plotted with the AND rule and a significance threshold of $p < .05$.

2.2.4 Interpreting results

We can now more clearly see how the network structure changes at increasing levels of depression. Looking only at the plots where thresholding was applied (i.e., Figure 2.4), we can see that only two relationships persist across levels of depression: (a) that between *nervous* and *hostile*, and (b) between *nervous* and *lonely*. Moreover, we can see that the latter relation remains relatively constant across levels of depression, spanning $[.14, .16]$, while the former relation exhibits a significant increase as depression increases, as reflected by the significant interaction effects that modulate the strength of the edge.

At low levels of depression, we see some other significant, yet small, relationships: (a) a positive relationship between *hostile* and *sleepy*, (b) a positive relationship between *sleepy* and *lonely*, and (c) a negative relationship between *nervous* and *sleepy*. None of these relationships breach our significance threshold at higher levels of depression. The latter two edges appear with dashed lines, however, indicating that they are significantly moderated by levels of depression. We can get a clearer sense of what this means by looking at the non-thresholded set of models in Figure 2.3. Specifically, both of these relationships switch signs at higher levels of depression, although they do not reach significance in those models.

If we want to hone-in on any of these results more specifically, we can also look at the marginal changes in the slopes across levels of the moderator, in terms of conditional marginal effects plots. In Figure 2.5, we can clearly see how the relationship between *hostile* and *nervous* increases as depression increases, as well as how the signs of the aforementioned relationships change across this range as well. Lastly, some limitations of this investigation are made clearer in these marginal effects plots.

First, the distribution of the *depressed* variable is highly right-skewed; there are a large number of observations at low levels of depression, with fewer and fewer observations as levels of depression increase. Second, the ordinal nature of this variable is made very apparent in the histograms at the base of these plots. Thus, the treatment of this variable as continuous—both in the marginal effects plots and in the analysis—seems somewhat dubious. It would be more appropriate to treat the 4 possible responses as discrete, ordered categories, at least based on the available sample values. Lastly, the values $\{0, 1, 2\}$ are not actually represented within the sample once the variable has been mean-centered, and so it would make more sense to plot networks at the values which are used in the analysis. I only chose the three values here for simplicity in this demonstration.

2.2.5 Summary

The goal of this section was to demonstrate a very basic analysis using the moderated network framework. We can see in this example that it is not enough to simply examine one network to

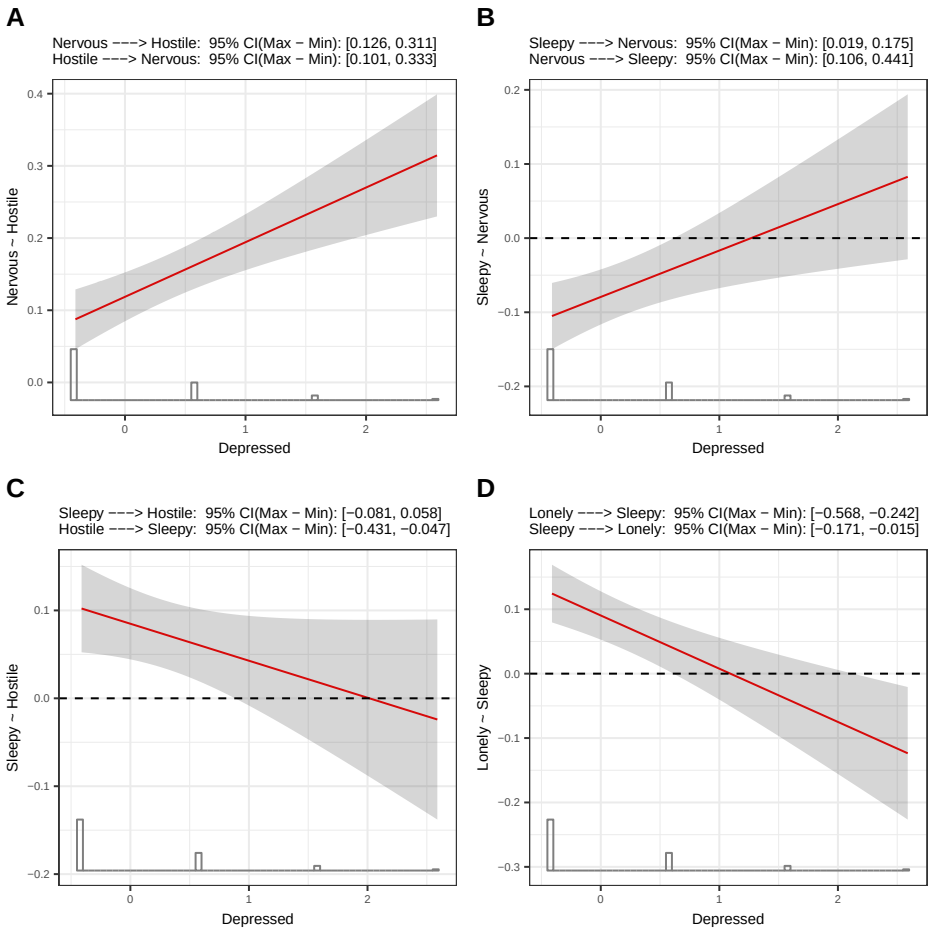

Figure 2.5: Four plots of conditional marginal effects. Each plot represents average values across both relevant interaction terms. In (A), the plot shows the average of two separate conditional marginal effects: that of *hostile* × *depressed* on *nervous*, as well as for *nervous* × *depressed* on *hostile*. In (B), the plot shows the average of the conditional marginal effects of *sleepy* × *depressed* on *nervous*, as well as for *nervous* × *depressed* on *sleepy*. In (C), the plot shows the average of the conditional marginal effects of *sleepy* × *depressed* on *hostile*, as well as for *hostile* × *depressed* on *sleepy*. In (D), this is the average conditional marginal effects of *lonely* × *depressed* on *sleepy* and *sleepy* × *depressed* on *lonely*.

understand the conditional dependencies among variables that are subject to interaction effects. Plotting and investigating conditional networks is a crucial part of the analysis, and can reveal patterns in the data that highlight which relationships are most important, which ones change across levels of the moderator, and which relations only emerge under certain conditions (e.g., low values of depression). Continuing with the present example, I will discuss goodness-of-fit indices and model comparison functions that further expand on this type of investigation.

2.3 Model Comparison

Beyond analyzing specific properties of the variables' conditional dependence structure, procedures for model comparison and variable selection (in this case, edge-selection) are important to determine whether or not interaction effects are valuable for the models, as well as whether more parsimonious models with fewer edges can be specified without sacrificing explanatory power. Luckily, when standard estimation procedures are used (as in the current example), standard fit indices and model comparison tests can be employed. There are two levels at which these indices can be assessed: (a) nodewise, and (b) global model fit.

2.3.1 Nodewise Fit Measures

In the case of models where all outcome variables are assumed to be continuous, we can calculate measures of predictive accuracy for each of the nodewise regressions such as R^2, and R^2_{adj} to determine the proportion of the response variance accounted for by the predictors overall, as well as after controlling for the number of predictors (respectively). Additionally, we can assess model fit by computing the log-likelihood, which for a continuous outcome $\mathbf{x}^{(j)}$ is

$$\ell(\hat{\boldsymbol{\beta}}, \hat{\sigma^2} \,|\, \mathbf{X}) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\hat{\sigma}^2) - \frac{1}{2\hat{\sigma}^2}\sum_{i=1}^{n}(x_i^{(j)} - \mathbf{x}_i^{(-j)}\hat{\boldsymbol{\beta}})^2, \tag{2.5}$$

where the superscript $^{(j)}$ indicates the jth outcome variable, and $^{(-j)}$ denotes the set of columns from the design matrix (here, $\mathbf{X}$) that excludes the jth variable. The predicted values of $\mathbf{x}^{(j)}$ are

therefore defined as the dot product $\hat{\mathbf{x}}^{(j)} = \mathbf{X}^{(-j)}\hat{\boldsymbol{\beta}}$, and the MLE for the residual variance is $\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i^{(j)} - \hat{x}_i^{(j)})^2$.

Using $\ell(\hat{\boldsymbol{\beta}}, \hat{\sigma^2} \,|\, \mathbf{X})$ we can then compare one model of j (perhaps one including interaction terms) with another by performing a *likelihood ratio test* (LRT), or by computing other fit indices such as the AIC and BIC. These information criteria are defined as

$$\text{AIC} = 2k - 2\ell(\hat{\boldsymbol{\beta}}, \hat{\sigma}^2 \,|\, \mathbf{X}), \quad \text{and} \quad \text{BIC} = k\log(n) - 2\ell(\hat{\boldsymbol{\beta}}, \hat{\sigma^2} \,|\, \mathbf{X}), \tag{2.6}$$

with k being the number of parameters estimated in the model. The AIC and BIC are transformations of the log-likelihood that afford different emphases in the model selection process by penalizing models with a larger number of parameters. The BIC enforces a stronger penalty than the AIC on models with a greater number of parameters, and is thereby the more conservative of the two (Yang, 2005). These criteria are particularly important when we have a large number of variables to consider, or are especially concerned with preserving model fit while limiting ourselves to select a more parsimonious model.

In the `modnets` package, all of these criteria are made available to assess model fit. Nodewise LRTs can be conducted to compare models for each node, individually, across different specifications of the network, and indices of predictive accuracy—namely, R^2 and R^2_{adj}—can be depicted graphically to accompany the network representations. This approach was first used by Haslbeck & Fried (2017), but has been expanded in the `modnets` package to allow for model comparison. For instance, in Figure 2.6 we visualize the same model as in Figure 2.1, but use blue rings around each node in (A) to denote the R^2 value of each model in comparison with a model that excludes the interaction terms, as well as the *depressed* variable. Specifically, the light blue represents the R^2 of the reduced model (excluding interaction terms and the *depressed*) variable, while the darker blue represents the additional R^2_Δ that is contributed by including *depressed* and the accompanying interaction terms.

When we include *depressed* as a covariate, however, we can now see that the increase in R^2

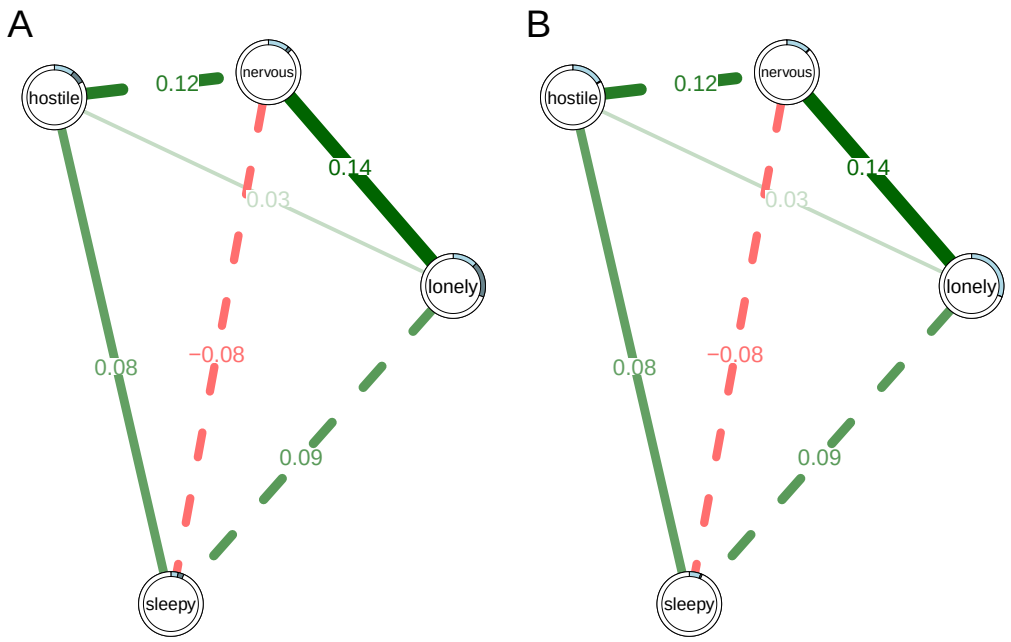

Figure 2.6: These plots show the same network model as in Figure 2, but with two different comparisons represented by the blue shading in each node. In (A), the light blue shading represents the R^2 for each nodewise regression where *depression* is not included in any of the models, while the dark blue shading shows the R^2_Δ given the models including *depression* as well as all of its interaction effects. In (B), the models being compared to the moderation models include *depressed* as a covariate.

after including the interactions is very small (B). This highlights the importance of using fit indices in addition to these metrics in order to determine which model should be interpreted.

2.3.2 Global Fit Measures

It is possible to obtain similar fit measures to describe the fit of all nodewise models taken together. That is, in certain circumstances we can compute the multivariate log-likelihood as well as the corresponding AIC and BIC values to assess global fit across models. This is currently only possible when all outcomes are continuous and assumed to be multivariate normal. However, even this may be undefined in certain cases when many interaction terms are included, particularly when this prevents the residual covariance matrix from being positive definite (Yang et al., 2014; Haslbeck et al., 2018). The multivariate normal log-likelihood can be written as

$$\ell(\hat{\mathbf{M}}, \hat{\mathbf{\Sigma}} \,|\, \mathbf{X}) = \sum_{i=1}^{n} -\frac{m}{2}\log(2\pi) - \frac{1}{2}\log(|\hat{\mathbf{\Sigma}}|) - \frac{1}{2}(\mathbf{x}_i - \hat{\mathbf{x}}_i)\hat{\mathbf{\Sigma}}^{-1}(\mathbf{x}_i - \hat{\mathbf{x}}_i) \tag{2.7}$$

where m is the number of nodewise regression models being estimated (which may be different from p, the number of variables, e.g. when the moderator is exogenous), $\hat{\mathbf{M}}$ is an $n \times m$ matrix containing the predicted values for each nodewise regression, $\hat{\mathbf{\Sigma}}$ is the $m \times m$ residual covariance matrix, and $|\hat{\mathbf{\Sigma}}|$ is its determinant.

The global AIC and BIC can also be calculated from $\ell(\hat{\mathbf{M}}, \hat{\mathbf{\Sigma}} \,|\, \mathbf{X})$, where we supplant k with k^* to denote the number of estimated parameters summed across *all* nodewise models. From here, we can conduct global LRTs and use the other fit indices to compare network models taken as a whole.

In sum, we can use both nodewise fit measures along with global fit measures to optimize information criteria at each level of analysis and select the best model in accordance with our (the researcher's) priorities. In each of these cases, however, I am assuming that there is a clear set of candidate models for us to consider. In the present example, I've presented 3 clear candidate models: (a) the set of models excluding *depressed*, (b) the set of models including *depressed* as

a covariate, and (c) the set of models including *depressed* and its associated interaction terms as covariates. Yet, it may not always be this easy, as we may have many more variables to consider, or may be conducting an even more exploratory analysis wherein the goal is to assess an array of potential moderators and determine the most parsimonious set of models.

2.4 General Discussion

As psychological network models become more widely used across different areas of research, it is clear that methodological extensions are much needed. One recent approach spearheaded by Epskamp et al. (2017) has been to integrate network models with latent variable models in an effort to create 'hybrid networks' for studying psychological phenomena. Other new approaches have included networks for time-series models, particularly those that apply to multiple subjects sampled over time (Hamaker et al., 2018), as well as those that use non-parametric, time-varying models to accommodate data that violate parametric model assumptions (Bringmann et al., 2017). The project described in this paper aims to expand on network models in another, complementary way, and that is by extending current approaches to move beyond strictly pairwise relationships and include higher-order effects such as moderators and interaction terms.

In the next chapter, I describe models selection approaches for MNMs in detail and discuss how we can use *penalized estimators* to obtain sparse, stable, and interpretable models. The inclusion of interaction terms introduces some more than trivial challenges for how best to go about this process, and so specific methods that accommodate these challenges are investigated. Secondly, I present an extension of a post-hoc stability analysis developed by Epskamp et al. (2018a) for studying the influence of sampling variability on parameter estimates from network models. Importantly, this can help us determine how stable interaction estimates from an MNM are, and so constitutes an important step in the analysis and application of these methods to empirical data.

Chapter 3

Model Selection and Network Stability

So far I have covered the background and foundation of estimating moderated networks in cross-sectional data. Before moving on to other applications (i.e., temporal networks), however, there are a few important topics in need of further discussion. These include *model selection* and post-hoc analyses of *network stability*. In this chapter I cover each of these concepts in turn; first, I briefly review approaches to model selection (particularly ℓ_1-regularization; i.e., the *LASSO*) commonly used in the psychological network literature, as well as discuss some unique challenges that emerge for these techniques when we move to moderated networks. In short, the problem in applying standard model selection techniques to MNMs is that they fail to ensure all relevant lower-order terms remain in a model when higher-order interactions are included. Consequently, I suggest an alternative technique—the *hierarchical LASSO*—for use with moderated networks, as well as introduce three novel resampling approaches that can be used for model selection in both unmoderated and moderated contexts. I conclude by discussing the importance of evaluating the stability of an estimated model, and present a method for assessing (post-hoc) the extent to which sampling variability influences different aspects of the network.

3.1 Model Selection

Model selection is a central concept, and issue, in fitting complex multivariate models. In general, model selection techniques are aimed at answering the question: *Given the array of possible models that could be fit to the data, how do we choose the best one?* And while the way in which we determine which model is 'best' will vary across circumstances, there are at least two basic goals

Estimation	Intercepts	Moderators	Total Parameters
Joint estimation	No intercepts	No moderators	$\frac{p(p-1)}{2}$
Nodewise estimation	p intercepts	No moderators	p^2
Nodewise estimation	p intercepts	One exogenous moderator	$2p^2$
Nodewise estimation	p intercepts	p endogenous moderators	$p\left(\frac{p!}{2(p-2)!}+1\right)$

Table 3.1: Total parameters for different estimation and moderation techniques in cross-sectional networks (excluding residual variances).

common to most approaches:

1. To find a model that provides an optimal balance between parsimony and accuracy.
2. To find a model from which meaningful, interpretable results can be obtained.

When it comes to psychological network models, these essentially boil down to obtaining a network with as few connections as needed to account for the data.[1] The reason this is both an important and challenging goal is that network models typically involve estimating a large number of parameters. This is already a central issue for unmoderated networks (e.g., Epskamp & Fried, 2018b), and is at least twice as complicated for even the most basic moderated networks; adding one exogenous moderator to a model quite literally doubles the total number of parameters (as shown in Table 3.1). Moreover, this number increases rapidly as more nodes or moderators are included. For simplicity, in this paper I only focus on situations with one exogenous moderator, especially because this mimics common applications of moderator analysis to hypothesis testing in psychology.

There are a variety of terms used in the network literature to refer to model selection, such as *variable selection*, *edge selection*, or *structure learning* (Drton & Maathuis, 2017). But in general these concepts all refer to the process of determining which nodes should be connected by an edge in a network. The terms *variable selection* and *model selection* are sometimes used in slightly different ways, however, where 'variable selection' refers to the process of determining which predictors to include in a univariate regression model—that is, should we estimate all possible interactions/pairwise coefficients, or only some? 'Model selection', on the other hand, is a more

[1] That is, as few connections as possible to adequately explain the inverse-covariance structure of the data.

general term that refers to the process of comparing models and selecting the final one to interpret. With the methods described in this chapter, I primarily use the term 'variable selection', as here we are concerned with how to generate more parsimonious models by constraining the number of parameters in each nodewise regression. And although in Chapter 5 we'll see that temporal MNMs are estimated in a multivariate fashion, the methods presented here function exclusively through the sequential estimation approach, and so are often performed with a 'two-stage' approach where variable selection is performed prior to (rather than in conjunction with) estimation.

3.2 Variable Selection in Moderated Networks

In general, by employing variable selection techniques we aim to estimate a relatively sparse network structure in order to control for spurious relationships and reduce the rate of false positive associations (Epskamp & Fried, 2018b). Moreover, we can produce a more interpretable model by constraining the number of parameters to only include those that reflect the most robust associations. A simple method of doing this is by fitting a saturated model and then removing connections that fail to reach a threshold of statistical significance (Drton & Perlman, 2004). Alternatively, when joint estimation techniques are used to construct an unmoderated network, confidence intervals can be computed for partial correlations via Fisher's z-transformation (e.g., Williams et al., 2019). However, these approaches may pose a problem of multiple testing (Drton & Perlman, 2007), and many methods of controlling for this problem (such as Bonferroni adjustment) can result in a loss of statistical power (Costantini et al., 2015).

3.2.1 Penalized Estimation via the LASSO

An alternative approach to thresholding that is commonly used in psychological networks is *regularization*, particularly ℓ_1-penalized estimation via the 'least absolute shrinkage and selection operator', or LASSO (Tibshirani, 1996). The LASSO was designed for high-dimensional data applications (where $p \gg n$), and can be thought of as a penalized extension of ordinary least squares

(OLS). Importantly, it has the unique property of performing variable selection in concert with model estimation by shrinking regression coefficients towards zero—particularly, by estimating otherwise small coefficients to be exactly zero. The general form of the LASSO can be stated as:

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} (y_i - \mathbf{x}_i \boldsymbol{\beta})^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}, \tag{3.1}$$

where λ is the penalty or 'tuning' parameter that weights the magnitude of the coefficients, and $\hat{\boldsymbol{\beta}}$ is the final vector that minimizes both the residual sum of squares (RSS) and the penalized coefficient values. The objective represented in Equation 3.1 can therefore be conceptualized simply as finding the β coefficients that minimize: RSS + penalty.

The tuning parameter λ governs the strength of the penalty, and thereby the degree of shrinkage which is applied to the coefficients. Larger values of λ enforce a larger penalty, which is likely to translate into more parameters being set to zero. At the most extreme, all coefficients may be set to zero, leading to an intercept-only model, and conversely when this parameter is small (or is itself zero), the model will converge toward unpenalized OLS estimates. The optimal value of this parameter is often determined through k-fold cross-validation, where an array of λ values are investigated by fitting models to different subsets of the data and then evaluating the quality of predictions on k hold-out sets to determine which value produces the lowest cross-validated mean-squared error. Information criteria (such as the AIC and BIC) may be used as an alternative to cross-validation for this purpose, but in all cases the goal is simply to find which value of λ that results in a model with the best balance between accuracy and parsimony.

The primary attractive feature of the LASSO is that it narrows down the number of predictors to only the most important (i.e., only those with the strongest relationships to the outcome) by setting coefficients for other predictors to zero. Not only does this have the advantage of making models more interpretable, but it can also increase prediction accuracy by reducing the possibility of overfitting—that is, fitting a model that gives too much weight to the noise within some particular dataset, and thereby creates a model that fails to generalize well to new cases.

3.2.2 The Graphical LASSO

One popular variant of the LASSO often used to estimate psychological networks is the GLASSO, or *graphical LASSO*, which was designed specifically for sparse inverse covariance estimation (Friedman et al., 2008). Here, rather than using the least-squares method in Equation 3.1, we instead search for penalized estimates of $\hat{\boldsymbol{\Theta}}$ such that

$$\hat{\boldsymbol{\Theta}} = \underset{\boldsymbol{\Theta} \geq 0}{\arg\min} \left\{ \log\det(\boldsymbol{\Theta}) - \text{tr}(\mathbf{S}\boldsymbol{\Theta}) - \lambda \sum_{i \neq j} |\boldsymbol{\Theta}_{i,j}| \right\}, \tag{3.2}$$

where λ is the penalty that shrinks (or 'regularizes') the estimates (as in Eq. 3.1), $\mathbf{S}$ is the sample covariance matrix, and we assume that $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$, i.e., that $\boldsymbol{\Theta}$ is the true inverse covariance structure. After finding $\hat{\boldsymbol{\Theta}}$, we can then standardize the result to obtain $\hat{\boldsymbol{\Omega}}$, which is the partial correlation matrix used to characterize the GGM. With the GLASSO, we directly penalize elements of the inverse covariance matrix $\boldsymbol{\Theta}$, rather than obtain them by aggregating p nodewise estimates of $\boldsymbol{\beta}$.

Both methods—the LASSO and the GLASSO—are quite common in the psychological network literature, where nodewise estimation techniques utilize the former (e.g., Haslbeck & Waldorp, 2015), and joint estimation techniques utilize the latter (e.g., Epskamp & Fried, 2018b). The GLASSO is typically taken to be the 'default' estimator for cross-sectional models, as it is both more computationally efficient and more direct of an approach (given that the ultimate goal is to obtain an estimate of $\boldsymbol{\Omega}$). Importantly, this is often used to fit *sample-split* models, which can be thought of the closest approach used thus far to approximate moderator effects (van Borkulo et al., 2017; Costantini & Epskamp, 2017). Sample-split methods are techniques designed to compare a network structure across two (or, in theory, more) groups. In such cases, the researcher splits a dataset into two subsets based on a binary moderator, or according to a median-split continuous variable, and then compares the networks estimated for each group to determine which edges (if any) differ between the two.

Sample-split methods will be described further in Chapter 4, and are tested against various MNMs in a simulation I conducted to evaluate their performance relative to one another. Given

that these methods have already seen applications in the literature, it will be important to show that a better understanding of moderator effects can be attained when one uses a method more properly suited to the circumstances (i.e., a MNM). As noted in Chapter 2, however, joint estimation techniques are not possible with cross-sectional MNMs, meaning that the GLASSO cannot be utilized in such cases. Moreover, recent research has shown that the GLASSO often fails to effectively control for spurious relationships, even in well-powered situations (Williams et al., 2019). This means that common approaches to detecting moderator effects in network models (i.e., via sample-split model comparisons) may be inadequate in more than one way, and can potentially be improved by the MNM framework.

Before moving on to that investigation, however, we must answer the question: how can we control false positives and utilize regularization in *moderated* networks? Although it is entirely possible to apply the standard LASSO when estimating an MNM, the presence of interaction terms introduces a new challenge; specifically, we now have two *types* of coefficients—those for pairwise relationships (i.e., main effects), and those for higher-level interactions. Critically, the LASSO does not differentiate between these types of coefficients, and so is vulnerable to selecting MNMs that *exclude* relevant main effects. As I'll demonstrate in the following section, omitting main effects can lead to biased estimates of interaction effects, and so presents a new challenge when moving from unmoderated to moderated networks.

3.2.3 Consequences of Omitting Main Effects

It is a problem to employ automatic variable selection techniques when higher-order terms (e.g., interactions) are included in a model, as this will often lead to interactions being included without one or more of their corresponding lower-order terms. This occurs because most common variable selection techniques—such as best subsets selection, stepwise regression, and the LASSO—do not treat these two types of parameters in different ways, and may thereby exclude variables without regard to their structural hierarchy. Indeed, these methods are actually more likely to violate this

constraint than abide by it.[2] This is problematic, as it is widely recognized that omitting constituent terms leads to biased estimates of interaction effects (Brambor et al., 2006; Friedrich, 1982; Aiken & West, 1991; Braumoeller, 2004).

Simply put, in a model where we aim to predict values of Y based on the interaction between X and Z, we must also include both main effect terms in order to obtain unbiased coefficients. Omitting the main effect of X on Y, for instance, is equivalent to assuming that X has no effect on Y when Z is zero. The degree to which this biases the other parameters will depend on how much the true value deviates from zero, as well as the magnitude of the effect relative to the other related parameters. Moreover, the degree to which the variables are correlated and whether or not any are binary also affects this bias. We can see a simple example of this in Figure 3.1. Here I constructed a simple model of $Y = \beta_0 + \beta_1 X + \beta_2 Z + \beta_3 XZ + \varepsilon$, where values of X and Z were drawn separately from a uniform distribution on the unit interval, and Z was then recoded to 1 if $Z \geq 0.5$ and 0 if $Z < 0.5$. The outcome Y was then constructed as

$$Y = 0 + 0.2X - 0.5Z + 0.2XZ + \varepsilon, \quad \text{where} \quad \varepsilon \sim \mathcal{N}(0, 0.5). \tag{3.3}$$

100 datasets were generated from this model for sample sizes $N \in \{50, 60, \ldots, 500\}$, and three separate models were fit in each case: (1) The fully specified model where all terms were included, (2) A model that excluded the main effect of Z, and (3) A model that omitted the main effect of X. For each of the four possible model parameters (including the intercept), I then plotted the average estimate along with standard deviation for each sample size. The solid black line in each facet of Figure 3.1 depicts the true value of that parameter, and we can see just how omitting main effects biases the estimates. The blue line represents the fully-specified model, and is unbiased across the range of sample sizes. Omitting Z ($\beta_2 = -0.5$) causes the greatest shift in the estimates, and we see that even though the true effect of X is relatively small ($\beta_1 = 0.2$), omitting it from the model leads to a distinct bias in the other parameters.

[2]This is simply due to the fact that, given a significant interaction, the likelihood of detecting just one significant main effect is greater than that of detecting two.

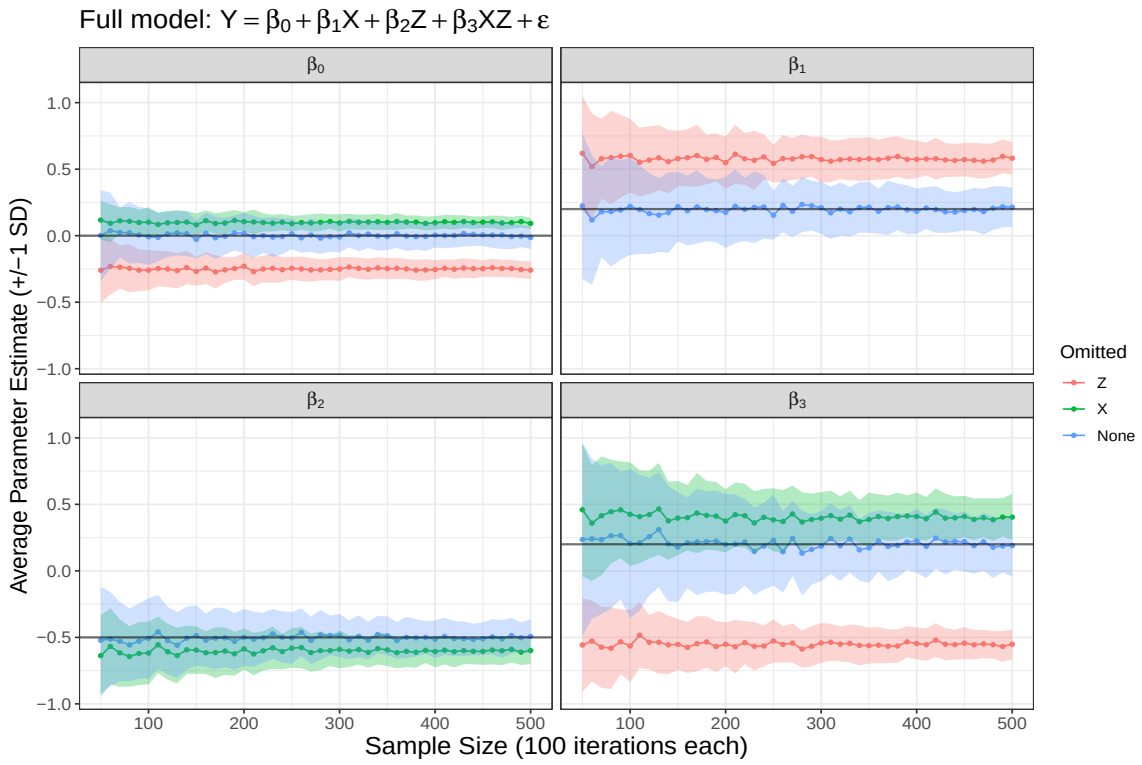

Figure 3.1: Simulated data with estimates of coefficients for different interaction models.

Most notably, we can see that the bias is strongest for the interaction effect itself, β_3. Thus, even in cases where one or more main effects are considered uninteresting, including both terms in the model will be best for obtaining a less biased estimate of the interaction. And while the example here only demonstrates this for a continuous $\times$ categorical interaction, it can be shown (for continuous outcomes) that the potential for bias is even stronger with categorical $\times$ categorical interactions, and for continuous $\times$ continuous is dependent on the degree to which the two predictors are correlated. It should also be noted that while the inclusion of both main effects does not guarantee an entirely unbiased estimate of an interaction effect (e.g., in the presence of model misspecification), it can at least be said that doing so can help to minimize this bias.

This example is meant to highlight the importance of imposing a *strong hierarchy* on models that include higher-order interactions. When a model obeys a strong hierarchy, this simply means that interactions are only included when both main effects are also included.[3] While there may

[3]This extends to any type of higher-order interaction term. For example, when including a three-way interaction we should also include both two-way interactions, as well as all three main effects.

be some circumstances where we have reason to omit main effect terms (or simply have different goals, such as prediction), it should generally be preferred to only select and interpret models that obey this hierarchy. Thus, given that the LASSO and other variable selection techniques do not adhere to this constraint, alternative methods will be necessary to support the analysis of moderated networks.

3.3 Constructing Moderated Networks with the Hierarchical LASSO

One relatively new method that can be used in this situation is the *hierarchical LASSO*, which was explicitly designed to integrate hierarchical constraints into the LASSO (Bien et al., 2013; Lim & Hastie, 2015). This method was first introduced by (Bien et al., 2013), where it has been implemented in the `hierNet` package for R. Lim & Hastie (2015) later refined this method with the `glinternet` package, which utilizes an algorithm that enforces strong hierarchy along with flexible model specification, allowing the user to restrict the search for interactions to one or more candidate variables. Both of these approaches are built on an overlapping *group LASSO*, wherein the LASSO is applied to predictor variables on the basis of their grouping into sets of main effects + interactions, including cases where certain variables may be allowed to participate in more than one interaction (for a detailed discussion, see Lim & Hastie, 2015).

While the hierarchical LASSO is only implemented for univariate models, we can extend it to the graphical modeling domain via nodewise estimation, in the same way that the standard LASSO is applied for unmoderated networks. That is, for each node in a network, we can perform variable selection with the hierarchical LASSO to incorporate both exogenous and endogenous moderators into the models without violating their structural hierarchy. Both the `hierNet` and `glinternet` packages in R produce similar results for a variety of settings, but `glinternet` exhibits a major computational advantage in terms of computing time. Furthermore, while both packages afford searching the entire model space of all possible interactions, `glinternet` also allows the user to select a subset of variables (or a single variable) to serve as candidates for interactions. This is particularly useful when the objective is to evaluate the influence of a particular moderator or set

of moderators according to theory or pre-specified hypotheses.

Usage of the hierarchical LASSO has been fully integrated in the `modnets` package, and allows for extremely flexible model specification as well automated variable selection for the MNMs discussed in this paper. While the hierarchical LASSO eliminates potential bias that would otherwise follow from violations of strong hierarchy, coefficients are still biased towards zero due to the shrinkage employed by the estimator. So, in order to address (although certainly not solve) this issue, the software allows the user to seamlessly separate the selection process from estimation, so that the hierarchical LASSO is applied in a nodewise fashion, and then only those variables/interactions that are selected are included in the final models. These are then fit with OLS for cross-sectional networks, and other least-squares methods for the models described in the following chapters.

Moreover, using classical estimators for obtaining the final coefficients affords the inclusion of standard errors and p-values, which are not obtained by penalized estimators. While this is desirable for error control and for refining the interpreted model via p-value thresholding, it should be noted that the variable selection process make inferences uncertain since the test statistics do not incorporate the potential effects of the selection process. This is known as the problem of 'inference after selection', and widely noted in the statistical literature as a problem for which no standardized solutions are known (Berk et al., 2010). However, this is nothing unique to the particular models or methods at hand, and is rather a more general issue that should be taken into account when reporting results or interpreting models when these methods have been applied.

Although use of the hierarchical LASSO affords automated variable and interaction selection in moderated networks, this method has the same characteristic as the LASSO in that sparsity is enforced in accordance with fixed values for the parameter λ. Different goals or assumptions about the data may lead to different decisions on how to select a value for λ, and so I have included four primary options for this in the `modnets` package, all of which involve minimizing either information criteria or prediction error. These include: (a) the AIC, (b) the BIC, (c) the extended BIC (EBIC), and (d) k-fold cross-validation.

For each of these procedures, a number of candidate λ values (50, by default) are generated on a log-scale with a range proportional to the smallest λ that returns an empty model. For the first three criterion-based options, these indices are simply calculated for each value of the parameter, and then whichever λ minimizes the criterion is the one selected for the final model. The EBIC is a common index used in graphical models, as it is even more conservative than either the AIC or BIC:

$$\text{EBIC} = k\log(n) - 2\ell(\hat{\boldsymbol{\beta}}, \hat{\sigma}^2|\mathbf{X}) + 2\gamma k\log(p). \tag{3.4}$$

This criterion adds an additional penalty not only for the number of parameters k included in the model, but also for the number of nodes p which is proportional to the number of possible parameters. There is also an extra hyperparameter γ which must be selected by the researcher, which helps to define the strength of this additional penalty. When $\gamma = 0$, this criterion will be equivalent to the BIC. Researchers typically set this value to $\gamma = .5$ at its most conservative, and $\gamma = .25$ for more exploratory settings.

The most common technique for selecting λ, however, is likely to be the fourth method, k-fold cross-validation (CV). This involves splitting the dataset into k complementary subsamples, where each subsample is treated as hold-out (test) set, and models for each value of λ are fit to the remainder of the data, and then prediction error is calculated on the test set for each model. Then, the value of λ that minimizes the average prediction error (typically MSE for continuous data) across test sets is the one that is selected for the final model.

Although it has been shown that these methods perform quite well with both the standard LASSO (e.g., Friedman et al., 010b), and the hierarchical variant (e.g., Lim & Hastie, 2015), there are still some potentially undesirable characteristics of regularization techniques, and variable selection procedures in general, that should be mentioned.

3.3.1 Disadvantages of the LASSO and its Variants

Although the LASSOs have some desirable properties for aiding in interpretation and increasing predictive accuracy, this comes at the cost of producing coefficient estimates that are biased, which as a result may produce inaccurate representations of their 'true' values. While this is not necessarily a problem when prediction is the only interest, it does create an obstacle for inference. Specifically, there is no known way of determining the standard errors for estimates derived from the different LASSO penalties, and thus no way to perform significance tests that can be assumed to reflect the characteristics of their sampling distributions (Tibshirani, 1996). Due to the unique bias induced by these procedures (i.e., that it can lead to some parameters to be estimated at zero), even bootstrapped resampling fails to be a valid method for determining the distributions of the parameters. Bootstrapping may be effective when the true coefficients are especially large, or truly equal to zero, but for cases in between it will usually generate asymmetric distributions that may not describe the population-level associations between the candidate variable(s) and outcome of interest.

A related disadvantage of the LASSO, although not unique to the LASSO, is inconsistent model selection. In cases where small changes in the tuning parameter λ lead to different variables being included or excluded from the final model, it may be difficult to make strong inferences about the values of those variables. To my knowledge, no variable selection technique is immune to this concern, however, and it may be a persistent challenge for any approach of this nature (e.g., best subset selection). However, despite the fact that bootstrapping does not seem to solve the inference problem for the LASSOs, it does provide the researcher with a sense of the selection procedure's stability with regard to each of the predictors. One possible way of increasing confidence in the chosen variables is by reporting the selection rates with reference to each predictor; for instance, we may have greater confidence in choosing variables that are selected for the best model within 90% of subsamples as opposed to those that are chosen only 15% of the time.

To address the first concern regarding biased estimates, I propose using a two-stage method where rather than interpreting the estimates returned by the hierarchical LASSO, the models are

instead fit in a second step using unregularized methods (e.g., OLS) to obtain the final parameter estimates used for interpretation. There is still some uncertainty for inferences on these parameters, but in many cases this type of two-stage approach has been shown to improve the accuracy of the coefficient estimates (Aoshima & Yata, 2011). To address the second concern, I discuss three resampling techniques below that I extended and implemented in R to perform variable selection while taking the consistency of the selection process into account.

3.3.2 Advanced Variable Selection Techniques

To extend the two-stage methods described above, I incorporated three resampling methods into the `modnets` package aimed at addressing the question: *how consistent is selection with the hierarchical LASSO for my sample?* Moreover, how can we use this information to ensure that we choose a stable model? Luckily, a number of researchers have attempted to answer these questions with regard to variable selection and frequentist inference in high-dimensional models (e.g., Wasserman & Roeder, 2009; Meinshausen & Bühlmann, 2010; Meinshausen et al., 2009; Dezeure et al., 2015). Broadly speaking, the methods proposed in these papers involve performing the variable selection and model-fitting procedures in an iterative fashion to subsamples of the original data, then aggregating the results and selecting final models based on error-corrected p-values or by choosing a model with only the most frequently selected predictors. I have extended these methods to the case of using the hierarchical LASSO to select variables for moderated networks, and will present two basic algorithms that have been implemented in the `modnets` package below.

The first two methods are variations on the multi-sample split approach with p-value aggregation developed by Meinshausen et al. (2009). In general, this approach involves splitting the dataset at random points into two complementary halves, performing variable selection (using the methods described above) on the first half, and then fitting the resultant models to the second half. This process is performed iteratively, and the results of the final model from each iteration are stored to obtain distributions of model parameters. Importantly, a distribution of p-values is also obtained for each parameter, and these can then be aggregated based on quantiles to return a fi-

nal p-value that summarize the degree of support for including that parameter in the model based on the resampling procedure. Moreover, an additional correction can be applied to control the family-wise error rate (FWER) before aggregating the p-values if desired.

I've created a multi-sample split version of this, that essentially extends the above procedure into a multivariate context, along with a bootstrapping variant. I'll present the main algorithm that I've implemented in the `modnets` function `resample` borrowing some notation from Meinshausen et al. (2009).

Algorithm 1 Multi-Sample Split Method

1. For iterations $b = 1, \ldots, B$:
   (a) Randomly split the data into two disjoint groups $D^{(b)}_{train}$ and $D^{(b)}_{test}$. The argument `split` can be used to indicate the size of $D^{(b)}_{train}$ as a proportion of N (defaults to .5).
   (b) For each node $i = 1, \ldots, p$:
      i. The `glinternet` algorithm is applied to $D^{(b)}_{train}$ and obtains the active set of predictors $\tilde{S}^{(b)}_i$.
      ii. The predictors selected in $\tilde{S}^{(b)}_i$ are fit to $D^{(b)}_{test}$ with OLS and the corresponding p-values $\tilde{P}^{(b)}_{i,j}$ are calculated for $j \in \tilde{S}^{(b)}_i$
      iii. The p-values for variables not included in $\tilde{S}^{(b)}_i$ are set to $P^{(b)}_{i,j} = 1,\ j \notin \tilde{S}^{(b)}_i$
      iv. OPTIONAL: A Bonferroni correction can be applied to each of the p-values:

$$P^{(b)}_{i,j} = \min\left(\tilde{P}^{(b)}_{i,j}|\tilde{S}^{(b)}_i|,\ 1\right),\ j = 1, \ldots, k$$

2. There are now B p-values for each predictor $j \in \{1, \ldots, k\}$ of each node $i \in \{1, \ldots, p\}$.
3. In order to aggregate the B p-values for each predictor, we will be adding a penalty θ_α to its associated quantile function. The candidate quantiles will be a sequence γ,

$$\text{from} \quad \frac{\lceil \alpha \cdot B \rceil}{B} \quad \text{to} \quad (1 - 1/B) \quad \text{by} \quad 1/B.$$

4. The penalty is then: $\theta_\alpha = 1 - \log(\gamma_{\min})$.
5. For each node $i = 1, \ldots, p$:
   (a) For each predictor $j = 1, \ldots, k$:
      i. Given the empirical quantile function $q_\gamma(\cdot)$, the aggregated p-value will be:

$$P_{i,j} = \min\left\{1,\ \theta_\alpha \cdot \min\left\{1,\ q_\gamma\left(\{P^{(b)}_{i,j}/\gamma;\ b = 1, \ldots, B\}\right)\right\}\right\}.$$

See Meinshausen et al. (2009) for further details on the steps of the algorithm. The objective is to use the iterated sample-splitting procedure to obtain aggregated p-values for each possible predictor of each node in the network, taking into account the sampling procedure and controlling the error rate at some pre-specified level α. The primary innovation here beyond the original algorithm developed by Meinshausen et al. (2009) (and extended by Dezeure et al., 2015) was to create a multivariate, network-specific version that integrates the hierarchical LASSO for variable selection.

Another variation of this algorithm that I've implemented in the `modnets` package performs (non-parametric) bootstrapping rather than multi-sample splitting, and the only difference between that case and the foregoing algorithm is that at each iteration b, rather than splitting the dataset into two disjoint sets $D^{(b)}_{train}$ and $D^{(b)}_{test}$, we simply take a sample of size N from the original data (with replacement), and perform both the variable-selection and model-fitting steps (essentially, the entire first step of the algorithm) on that same sample.

These two algorithms are designed to allow for variable selection based on these adjusted, aggregated p-values. However, another approach we may consider is to select variables based on the frequencies with which they are selected throughout the resampling process. This is referred to as *stability selection*, and involves repeatedly splitting the dataset into separate halves, performing variable selection to each half, and then for each predictor computing the proportion of iterations that it was simultaneously selected in both halves of the data. The algorithm is implemented as follows:

Fundamentally, the goal is to obtain the *simultaneous selection* probabilities for each predictor of each node, which denotes the proportion of iterations where a predictor was selected in each of the two subsamples. The threshold $\pi_{\text{thr}} \in (0, 1)$ is chosen by the researcher, and indicates the lowest simultaneous selection probability required for a predictor to be selected in the final model. The default value is set to $\pi_{\text{thr}} = .6$, as this value has been shown to perform well and lead to consistent model selection by Meinshausen & Bühlmann (2010). Larger values can be chosen for more conservative selection procedures.

Algorithm 2 Stability Selection

1. For iterations $b = 1, \ldots, B$:
   (a) Randomly split the data into two disjoint groups $D_1^{(b)}$ and $D_2^{(b)}$.
   (b) For each node $i = 1, \ldots, p$:
      i. Apply the `glinternet` algorithm to each half of the data to obtain two active sets of predictors $\tilde{S}_{i,1}^{(b)}$ and $\tilde{S}_{i,2}^{(b)}$.
      ii. For each predictor $j \in \{1, \ldots, k\}$, assign a value of 1 to $S_{i,j}^{(b)}$ if $j \in \tilde{S}_{i,1}^{(b)} \cap \tilde{S}_{i,2}^{(b)}$ and 0 otherwise.
2. For each node $i = 1, \ldots, p$:
   (a) For each predictor $j = 1, \ldots, k$:
      i. Compute the proportion of times that variable was selected in both disjoint sets taken at each iteration, and include it in the final set $\tilde{S}_i$ if $\frac{\sum S_{i,j}^{(b)}}{B} \geq \pi_{\text{thr}}$
   (b) Given the active set of variables $\tilde{S}_i$, use OLS to fit the final model to the full dataset.

In each of these cases, `glinternet` is taken as the standard variable selection algorithm in order to search for interactions given a particular moderator or set of moderators, as well as to impose the strong hierarchy constraint throughout the process. The resampling algorithms described above also impose the strong hierarchy constraint when the final models are being selected, to ensure that the procedures do not lead to violations of the constraint. Moreover, in order to make these algorithms as general as possible, other variable selection techniques—such as best subsets selection (via the `leaps` package), as well as the standard LASSO, ridge regression, and elastic net (via the `glmnet` package)—are included for researchers to fit unmoderated networks as well.

One downside of these algorithms, however, is that they can be very computationally expensive. Most of the models fit in the present study, each with $B = 100$ iterations, would take 1-2 hours to run. To reduce the run time, however, I've made it easy to parallelize these procedures through the `resample` function in `modnets`. Nevertheless, it is worth noting that these methods can be computationally burdensome, and thus it will be important to investigate whether this cost comes with the added benefit of improving the accuracy and stability of parameter estimates.

3.4 Network Stability

After the selection process is complete and the final model has been chosen, there is still more we can do in the way of evaluating the stability of our model. The methods described so far have focused on ensuring that variable selection with an MNM obeys certain rules, and is done in a way that takes the consistency of the selections into account. But once we select our model, how can we evaluate its stability?

One way we can conceptualize this question is by thinking of 'stability' as the degree to which certain properties of a model are dependent on the particular make-up of the sample. For instance, imagine that we changed some minor aspect of the sample, such as removing 10 observations. An *unstable* model would be one that changes dramatically even after a relatively small perturbation like this, and a *stable* model would be one that maintains the same general characteristics even after larger perturbations, such as removing 100 observations.

Indeed, a method based on this idea was developed for assessing the stability of centrality estimates in social networks (Costenbader & Valente, 2003). Centrality measures are popular ways of analyzing structural aspects of a network, such as identifying which nodes have the greatest number of connections (degree centrality), or lie on the average shortest path between all possible pairs of nodes (betweenness centrality). Essentially, centrality measures describe some aspect of the interconnectedness of any given variable with the rest of the network.

Some centrality measures have been criticized as not having a clear interpretation for weighted networks (Bringmann et al., 2019), although others—such as *strength centrality* and *expected influence*, have been shown to be relatively stable and have clearer meaning in psychological contexts (Robinaugh et al., 2016). Strength centrality measures the overall connectedness of a node as the sum of the absolute values of all edges connected with it, while expected influence is simply the sum of those raw edge weights. The latter therefore reflects the overall *positive* connectivity of any given node, while the former reflects overall magnitude of each node's connectivity.

3.4.1 Case-Dropping Bootstrap

Strength centrality and expected influence are both frequently studied measures in psychological networks, and it is therefore desirable to consider how these—as well as simply the edge weights and interaction terms—are affected by sampling variability. Thus, the method for assessing stability in social networks was extended into the domain of psychology to achieve this goal, where it was termed the *case-dropping bootstrap*, and has an associated metric—the *correlation stability coefficient*—that can be used to directly quantify the stability of edge weights and centrality estimates (Epskamp et al., 2018a).

The procedure used for the case-dropping bootstrap involves taking bootstrapped subsamples from a dataset while dropping increasingly large proportions of the sample size. For example, if we have a dataset with 100 observations, we could take a 10%-dropped subsample with only 90 observations; and if we took a 25%-dropped subsample then we would only take 75 observations, and so on for increasingly large drops. Then, for each subsample we re-fit the selected model and correlate metrics of interest from that sample with those from the original model. This is done repeatedly and across an array of drop-sizes to investigate the stability of the correlations across each size. Typically, researchers will take 10 subsample sizes spanning from 5–75% drops, meaning that a range of sizes are tested in order to assess the stability of correlations with the original sample as larger amounts of data are dropped.

The *correlation stability coefficient (CS)* then represents the *largest proportion of the sample that can be dropped while retaining a sufficient correlation with the original sample*. This coefficient can be computed and evaluated separately for each metric in order to see which aspects of the network are most robust to sampling variability, as well as which are the least robust. The criterion for a 'sufficient correlation' is decided by the researcher, although Epskamp et al. (2018a) recommend specifying the value as the largest proportion of the sample that can be dropped while retaining a correlation of at least .70 with the original sample across 95% of the subsamples for that size. To see this more clearly, lets take a look at this method applied to the depression data analyzed in the previous chapter.

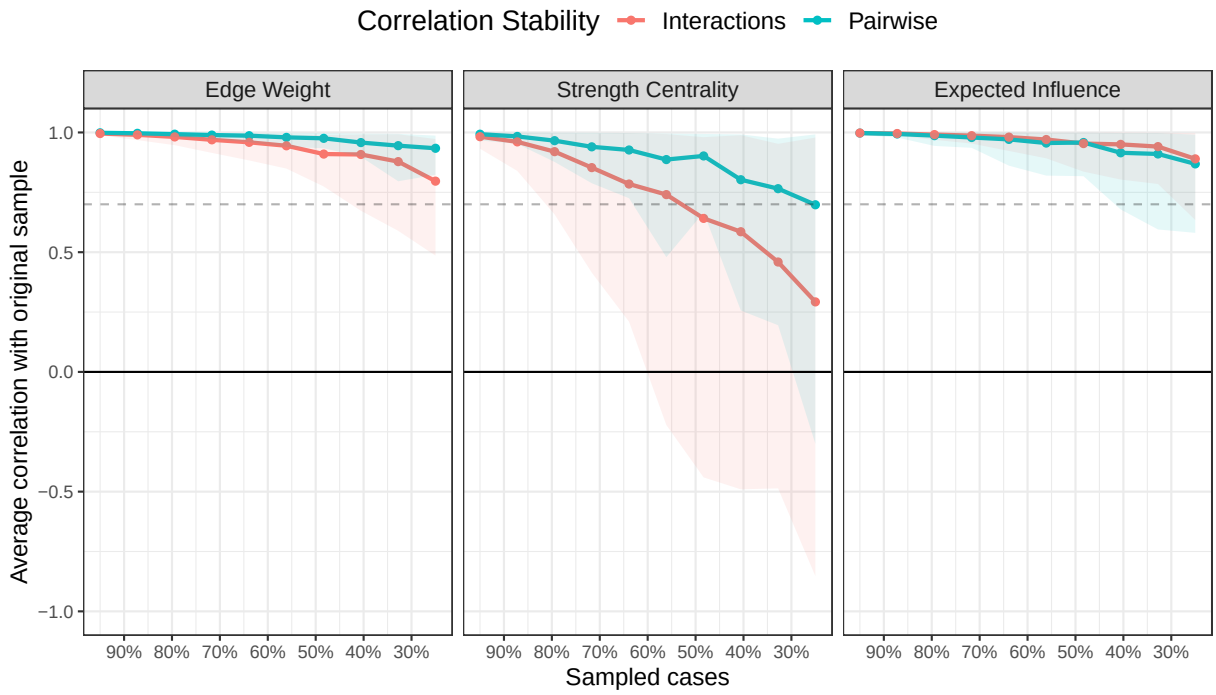

Figure 3.2: Average correlations plotted with 95% coverage intervals for each subsample size. In total, $B = 1000$, which means that there were 100 iterations for each of the 10 drop sizes tested.

In Figure 3.2 we see the average correlations between subsamples of different sizes and the original dataset. Error ribbons reflect the middle 95% of the empirical distribution at each subsample size. In this case, the original data had $N = 3841$ observations, and 10 equally-spaced subsample sizes were tested ranging from 5% drops ($N = 3649$) to 75% drops ($N = 960$). A total of $B = 1000$ iterations were used, meaning that 100 subsamples were taken for each of the 10 drop sizes. This method was first implemented in the `bootnet` package for R (Epskamp & Fried, 2018a), but I have extended it into `modnets` by allowing for the incorporation of interaction effects. Thus, we see that correlations for each metric are computed for both pairwise and interaction terms, and so we can compute separate *CS* coefficients for each type of parameter.

Correlation Stability	$CS(\rho_{.95} \geq .70)$		
	Edge	Strength	EI
Pairwise	.75	.36	.52
Interactions	.52	.13	.67

Table 3.2: Correlation stability coefficients.

In Table 3.2 we see the final *CS* coefficients associated with Figure 3.2. Given that the largest drop was 75%, the largest possible *CS* value was .75. Had we tested a larger range of drop sizes, the range of coefficient values could also change. Looking at the values in Table 3.2, we see that the *CS* value for pairwise edge weights is .75, meaning that even when we dropped 75% of the total sample size, at least 95% of those subsamples had pairwise edge weights correlated with those from the original sample at a value greater than .70. With interaction estimates, however, we see that this value is .52, meaning that the maximum drop-size where a correlation of .70 was retained across 95% of subsamples was 52%. We can see this difference reflected in the left-most plot in Figure 3.2, where the error band around around the pairwise edge weight correlations never goes below the dashed line (where $\rho = .70$), while that for the interaction parameter correlations falls below the line once the sampled proportion falls below 48%.

Based on simulation studies conducted by Epskamp et al. (2018a) and Costenbader & Valente (2003), it is recommended to only interpret properties of network models that have a *CS* coefficient of at least .25, and preferably above .50. Thus, we see that the present dataset actually performs quite well on most metrics, where coefficients for edge weights and expected influence are greater than .50 for all pairwise and interaction terms, while these values are lower for strength centrality. This tells us that we should be cautious in interpreting estimates of strength centrality from this model, as the values have low stability.

As we see here, this method provides us with important information about a network model and allows us to identify statistics that are more or less vulnerable to sampling variability. Moreover, since this method is conducted post-hoc, one can assess the efficacy of any given variable selection method by seeing whether the selected model also produces more stable correlation stability

estimates. And while this method has already seen various applications in the literature, I extend it here to accommodate moderated networks, which allows researchers to gain a more complete picture of their model as well as an understanding of whether or not the estimated interactions 'hold-up' against tests of their stability.

3.5 Summary

In the present chapter, I described the characteristics of various variable selection methods, as well as how I deal with the problem of selection under strong hierarchy in moderated networks. Specifically, I have integrated the hierarchical LASSO into the `modnets` package via the algorithms implemented in the `glinternet` package. Then, I expanded available options for variable selection to include three resampling procedures that incorporate the hierarchical LASSO with moderated networks, as well as other variable selection techniques for other types of networks. As will be described in Chapter 5, these methods can be extended to accommodate temporal networks as well.

The next topic covered in this chapter was network stability; specifically, given that we wish to assess the stability of our models after they have been selected, we can use the case-dropping bootstrap to compute the *CS* coefficient for a variety of metrics we may be interested in. In this chapter I only presented results for edge-weight, strength centrality, and expected influence stability, but we can test any metric we wish to evaluate in a network (e.g., closeness centrality; betweenness). Moreover, we can now conduct these analyses to attain separate estimates for both pairwise and interaction effects.

It is also worth noting that this is the first paper (that I know of) which discusses the application of centrality measures to interaction terms. This is a potentially interesting aspect of moderated networks that has not been considered in this paper, although is certainly a topic worthy of further investigation. For instance, an interaction term with high strength centrality could indicate that it is also the most robust, or that it potentially drives a signification proportion of the patterns seen in the

network.[4] These are purely hypothetical for now, however, as more research is needed to determine what different measures of interaction centrality contribute to an understanding of MNMs. Still, if nothing else the analysis can be used to reveal whether or not the interactions estimated in a model are relatively stable, or whether they are highly sensitive to variations in the sample.

In the next chapter, I cover a large-scale simulation study I conducted to evaluate the performance of moderated network models against more common sample-split techniques (e.g., comparing separate networks fit to different values of a moderator), as well as more broadly to assess the performance of the variable selection techniques discussed in this chapter.

[4] Although, some research has indicated that many causal interpretations of centrality measures are likely to be invalid (Dablander & Hinne, 2019).

Chapter 4

Simulation Study

Next, the goal was to conduct a simulation study comparing different MNM procedures—those that fit saturated models as well as those using the various variable selection techniques—to alternatives that are currently used to approximate certain types of moderation effects. Specifically, when a potential moderator is a continuous variable, median-split or other sample-splitting methods have been used to determine whether high and low levels of the variable are associated with different network structures for the phenomenon at hand (e.g., van Borkulo et al., 2017). This may be considered an indirect way of detecting moderator effects, and until now has been the only method utilized in the psychological network literature for this purpose.

4.1 Sample-Split Methods

There are two methods in particular that have been used for sample-split analyses in the psychological network literature: (1) the *Network Comparison Test (NCT)* (van Borkulo et al., 2017), and (2) the *Fused Graphical LASSO (FGL)* (Costantini & Epskamp, 2017; Danaher et al., 2014). I'll describe each in turn, as well as how—in the context of the current study—I used each method as a means of detecting moderator effects. Importantly, the performance of these methods is compared with that of various MNMs on metrics described in the following sections.

4.1.1 Network Comparison Test (NCT)

The goal of the NCT is to take two dataset and compute separate, unmoderated network models for each, and then use permutation tests to compare each edge parameter across the two models

and determine which (if any) are reliably different. Given that the NCT requires the comparison of two datasets—or groups—in cases where a candidate moderator is a continuous variable, the dataset may be artificially split at the median of that variable to create groups based on 'high' and 'low' values. If the moderator is a binary variable, however, the dataset may instead be split based on the pre-defined groups. The NCT is implemented in the R package `NetworkComparisonTest` (van Borkulo et al., 2017).

In the first step of the NCT, a model is estimated for each dataset and the absolute values of the differences between corresponding edges are taken as test statistics. In the second step, a sampling distribution under the null hypothesis of no difference is generated for each test statistic by randomly permuting the group membership data points B times, re-estimating the two models, and computing the absolute value of all edge differences. This affords significance tests for each edge parameter to indicate the probability of obtaining the observed differences given the null hypothesis that the parameters do not differ across datasets. Thus, this can be taken as an indirect way of testing whether or not the grouping variable significantly moderates particular pairwise relationships, as in the case of moderated network models.

For estimation, the NCT uses the GLASSO (Friedman et al., 2008), and to select the value of the penalty parameter λ, the defaults to using the EBIC with the hyperparameter $\gamma = .5$. For the current study, I used $B = 1000$ iterations of the permutation tests, along with a significance threshold of $\alpha = .05$. Then, for each permutation test that produced a test statistic with $p < \alpha$, I marked that a significant interaction had been detected for that edge.

4.1.2 Fused Graphical LASSO (FGL)

The FGL produces a similar type of result as the NCT although through a different means. Broadly, the FGL jointly estimates two GGMs by using two ℓ_1 penalties, and starts from the same position as the NCT, where two separate datasets are taken as input. The first penalty, λ_1, includes all parameters of the model (i.e., from two inverse-covariance matrices), while the second penalty, λ_2, weights the difference between the two matrices, thereby penalizing the degree to which the

corresponding GGMs are allowed to differ. Danaher et al. (2014) provide a full description of the FGL, while the method has been implemented for network models by the `EstimateGroupNetwork` R package (Costantini & Epskamp, 2017).

For the current study, the penalty parameters are selected through the same means as in the NCT, where the EBIC is used with the hyperparameter $\gamma = .5$. This is often the default selection approach in the psychological network literature, and was therefore chosen to mimic common approaches to approximating moderator effects. Given the penalty for the parameter differences λ_2, the FGL only returns different values for edges that are deemed sufficiently different by the selection procedure. Therefore, any edges that differed across the two GGMs were taken to reflect a significant interaction in the present study.

4.2 Data Generation

Due to the inclusion of moderation effects, data could not be directly sampled from multivariate distributions. To work around this, I created a Gibbs-sampling function to generate data from moderated network models. Gibbs samplers utilize a Markov-chain Monte Carlo (MCMC) approach to sampling in order to generate data from a potentially undefined multivariate probability distribution. This allows us to define network structures that incorporate interaction effects, and then simulate data according to a variety of specifications. General information about Gibbs samplers can be found in Casella & George (1992), although the general approach I took to approximating joint distributions for the models was to start with a network structure, along with a pattern of interactions, and then generate data from the corresponding conditional distributions. In addition to the study parameters described in the section below, detail about the specific procedure used can be found in Appendix A.

4.2.1 Study Parameters

There were a number of general parameters that were fixed for this particular study:

1. Two different network sizes were used: $p = 5$ and $p = 10$. Moreover, in each case an additional variable—an exogenous moderator—was generated, making the total variables in each case 6 and 11.

2. In all cases, the sparsity of the pairwise network was fixed at .5. This means that when $p = 5$ there will be 5 non-zero edges out of 10 possible edges, and when $p = 10$ there will be $22/45$ non-zero edges.

3. No main effects of the moderator variables were included. This means that all coefficients for the effects of the moderator on any variable were fixed to zero.

4. There were 4 different sparsity levels set for the interaction effects: $m \in \{.1,\ .2,\ .3,\ .5\}$. Interaction effects were specified in a similar way to pairwise effects, meaning that they were always symmetric, and filled the off-diagonal of a matrix the same size as the associated pairwise network. Thus, a sparsity level of .1 meant that there was only 1 non-zero interaction out of 10 possible interactions in a 5-node network; 2/10 non-zero interactions when the sparsity level was .2, and so on.

5. For all parameters—interactions and pairwise effects—all non-zero values were fixed at either .2 or $-.2$. On average, half of the values were positive and half negative. The reason for this was to ensure that all values would be the same size, thereby eliminating possible confounding effects of coefficient magnitude. Moreover, this value had to be large enough to be detectable, but also small enough so that data could be feasibly sampled when a number of interactions are involved.

6. All intercepts were set to zero in all models.

7. For each dataset, a random value was drawn from $\mathcal{N}(0,1)$ to be used as the mean for the moderator variable μ_m. Then, for each iteration of the Gibbs sampler, a random value was drawn from $\mathcal{N}(\mu_m,1)$. This variable was sampled differently from the other variables, given that its values were never dependent on the other data.

8. For each iteration of the Gibbs sampler, values for each variable were drawn from a normal distribution with $\sigma^2 = 1$, and a mean that would be defined by the parameter values along with preceding draws.

9. There were 5 different sample sizes that were simulated: $N \in \{50,\ 100,\ 250,\ 500,\ 1000\}$.

10. For all moderated networks (i.e., all models except the NCT and FGL), p-value thresholding was applied at $\alpha = .05$, and the AND rule was used to aggregate model parameters. Interaction effects were aggregated in the same way as pairwise effects, where the final values are averages of each pair of corresponding nodewise parameters whenever $p < \alpha$ for both coefficients. Otherwise, the final value is fixed to zero.

Given these specifications, there were 40 conditions in total. For each condition, 100 models and datasets were generated, producing 4000 datasets—and moderated networks—altogether.

To begin the data generation process, a network would be generated based on a set of simulation parameters—for instance, when $p = 5$, a 5×5 matrix of zeros was created and 5 elements in the off-diagonal (given that pairwise sparsity was set to .5) would randomly be set to either positive or negative .2 with equal probability. Then, a 5×5 interaction matrix would be created, and the above procedure would be applied in accordance with the sparsity parameter $m \in \{.1,\ .2,\ .3,\ .5\}$. In all cases, the two matrices were symmetric, although never identical. Together, the two matrices comprised the input for generating a single dataset from the Gibbs sampler.

To give a broad sense of the data that were generated, we can see in Table 4.1 the total proportion of variables that were not normally distributed, as well the proportion of datasets that violated multivariate normality. The former consideration is not actually an assumption of the GGM, but the latter is. We can see from this table that the tests deemed about 25–45% of the datasets to be non-normal. I chose to report proportions for both unadjusted and Bonferroni-adjusted p-values in order to show less conservative and more conservative estimates of the violation rates.

This is a positive aspect of the study, as it means that this isn't necessarily an over-idealized case, allowing us to see how the models perform in a relatively 'messy' situation (at least compared

Normality Violation Rates					
		Continuous		Ordinal	
Test	Type	Unadjusted	Bonferroni	Unadjusted	Bonferroni
Shapiro	Univariate	23.3%	15.7%	100%	100%
Henze-Zirkler	Multivariate	46.1%	26.1%	98.9%	89.7%
Royston	Multivariate	45.8%	29.0%	100%	100%
Doornik-Hansen	Multivariate	36.4%	22.4%	98.1%	85.4%

Table 4.1: Percentage of variables and datasets judged to be *non-normal* based on outputs with $p < .05$ from Shapiro tests (assessing univariate normality for each simulated variable), as well as three tests of multivariate normality. For the latter, the tests produced one test statistic for each of 4000 datasets, and for the former there was one for each of 34,000 total variables. The Bonferroni correction was applied at a family-wise level, meaning that p-values were corrected within each dataset for the univariate tests, and across datasets for the multivariate tests. The `MVN` package in R was used to conduct all analyses (Korkmaz et al., 2014).

to the case where all data are multivariate normal). It is important to note, however, that about half of the violations occurred in the $m = .5$ condition, which only comprises a quarter of the data, and contains the largest proportion of interactions. When this condition is excluded, the values drop closer to a range of 16–26%. This was also a motivating factor in performing the study a second time after 'ordinalizing' the datasets, which, as shown in Table 4.1, dramatically increased the prevalence of non-normal distributions. Discussion of the results and plots for the ordinal version of the study are provided in Appendix E. Still, it shows that the data were not overly idealized and still contained plenty of variability as well as violations of model assumptions.

Lastly, a variation on this study was also conducted, wherein the same data were all converted from continuous variables to ordinal variables with 4 levels. The reason for this was to mimic a common practice in psychological research, wherein Likert scales or a ranking system is used to measure variables. Moreover, violations of normality may be more common in empirical data than in data simulated directly from normal distributions, and so we see in Table 4.1 that the ordinal versions of the simulated variables tended to have substantially higher rates of non-normality. The results of this study were similar to what was observed in the continuous-variable setting, and so

is provided separately (along with details about the 'ordinalizing' process) in Appendix E.

4.2.2 Models Investigated

Once all data had been generated, a number of models were fit to each dataset. All models utilized the moderated network procedures that have been described thus far, except for the models fit with the NCT and the FGL. The first model in Table 4.2 is the saturated network, where all possible interactions are estimated along with all pairwise relationships, although pruning via p-value thresholding and the AND rule were applied to obtain a sparse model (this procedure was applied to all MNMs).

The 'Two-Stage' designation refers to variable selection via the hierarchical LASSO where the `glinternet` algorithms are first applied to select the active set of predictors, and then the final models are fit separately using OLS. Moreover, the difference between CVmin and CV1se is that in the former we take the λ that minimizes the average prediction error, whereas in the latter we take the λ 1 standard error above the former (in terms of prediction error). Then, all three resampling-based selection techniques were applied with $B = 100$ iterations, and the AIC to select the λ penalty. The reason for using the AIC in these cases is because I expect the resampling techniques to be relatively conservative, and so I wanted to combine these with a less conservative selection criterion.

4.3 Performance Measures

The primary goal of this study was to evaluate how well these different methods perform in terms of detecting interaction effects. It should be noted that the two sample-split methods (i.e., the NCT and FGL) do not estimate the actual values of interactions, and also estimate very different pairwise networks than in the other methods (as a result of the median-split moderator). Thus, parameter accuracy could not be assessed across all models, and the priority was simply to focus on how well the different methods perform in terms of detecting interactions. However, parameter accuracy

Name	Model	Type	λ Selection
Prune	Saturated	MNM	-
AIC	Two-Stage	MNM	AIC
BIC	Two-Stage	MNM	BIC
EBIC25	Two-Stage	MNM	EBIC ($\gamma = .25$)
EBIC50	Two-Stage	MNM	EBIC ($\gamma = .5$)
CVmin	Two-Stage	MNM	10-fold CV_{min}
CV1se	Two-Stage	MNM	10-fold CV_{1SE}
split	Multi-Sample Split	MNM	AIC ($B = 100$)
bootstrap	Bootstrapping	MNM	AIC ($B = 100$)
stability	Stability Selection	MNM	AIC ($B = 100$)
NCT	NCT	Sample-split	EBIC ($\gamma = .5$)
FGL	FGL	Sample-split	EBIC ($\gamma = .5$)

Table 4.2: Models fit to each simulated dataset, along with the names used in plots.

was investigated among the moderated network models (MNMs) in a secondary analysis to test the hypothesis that the resampling methods will more closely approximate the true network structures than the two-stage variable selection methods, which will in turn outperform the saturated model after p-value pruning.

4.3.1 Primary Analysis

The central metrics used as outcomes were: Sensitivity, the false-discovery rate (FDR), and Matthew's correlation coefficient (MCC). *Sensitivity* refers to the true positive rate, and is calculated as the number of correctly detected interactions (true positives, TP) divided by the number of true positives plus the number of false negatives (FN; the number of undetected interactions). The *false-discovery rate (FDR)* refers to the proportion of false positives out of the total number of interactions detected (i.e., the sum of true and false positives). These can be written as:

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad \text{and} \quad \text{FDR} = \frac{\text{FP}}{\text{FP} + \text{TP}}. \tag{4.1}$$

A true positive (TP) occurs when there is a non-zero interaction in the original network that is also estimated to be non-zero in the model; a false positive (FP) occurs when an interaction is estimated to be non-zero when it is actually zero in the true model; and then a false negative (FN) occurs when an interaction effect is estimated as zero in the model, but was actually non-zero in the true network. Importantly, for each moderated network model these metrics were computed for the final, aggregated interaction effects *after* the AND rule and p-value threshold had been applied.

The third metric—*Matthew's correlation coefficient (MCC)*—is a more general performance measure that takes into account all aspects of binary classification (Powers, 2011). That is, we can interpret the MCC as a metric that captures how well the different methods perform *overall* in terms of correctly detecting interactions. The MCC is defined as:

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP}+\text{FP})(\text{TP}+\text{FN})(\text{TN}+\text{FP})(\text{TN}+\text{FN})}}. \tag{4.2}$$

As with the other measures, MCC can take on values between -1 and 1.

Additionally, *specificity* (the true negative rate) was calculated and the results are provided in Appendix B. Across conditions, most all of the models performed exceptionally well on this metric, and so the results will not be discussed here.

4.3.2 Secondary Analysis

In this analysis I turn to investigating the accuracy of parameter estimates. The sample-split methods are not included in this analysis, but we can still evaluate how the different MNMs perform across the tested conditions. To do this, I use two metrics: the Pearson correlation, and mean absolute error (MAE).

For calculating the correlation, I simply correlate the network structure estimated by each model with the true network from which the data were simulated. Then, this is done separately for the matrix encoding the pattern of interactions. This allows us to look at how well the different methods approximate both the pairwise network, as well as the nature of the interaction effects.

While the correlation allows us to look at how accurate parameter estimates are, the MAE allows us to look at how biased they are. To calculate this value, I take the mean of the absolute values of the differences between each estimate and the corresponding true parameter value.

4.3.3 Overview

In sum, the metrics described above are utilized to address the following research questions:

1. Primary analysis:
    (a) When interactions are present, how well do these methods detect them? (Sensitivity)
    (b) To what extent do these methods detect spurious interactions? (FDR)
    (c) What is the overall quality of their ability to correctly detect interactions? (MCC)
2. Secondary analysis:
    (a) Among the MNMs, how accurate are the parameter estimates? (Correlation)
    (b) Among the MNMs, to what extent are the parameter estimates biased? (MAE)

4.4 Results

Here I will discuss the results of the study, primarily through visual analysis of the patterns observed in the figures.[1] It is worth restating that for all of the primary analyses, the only aspects of the simulation being investigated are the interaction-detection capabilities of the different models; the actual parameter estimates themselves are disregarded. Further investigation of this component is reserved for the secondary analysis, wherein the objective is to focus on the accuracy of the parameters.

Additionally, after these two analyses I also conducted a separate correlation stability analysis using the case-dropping bootstrap on each simulated dataset. The goal of this was to assess how the different conditions affected the *CS* coefficients for each of the three metrics described in the previous chapter—namely, these look at the stability of the edge weights, node strength centralities,

[1] A variety of hypothesis tests were also conducted, but given the large number of simulated datasets almost all comparisons were significant in all cases. In fact, the most interesting results were the comparisons that did *not* reach significance, and so these are mentioned in the text.

and expected influence estimates. For these analyses, the only model that was tested was the saturated MNM, because the goal was simply to see what coefficients are obtained in the most basic moderated network scenario.

4.4.1 Sensitivity

To start, we can think of sensitivity as a measure of how conservative a method is. That is, methods that perform 'worse' on sensitivity simply reflect those that are most conservative, while those that perform 'best' are the least conservative. Overall, higher values of sensitivity are desirable—we want to find interactions when they are truly there—but, as we'll see in the other analyses, high sensitivity may be less useful when it comes at the cost of low precision.

As shown in Figure 4.1, there are some general trends that are immediately apparent for all models. First, across conditions we see that sensitivity increases with sample size. This is to be expected of a method that reliably detects interactions, and reflects positively on all of those evaluated. The only minor exception to this trend is the FGL, which exhibits average sensitivity greater than .99 across all conditions and sample sizes.

The next trend we observe is that sensitivity is typically higher when there are fewer parameters. This is especially apparent at smaller sample sizes ($N \leq 250$), wherein sensitivity tends to be higher when $p = 5$ rather than $p = 10$. The same trend is seen to a lesser extent for the sparsity of the true pattern of interactions, wherein sensitivity tends to be slightly higher when there are fewer true interactions (i.e., when $m = .1$ over $m = .5$). However, while average estimates tend to be higher with fewer parameters, they also tend to have greater variability. At larger sample sizes these differences diminish as we see that most models converge to the upper limit, particularly when $N = 1000$.

When comparing sensitivity across models, we can see that when $N \leq 500$ there are certain methods that stand out as performing better or worse than most others. The FGL, however, always performs best, and shows a ceiling effect across all sample sizes. Excluding the FGL, the methods that consistently demonstrate the highest sensitivity are the AIC, CVmin, and bootstrap models.

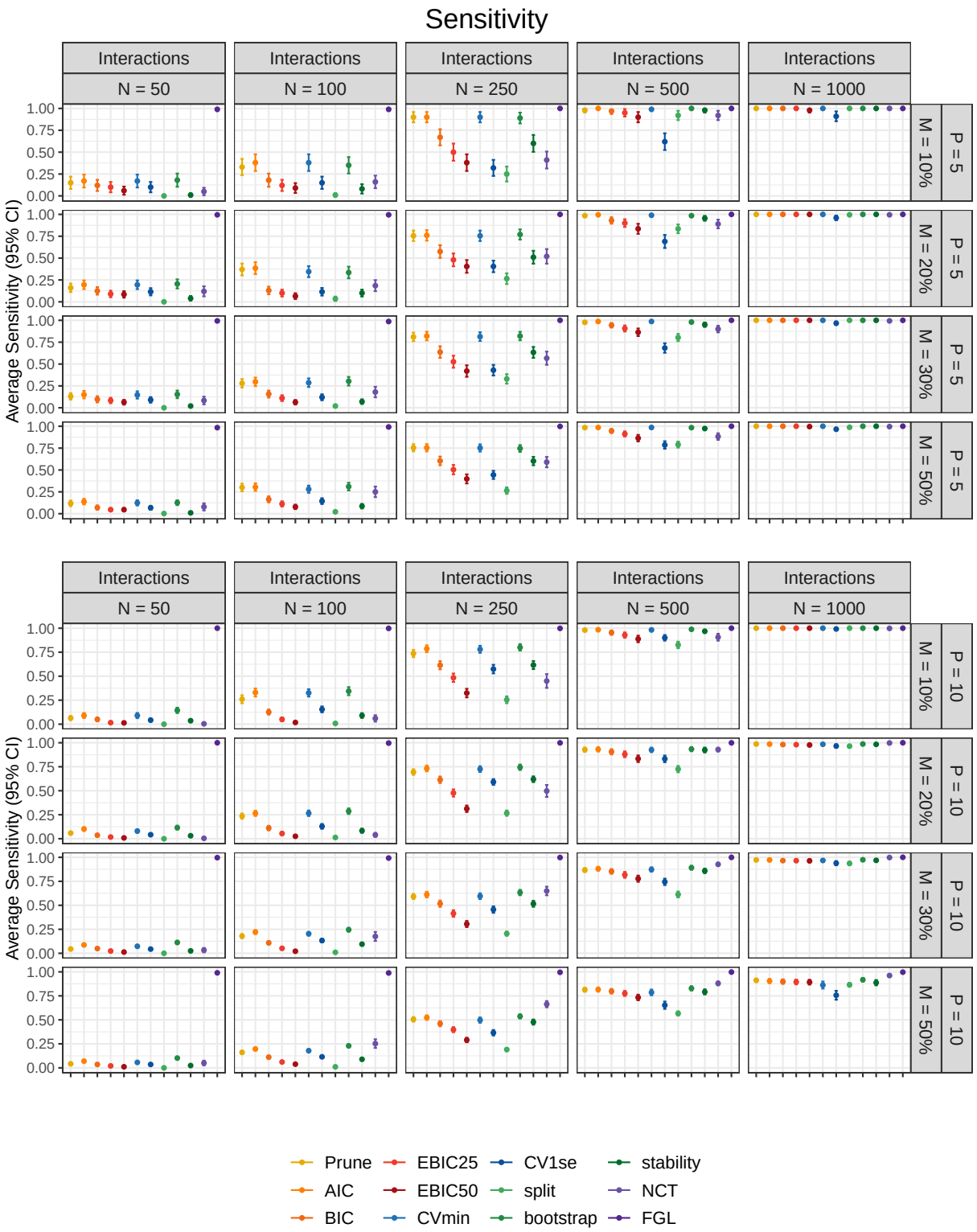

Figure 4.1: Average sensitivity across models and conditions. Each plotted point is based on 100 simulated datasets.

Also, the pruned saturated model, somewhat surprisingly, performs quite well at most sample sizes, although is always slightly lower than the former three. The first two models use the two-stage variable selection process described in Chapter 3 (where the AIC or 10-fold CV error are minimized, respectively), while the third uses the bootstrapping variant of Algorithm 1, along with the AIC to select λ at each of the $B = 100$ iterations.

As for the worst performing methods, the multi-sample split method has the lowest sensitivity for $N \leq 250$ when $p = 5$, and for $N \leq 500$ when $p = 10$. For the converses of those sample sizes, we see that the CV1se has the lowest sensitivity. Beyond that, we see that the EBIC50, the stability selection method, and the NCT perform particularly poorly on this metric at lower sample sizes.

To summarize, the FGL method stands out as having the highest sensitivity across conditions, and the NCT appears to be relatively low (although performs slightly better than some of the MNM methods at smaller sample sizes). While average sensitivity for the NCT does not exceed .6 until $N = 500$, we see that the better performing MNMs reach or exceed .8 when $N = 250$.

4.4.2 False-Discovery Rate (FDR)

Next we have the FDR, which is the inverse of precision. That is, the FDR tells us to what extent a method is susceptible to returning false positives. These results are shown in Figure 4.2, and at coarse-grained level appear to reflect the inverse of the results from Figure 4.1, although there are some key deviations to be discussed. In general, an ideal method is one with a relatively low FDR and relatively high sensitivity; we want a method that reflects a reasonable balance between these two metrics.

The first major pattern that stands out in Figure 4.2 is how the FDR for the FGL changes as sample size increases. We see that while the FGL has the lowest FDR when $N = 50$, this value increasing substantially as sample size increases. This indicates that the FGL is not a consistent estimator of interaction effects (at least not of the type investigated in this study), and that it is inappropriate to be used for this purpose. At the very least, even if a particular method has a relatively high FDR, we should expect that this will improve as we increase the sample size. In

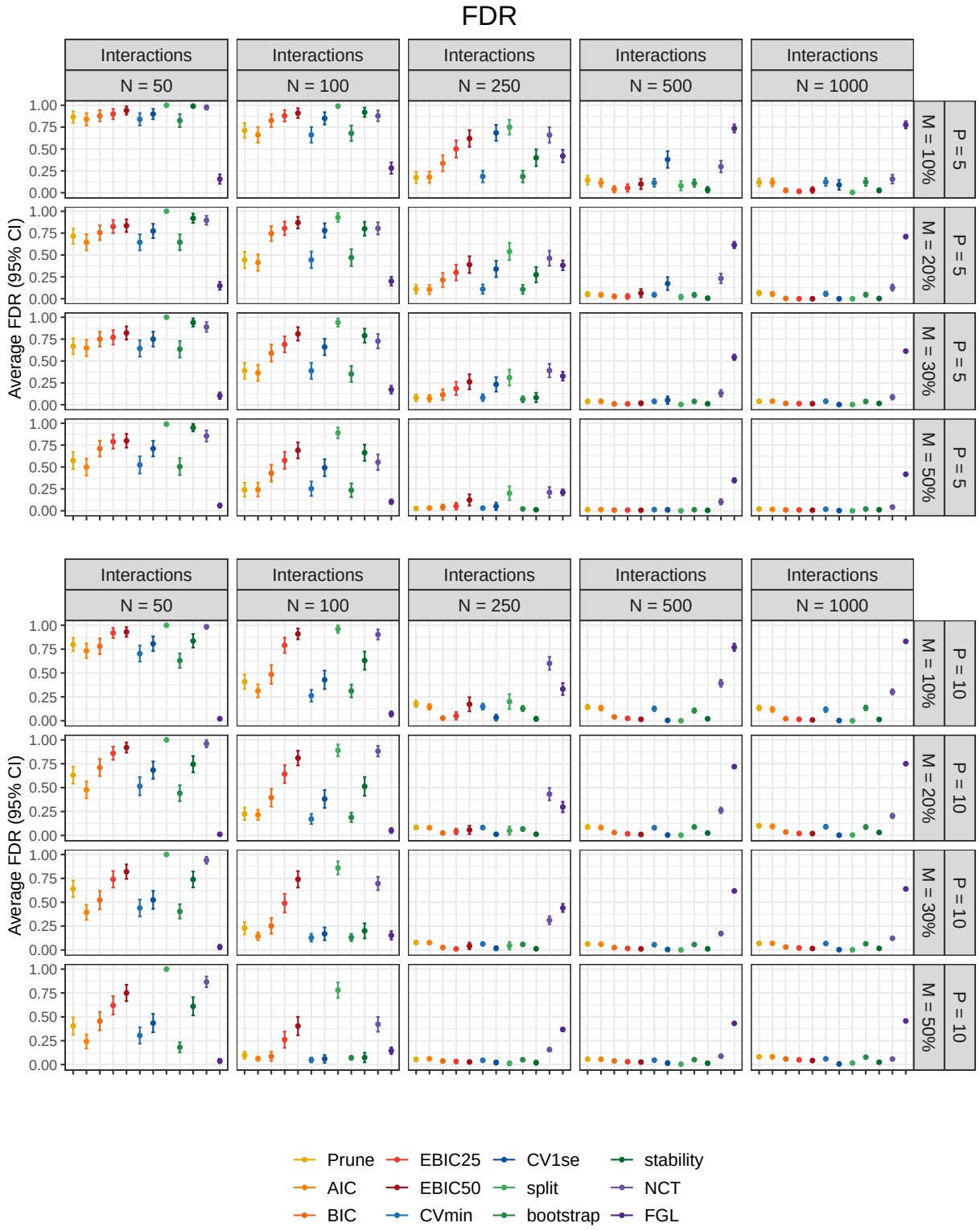

Figure 4.2: Average false-discovery rate across models and conditions. Each plotted point is based on 100 simulated datasets.

sum, the FGL returns a substantial number of false positives at larger sample sizes, and therefore should not be used to explore or test moderated network models.

For the other sample-split method—the NCT—we see that it does not suffer from the same issue as the FGL. However, we can clearly see that it performs relatively poorly in all conditions. For both $p = 5$ and $p = 10$, and across levels of m, the NCT always has either the second or first highest FDR for $N \geq 250$. For the two smaller sample sizes, we see that split and stability (two of the resampling-based MNMs) have slightly higher FDRs than the NCT, but this relation is reversed for larger samples. Lastly, while we can see that the FDR for the NCT shrinks as sample size increases, it does not always converge to the lower bound even when $N = 1000$, and the only instance in which the mean is lower than .05 is when $N = 1000$ and $m = .5$. Conversely, most of the MNMs converge at either $N = 250$ or $N = 500$, and nearly all have converged to zero by $N = 1000$, with a few exceptions that only converge to .05 when $m = .1$.

Among the MNMs, we can see that at smaller sample sizes the split, stability, EBIC50, and CV1se methods have the highest FDRs, while the saturated model, AIC, CVmin, and bootstrap methods all have the lowest. We also see that average FDRs tend to be higher when there is a smaller proportion of true interaction effects, as is to be expected. In general, though, the MNMs perform fairly well, although the EBIC50 and multi-sample split method don't show acceptable performance until $N = 250$. Moreover, all methods have relatively high FDRs when $N = 50$, indicating that this sample size is likely too small to effectively control false positives. $N = 100$ is we when we really see differences among the methods, where the AIC, CVmin, and bootstrap models (along with the pruned model) stand out as performing much better than the others.

4.4.3 Matthew's Correlation Coefficient (MCC)

Finally, the average MCCs are displayed in Figure 4.3. There is nothing surprising here, given the results for the preceding metrics, although it provides a clear summary of our conclusions regarding the quality of the different methods. For the MCC, values closer to 1 are indicative of superior performance. Moreover, we should expect consistent estimators of interaction effects to

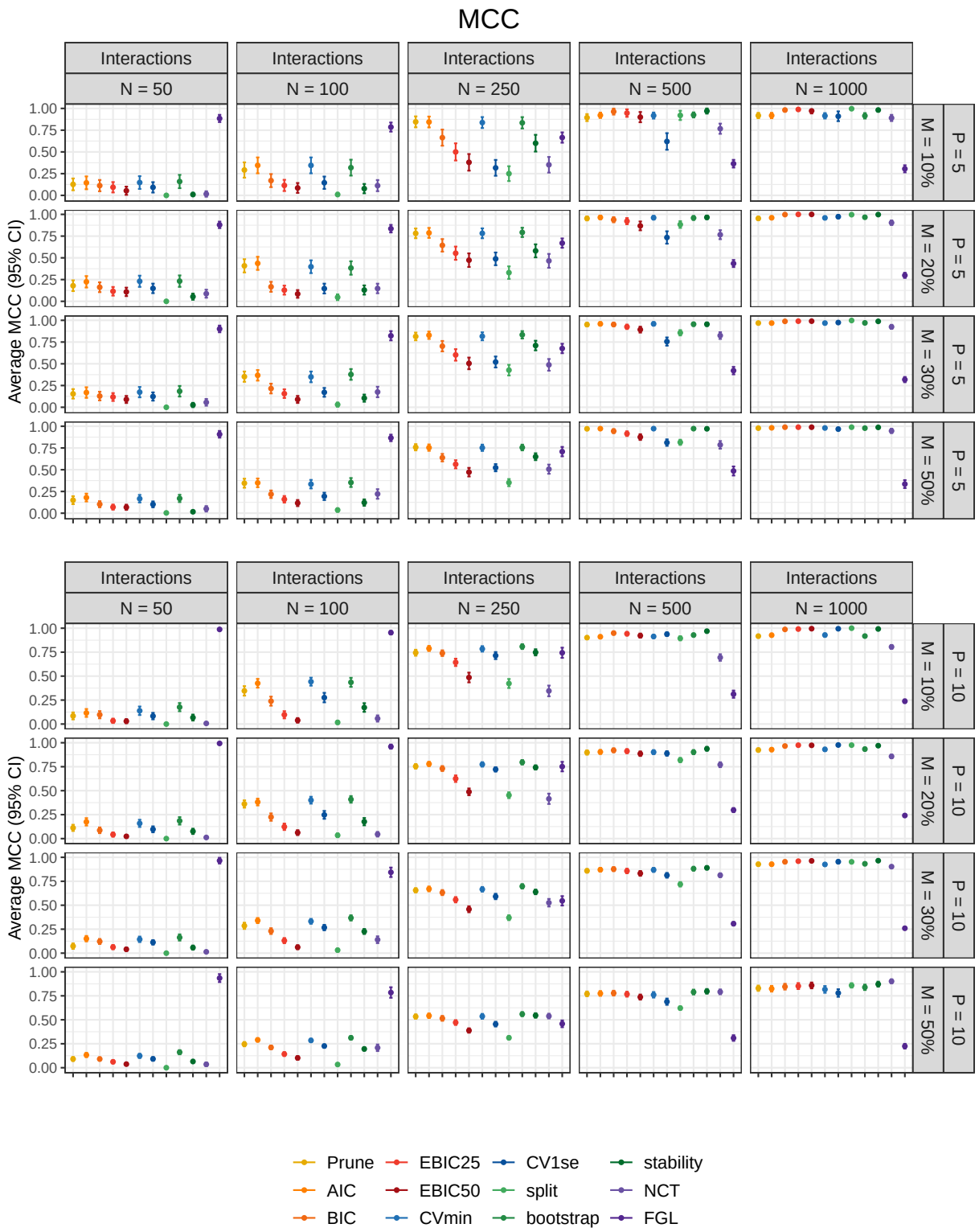

Figure 4.3: Average MCC across models and conditions. Each plotted point is based on 100 simulated datasets.

demonstrate performance that improves with increases in sample size.

As we saw with the FDR, the FGL appears to perform extremely well at small sample sizes, but then showing worsening performance as sample size increases. Again, this indicates that the FGL should not be considered an acceptable methods for detecting interaction effects within the present context. With the NCT, we also see a similar pattern in that it exhibits a higher average MCC as the sample size increases, but its performance tends to be lower than most all of the MNMs across conditions. This difference is especially pronounced for the larger networks ($p = 10$), as well as at larger sample sizes ($N \geq 500$).

The methods that consistently perform the best on this metric are the same as for the other metrics: the two-stage model with either AIC or CV_{min} selection, and the bootstrapping method with AIC selection. Also, the stability selection method has the highest average MCC at larger samples ($N \geq 500$), although has relatively poor performance in smaller samples.

To summarize, we can see from the primary analysis that all of the MNMs are consistent estimators of interactions, and on the whole outperform both of the sample-split methods. We also saw that while the NCT appears to be a consistent estimator, the FGL is not and therefore should be avoided in this context.

Lastly, we saw that there was a lot of variability in the performance of the resampling-based MNMs. In general, the multi-sample split and stability selection algorithms performed poorly at small sample sizes, but showed excellent performance in large samples. The bootstrapping method, however, had the best performance in comparison with all other methods. Unreported significance tests revealed that while this method produced the best results, these were not significantly better than those from the two-stage methods fit with either AIC or CV_{min} λ selection.

4.4.4 Correlation

Results are displayed in Figure 4.4, and a table containing the grand means is provided in Appendix D. In general, we see that all methods produce higher average correlations for both pairwise and interaction parameters as sample size increases. These values tend to be higher for smaller networks

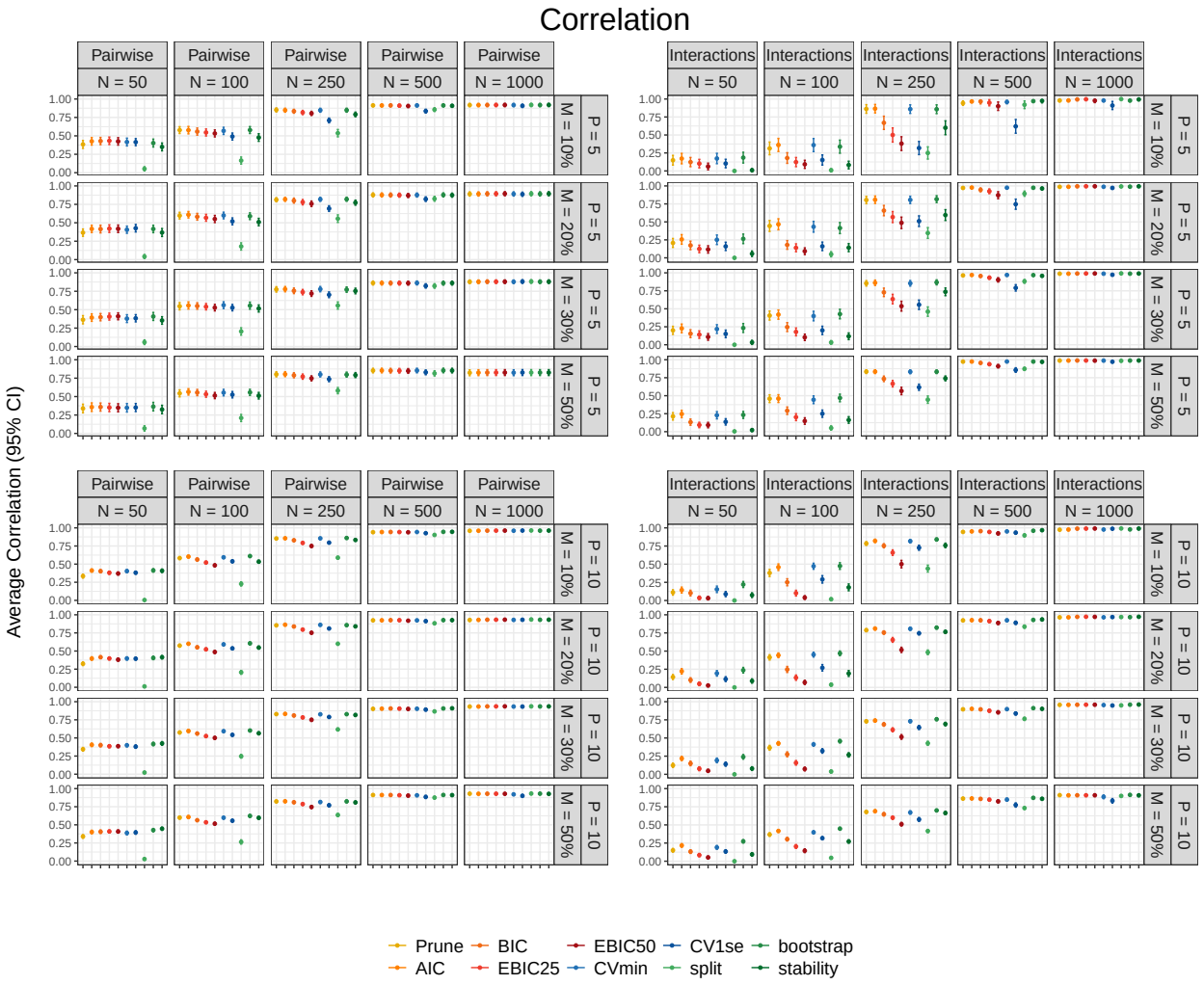

Figure 4.4: Average correlations across MNMs and conditions. Each plotted point is based on 100 simulated datasets.

($p = 5$) and lower for larger networks ($p = 10$). Also, while correlations do not significantly differ across levels of m (interaction sparsity) for pairwise effects, we see that they tend to be slightly higher for interactions at lower values of m (i.e., when true interactions are sparse).

Overall, all methods perform quite well at estimating true parameter values for $N \geq 500$. The multi-sample split and two-stage CV_{1se} methods showed the worst performance across conditions, where the split method produced the lowest average correlation for both pairwise and interaction effects when $N \leq 250$, as well as the lowest when both $N \geq 500$ and $p = 10$; conversely, the CV1se method had the lowest values when $N \geq 500$ and $p = 5$. Collapsing across network sizes and interaction sparsity, at $N = 500$ all methods produced an average pairwise correlation $\geq .90$ and an average interaction correlation $\geq .80$, with the exception of the split method which returned average values of .80 and .70, respectively.

The methods that consistently performed the best on this metric—for both pairwise and interaction effects—were the two-stage models with either AIC or CV_{min} selection, and the bootstrapping method with AIC selection. In terms of magnitude, the bootstrapping method produced the highest average correlations with the true models, although not significantly higher than that for the AIC and CVmin methods. For these three methods—as well as the pruned, saturated model—average pairwise correlations exceeded .81, and interaction correlations exceeded .71 when $N = 250$.

4.4.5 Bias (MAE)

The degree to which each method produced biased parameter estimates is depicted in Figure 4.5. Overall, the patterns here generally reflect the inverse of the patterns seen in Figure 4.4. That is, methods that produced the largest correlations also tended to produce the lowest average MAE. One difference between the patterns seen for these two metrics is that for MAE, smaller values were obtained for interaction effects than pairwise effects. This is particularly true at lower levels of m (i.e., when there were fewer true interactions to estimate), while at higher values ($m = .5$) average MAE was comparable for both. This is simply a product of the fact that there is a lower likelihood of obtaining unbiased parameter estimates with a larger number of non-zero effects.

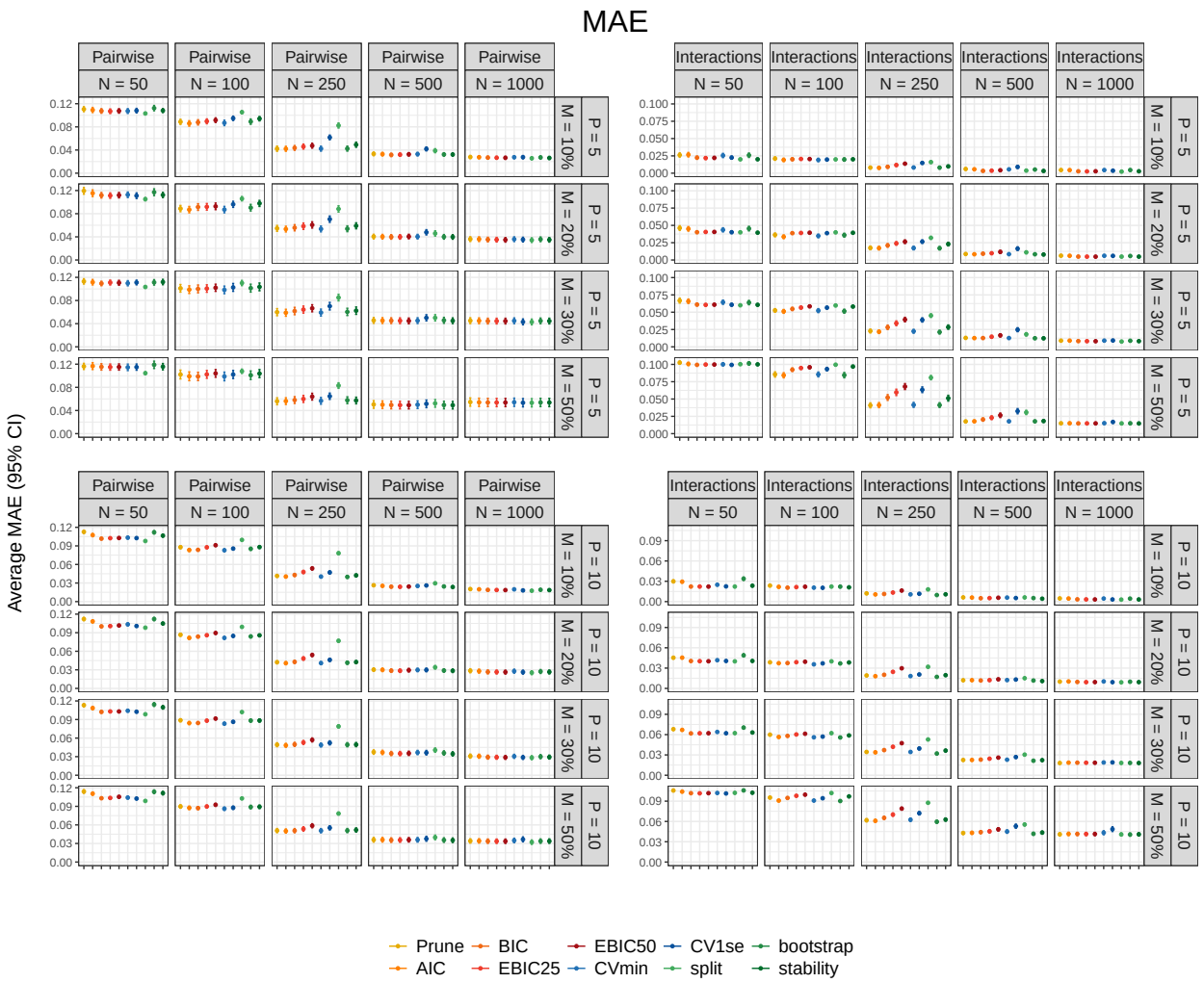

Figure 4.5: Average MAE across MNMs and conditions. Each plotted point is based on 101 simulated datasets.

Beyond this, our interpretation of the MAE is similar to what we saw for the correlations, as it identifies the same high-performing and low-performing methods. The only minor difference is that the CVmin method had slightly lower average MAE than the bootstrap method; the same was true for the AIC method with relation to the bootstrap, but only for pairwise effects (these are displayed in Appendix D). However, as was the case for previous metrics, these three methods did not produce significantly different means.

4.5 Correlation Stability Analysis

This is a separate analysis from those described thus far, although the same datasets have been used. Specifically, a correlation stability analysis was conducted on each dataset using the case-dropping bootstrap with $B = 1000$ iterations across 10 subsample sizes ranging from 5–75% drops. We can see the distributions of these statistics for each condition in Figure 4.6, as well as some descriptive statistics in Table 4.3.

As is to be expected, we see that correlation stability increased with sample size across all metrics and conditions. Also, the *CS* values associated with pairwise terms were always higher, on average, compared to those associated with interaction terms. This not surprising, as many researchers have noted that interaction terms are typically subject to more variability than pairwise terms (e.g., Brambor et al., 2006; Friedrich, 1982). Additionally, we tend to see greater variability for smaller networks ($p = 5$), as well as more stable interaction estimates with a greater number of interactions (i.e., larger values of m).

Looking at Table 4.3, I was interested in seeing at which sample sizes values began to exceed .25 and .50 for any given metric and condition, given that these reflect rules-of-thumb cutoffs for identifying acceptable and excellent coefficient values, respectively (Epskamp et al., 2018a). In each case, summary statistics were computed across levels of m. For edge weight stability, we see that all *CS* values for pairwise and interaction terms exceeded .25 for $N \geq 100$. Moreover, all pairwise terms exceeded .5 for $N \geq 250$, and the same is true for interaction terms when $N \geq 500$. This further reflects the idea that interaction terms tend to be less stable than pairwise terms, although it also shows that stable estimates for interaction terms can be attained with larger sample sizes.

For strength centrality, however, these values were smaller across the board. Average *CS* coefficients for pairwise terms exceeded .25 when $N \geq 250$, and interaction terms exceeded this value when $N \geq 500$. Additionally, both terms did not produce means that exceeded .5 until $N = 1000$. This shows that strength centrality is typically more unstable than edge weights, and that this is especially true for interactions. Nevertheless, we still see that at very large sample sizes we are

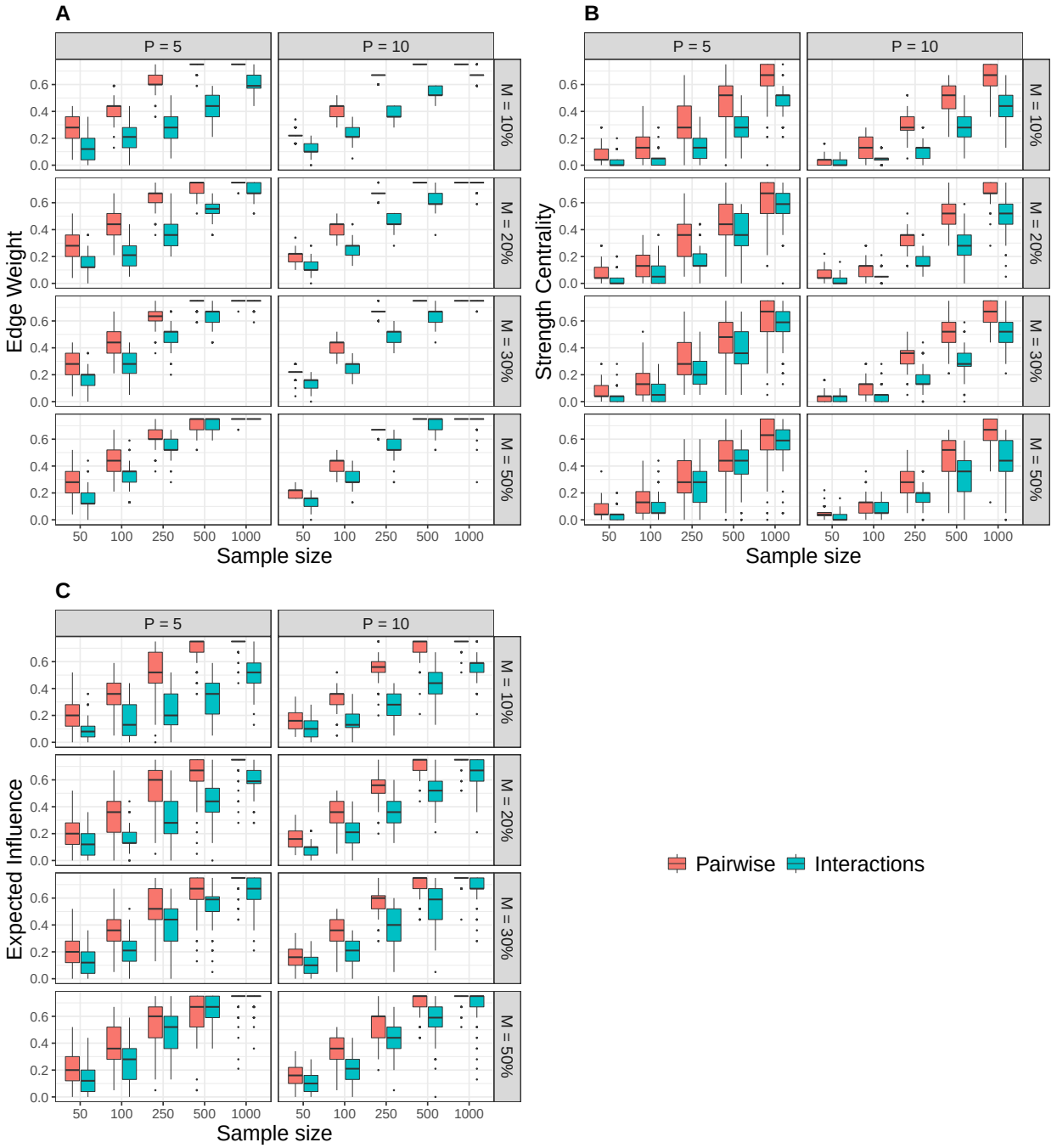

Figure 4.6: Distributions of correlation stability coefficients across conditions for the saturated MNM fit to all simulated datasets.

able to obtain stable estimates in each case. Lastly, expected influence showed very similar patterns (albeit with slightly lower means) to the analysis of edge weights. This reflects the general finding from Robinaugh et al. (2016), wherein it was concluded that expected influence is typically the most reliable centrality measure within psychological networks. The present analysis extends this conclusions by also showing that the same is true for interaction terms.

4.6 Summary

In addition to the plots shown on the previous pages, I have provided the grand means (collapsing across all conditions) for the primary analysis in Table C.1, along with tables containing all information from the plots in Appendix C. Also, these same tables for the secondary analysis can be found in Appendix D. Overall, we see from this analysis that the moderated network framework generally performed quite well with simulated data, and that the variable selection techniques can play a potentially valuable role in improving the accuracy of parameters and reducing the rate of false positive associations. We also saw that the sample-split methods—which have been used as an alternative to moderated networks—perform much worse than the true MNMs. Namely, the NCT tended to produce outputs that had the lowest or near-lowest values in comparison with the other models, and the FGL was shown to be an inconsistent estimator as well as one that should be thought of as inappropriate in this context.

4.6.1 Limitations

There are of course a number of limitations to this study, where one particularly unrealistic component was that all moderator variables were generated to be independent of the other variables in each model. Moreover, the main effects of the moderators were always fixed to zero, along with the intercepts for the other variables. It's hard to imagine real-world scenarios where these assumptions would hold, and so additional studies must be conducted to vary these and other parameters in order to more comprehensively address the questions posed in this chapter.

Correlation Stability $CS(\rho_{.95} \geq .70)$

Edge Weights

	$p = 5$						$p = 10$					
	Pairwise			Interactions			Pairwise			Interactions		
N	*M*	*SE*	*Mdn*	*M*	*SE*	*Mdn*	*M*	*SE*	*Mdn*	*M*	*SE*	*Mdn*
50	.26	.005	.28	.15	.005	.12	.22	.002	.22	.13	.003	.10
100	.43	.005	.44	.25	.005	.28	.42	.002	.44	.26	.003	.28
250	.62	.004	.67	.43	.007	.44	.66	.001	.67	.47	.004	.44
500	.73	.002	.75	.58	.006	.59	.75	.000	.75	.62	.004	.59
1000	.75	.000	.75	.70	.004	.75	.75	.000	.75	.72	.003	.75

Strength Centrality

	$p = 5$						$p = 10$					
	Pairwise			Interactions			Pairwise			Interactions		
N	*M*	*SE*	*Mdn*	*M*	*SE*	*Mdn*	*M*	*SE*	*Mdn*	*M*	*SE*	*Mdn*
50	.08	.004	.04	.03	.002	.04	.05	.002	.04	.02	.001	.10
100	.15	.005	.13	.07	.004	.05	.12	.004	.13	.06	.002	.05
250	.31	.007	.28	.19	.006	.20	.31	.005	.28	.16	.004	.13
500	.45	.008	.52	.36	.008	.36	.49	.000	.52	.31	.006	.28
1000	.61	.000	.67	.70	.007	.59	.65	.000	.67	.48	.007	.52

Expected Influence

	$p = 5$						$p = 10$					
	Pairwise			Interactions			Pairwise			Interactions		
N	*M*	*SE*	*Mdn*	*M*	*SE*	*Mdn*	*M*	*SE*	*Mdn*	*M*	*SE*	*Mdn*
50	.21	.006	.20	.12	.005	.12	.16	.004	.16	.10	.003	.10
100	.36	.007	.36	.20	.007	.21	.33	.005	.36	.19	.005	.21
250	.54	.008	.60	.36	.009	.36	.55	.005	.60	.37	.007	.36
500	.65	.007	.67	.49	.009	.52	.70	.000	.75	.50	.007	.52
1000	.72	.000	.75	.70	.007	.67	.74	.000	.75	.64	.006	.67

Table 4.3: Means, standard errors, and medians for three correlation stability coefficients at each sample size and network size. Collapsing across levels of m (interaction sparsity), each statistic is based on 400 datasets, where $B = 1000$ subsamples were evaluated in each case to obtain a single *CS* value on each metric. On average, 100 subsamples at each of 10 sizes were used, ranging from a 5% drop to a 75% drop. Low values indicate low stability, and high values indicate high stability, with .75 being the largest possible value and zero being the lowest. In all cases, a saturated MNM without p-value thresholding was fit to each subsample.

Other components such as the sparsity of the pairwise networks (which was always .5), the sizes of the parameters (which were all set to $+/-.2$), and the distributions of the error terms could be varied to better understand a wider array of cases that might be encountered in empirical settings. The briefly mentioned 'ordinalized' version of the study was conducted to extend this analysis into slightly different territory, and we did see that this analysis (i.e., with ordinal variables) generally led to lower performance across all metrics (these results are discussed and shown in Appendix E).

Thus, while this study allowed us to get an initial sense of how moderated networks function on somewhat realistic data, additional research must be done to more comprehensively evaluate these models. Next, now that we have a strong sense of how to estimate and test moderated network models in cross-sectional data, we can move to thinking about *temporal MNMs* in the following chapter.

Chapter 5

Moderators in Temporal Networks

Thus far, we have only considered how to model moderators in cross-sectional networks. However, there is a great deal of interest in applying and testing psychological network models with longitudinal data (e.g., Bringmann et al., 2013, 2017; Haslbeck & Waldorp, 2015). In order to study psychological processes as they unfold over time, researchers often use repeated sampling techniques, including longitudinal research designs and experience sampling methods. These methods allow one to investigate *within-person processes*, such as the extent to which certain variables predict the values of others (as well as themselves) at subsequent points in time. It is not possible to directly study these types of relationships in cross-sectional data, and so temporal models represent an important complement to the methods covered in previous chapters (c.f. Molenaar, 2004). Thus, here I introduce *temporal MNMs*, and focus on cases where idiographic data $(N = 1)$ are modeled with moderated temporal networks. Many of the methods discussed so far (particularly for variable selection) can be extended to temporal MNMs, although there are a number of additional factors that must be taken into consideration given the temporal dependencies in the data.

5.1 Experience Sampling Methodology

To study moment-to-moment changes in psychological and affective experience, researchers have utilized *intensive longitudinal modeling* to better understand the dynamics of daily life. For instance, participants may be sent surveys to their phones via text message multiple times per day (e.g., Nielson et al., 2015; Wild et al., 2010; Myin-Germeys et al., 2009) with questions designed to measure mood states, the experience of stress, and social interactions as they occur in real time.

These techniques are referred to as *experience sampling methods (ESM*, and are often used to measure variables within short time frames and across large time windows (e.g., weeks, or even months; Nielson et al., 2015; Wild et al., 2010; Myin-Germeys et al., 2009). The relatively recent increase in applications of these methods is due to the greater prevalence and use of 'smart' devices, such as mobile phones, watches, and other similar products. Many processes realized in daily life may be difficult to study within lab settings, and these tools therefore provide a potentially valuable window into studying psychological phenomena as they occur 'in the wild'.

Given the high sampling frequency of most ESMs, we can utilize methods from *time series analysis* to study the dynamics of mental processes, rather than simple long-term trends or changes in means. Common techniques for analyzing longitudinal data—such as latent growth curve models, the repeated measures ANOVA, and panel models—typically focus on the latter aim; e.g., assessing trends over time (for example, *do average levels of depression increase linearly or quadratically in this sample?*), or comparing differences in intercepts and slopes across groups. In contrast, time series methods allow us to investigate the *short-term dynamics* in longitudinal data. These methods tend to be accompanied by stricter assumptions and data requirements than alternative methods. In particular, a much larger number of time points are typically required even for simple analyses. Thus, access to and availability of ESM data has been crucial for the development of these models, where it is not uncommon to see data with hundreds of observations taken for single individuals.

5.2 The Building Blocks of Temporal Networks

In general, temporal networks are directed graphs with weighted edges that signify the magnitudes and directions of time-lagged relationships between pairs of nodes. These parameters are the outcomes of multivariate, multiple regression models (as is the case for cross-sectional networks), although in these situations we are not only regressing variables onto one another to understand their conditional (in)dependence structure, but we are doing so with respect to some time lag h. In many cases, a 'lag-1' model is used, where we simply regress each variable onto itself (and the

other variables of interest) at the preceding time point. Models with additional lags can also be tested (e.g., by regressing a variable onto its values at the previous 2, 3, or more time points), but it is most common to see these models restricted to only the previous time point.

The key contribution of time series methods to network modeling is that they afford a richer picture of the temporal dynamics governing intraindividual processes (Molenaar, 2004). Moving our focus to time-lagged relationships allows us to assess the extent to which certain variables are able to predict future states of the psychological system they are embedded in. Indeed, a great deal of research has emerged touting the value of studying short-term, within-person fluctuations in psychological data (Hamaker & Dolan, 2009; Hamaker et al., 2018).

In the present chapter, I focus exclusively on idiographic applications of temporal networks, where samples are taken and studied within a single individual. However, it is worth noting that these methods can be extended to samples of individuals who are measured over time and analyzed via *multi-level temporal networks* (Epskamp et al., 2018c; Bringmann et al., 2013). Two methods for incorporating moderates in these types of analysis are available in the `modnets` package, and will be touched on briefly at the end of this chapter. I will now move to introduce some basic statistical methods used for specifying time series models with psychological data, and will present an example illustrating how these models can be interpreted.

5.2.1 Autoregressive Models

The most basic type of time series model used in psychological research is the *autoregressive model (AR)*, which, in its most basic form, describes the degree to which past values of some variable predict its subsequent values at some fixed time lag h. For instance, this may be captured in a model that is specified to describe how an individual's current emotional state (e.g., level of sadness), predicts their emotional state at the next observation. The goal of these models is to capture the degree to which current states of some variable is dependent on its past states, or the

past states of some other variables. This model can be estimated via standard OLS regression with

$$X_t = \beta_0 + \beta_1 X_{t-1} + \varepsilon_t, \tag{5.1}$$

where X_t represents the value of some variable at a particular time point, and X_{t-1} is its value at the preceding time point. The more general form of this model (i.e., that which takes into account other lag structures) can be defined as:

$$X_t = \beta_0 + \beta_1 X_{t-1} + \cdots + \beta_h X_{t-h} + \varepsilon_t \tag{5.2}$$

where, given some number of lags h, we can add as many relevant terms to the model as desired (or to the extent that it is afforded by the data).

5.2.2 Vector Autoregressive Models (VAR)

While the most basic AR model only requires that a single variable is measured repeatedly over time, other variables may be included as predictors as well. Similarly, multivariate extensions of the AR model exist wherein more than one response variable, in addition to multiple predictors, are considered. These multivariate extensions are referred to as *vector autoregressive models (VAR)*, and are the most common types of model used to construct temporal networks. In general, a set of measured variables $\mathbf{X}_t \in \mathbb{R}^p$ at time point $t \in \mathbb{Z}$ are modeled as a linear combination of the same variables at earlier time points. For a lag-1 model, VAR(1), we model $\mathbf{X}_t$ as

$$\mathbf{X}_t = \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{X}_{t-1} + \boldsymbol{\varepsilon}_t = \begin{bmatrix} X_{t,1} \\ \vdots \\ X_{t,p} \end{bmatrix} = \begin{bmatrix} \beta_{0,1} \\ \vdots \\ \beta_{0,p} \end{bmatrix} + \begin{bmatrix} \beta_{1,1} & \cdots & \beta_{1,p} \\ \vdots & \ddots & \vdots \\ \beta_{p,1} & \cdots & \beta_{p,p} \end{bmatrix} \begin{bmatrix} X_{t-1,1} \\ \vdots \\ X_{t-1,p} \end{bmatrix} + \begin{bmatrix} \varepsilon_{t,1} \\ \vdots \\ \varepsilon_{t,p} \end{bmatrix}, \tag{5.3}$$

where $\mathbf{B}$ is a matrix of time-lagged relationships for p predictors of p outcomes, $\boldsymbol{\beta}_0$ represents the vector of intercepts for each variable, and $\boldsymbol{\varepsilon}$ reflects their associated error terms.

One attractive feature of these models is that they don't require any prior assumptions about the structure of relationships between variables. Moreover, we can see how the models are easily extended to incorporate interaction terms—given a particular moderator M, we simply add one vector of predictors to Equation 5.3 to reflect the time-lagged main effects for M_{t-1} on some $X_{t,p}$, as well as the interaction effects of each predictor $X_{t-1,p} \times M_{t-1}$ on some $X_{t,p}$. However, there are some important assumptions that must hold in order to preserve the validity of inferences drawn from these types of models.

5.2.2.1 VAR model assumptions

The primary assumption behind linear models of time series data is that the processes are *stationary* over time. There are different types of stationarity to be concerned with, but for Gaussian processes there are only two: (1) mean stationarity, or trend stationarity, and (2) covariance stationarity. The idea behind mean stationarity is that, within each varaible, the mean of the series is expected to be the same across the entire observation window, $\mathbb{E}[\mathbf{x}_t] = \boldsymbol{\mu}_t = \boldsymbol{\mu}$. Linear and nonlinear trends in the data represent violations of this assumption, and are therefore typically investigated at the first stage of an analysis. When univariate trends are discovered, a common technique for transforming the data is by *detrending*, or removing different types of trends from the data by regressing a variable on time and taking the residuals (see Hamaker & Dolan, 2009; Falkenström et al., 2020).

A visual example of this can be seen in Figure 5.1, where data on the number of monthly airline passengers recorded from 1949–1960 (in thousands) are displayed (Box et al., 2015). On the left we see the original time series, for which a positive trend can be observed across the time window (i.e., the average number of airline passengers increased over time). After detrending this series, we see the trend-stationary version on the right. We can see that while this process removed the increasing trend from the data, it preserved the short-term fluctuations, which in this case may be reflecting patterns such as increased air travel during the holidays and decreased travel during other periods. In essence, temporal networks are focused on isolating and testing this aspect (i.e., short-term fluctuations) of time series data.

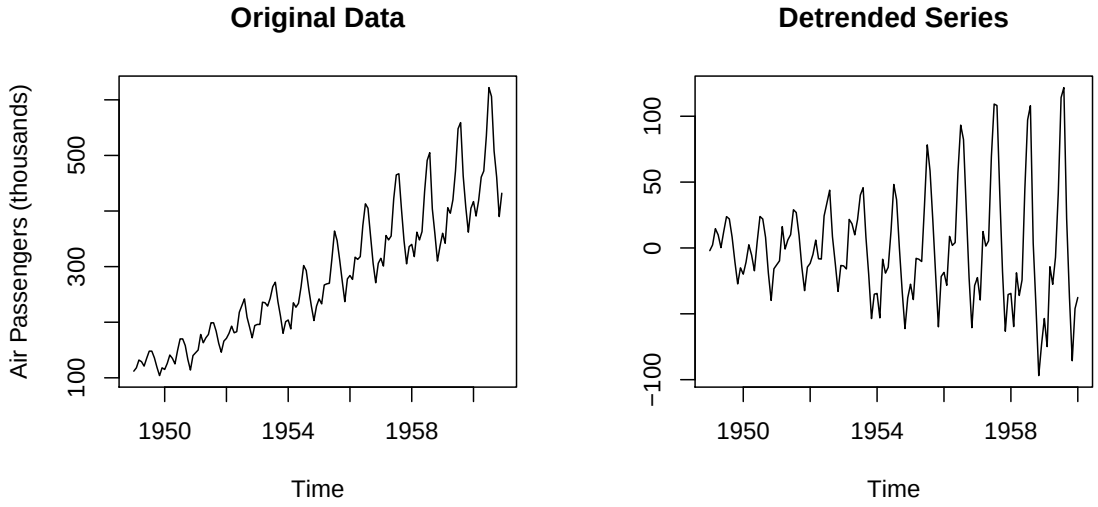

Figure 5.1: Original `AirPassengers` data (left; Box et al., 2015), along with its detrended counterpart (right). Both linear and quadratic trends were removed from the series, where the detrended series was obtained by taking the residuals of the model: $AirPassengers \sim time + time^2$.

Moving over to covariance stationarity, however, we are instead concerned with the autocorrelation structure of the data (as well as the cross-correlation structure, in multivariate settings)—that is, we want to ensure that the relationships between variables (or between a variable and its own past states) at a particular fixed lag (e.g., $h = 1$) remain stable across the observation window. This implies that

$$\boldsymbol{\Sigma}(\mathbf{x}_t, \mathbf{x}_{t-h}) = \boldsymbol{\Sigma}_h = \mathbb{E}[(\mathbf{x}_t - \boldsymbol{\mu})(\mathbf{x}_{t-h} - \boldsymbol{\mu})^{\mathsf{T}}], \tag{5.4}$$

meaning that the auto- and cross-covariances do not depend on occasion t but only on lag h, a fixed interval between any given pair of measurements which lie h observations apart.

To put this in a clearer context, imagine that we're studying how *Depression* and *Anxiety* relate within a single individual over time, and that these variables are each measured once per day across some number of days. In applying the VAR(1) model, we assume that the lag-1 cross-covariances (i.e., for $\{Anx_t,\ Dep_{t-1}\}$, and $\{Anx_{t-1},\ Dep_t\}$), and lag-1 auto-covariances (i.e., $\{Anx_t,\ Anx_{t-1}\}$ and $\{Dep_t,\ Dep_{t-1}\}$) remain the same across the entire sample. In other words, we expect that the relationship between feelings of depression today and feelings of anxiety tomorrow will be similar to (or the same as) that for two consecutive days measured a week from now, or a month from

now. This assumption applies to all other time-lagged relationships as well, and also extends to interaction effects when a moderator is included.

To my knowledge there are not transformation methods available for inducing covariance stationarity, as there are for inducing trend-stationarity, and so some researchers recommend switching to non-parametric methods when substantial violations of this are observed (e.g., Bringmann et al., 2017; Haslbeck et al., 2017). An example of what such a violation might look like (at least in the univariate case) can be seen for the detrended plot in Figure 5.1. Here we can see that the variance of the series increases over time, meaning that for the lag-1 relation $\{X_t,\ X_{t-1}\}$ we will be likely to see smaller values at earlier points in the series, and larger values at later times.

In sum, it is important to consider these assumptions when fitting VAR models to time series data, particularly because they can have substantial impact on both the values and interpretation of parameter estimates. While I will not consider these particular assumptions further in this paper (although see Hamaker & Dolan (2009) for a detailed discussion), there is one final assumption that will be almost ubiquitously violated in practice and is therefore dealt with by methods described in this chapter.

5.2.2.2 Correlated errors and contemporaneous relationships

An important, implicit assumption in VAR models is that the error terms are independent (i.e., uncorrelated) across models. However, all variables are clearly likely to be correlated with themselves across time, and by restricting the model to only consider certain fixed lags (or simply lag-1) we are also imposing the assumption that there are no relationships between variables at other lag sizes. Moreover, our time-lagged model does not take into consideration the relationships between variables *within* time.

This last point reflects the core issue: with standard estimation techniques for VAR models (i.e., OLS), we make the assumption that there are no *contemporaneous relationships* between variables; there are only relationships across some particular lag(s). Because this assumption is unlikely to hold in empirical data, common techniques used to model temporal networks also

model a *contemporaneous network* which accounts for within-timepoint relationships and is also used to inform better estimates of the between-timepoint associations (Epskamp et al., 2018c; Abegaz & Wit, 2013; Rothman et al., 2010).

Before going into more detail about how to estimate these types of models, however, it is worth taking a moment to step back and illustrate what these sorts of models look like, how we can interpret them, and what is added to our understanding of a temporal network by considering contemporaneous relations. For now, it is enough to know that the contemporaneous network models conditional associations (i.e., partial correlations) between the error terms associated with each node in the temporal network.

5.3 Motivating Example

To connect these statistical considerations with some real-world meaning, lets consider a simple illustration of what we might gain by applying moderated network models to ESM data. Imagine we are studying how *Depression*, *Anxiety*, and *Insomnia* relate over time, as well as how those inter-temporal relationships are moderated by *Stress*. We can use a multi-level variant of the temporal MNM if we sample multiple individuals on these variables over time, but I will restrict the present case to the study of a single individual. This echoes the call for 'personalized network models' which could have practical value in clinical settings (c.f. Epskamp et al., 2018b).

In this example, say we conduct a study where these variables are measured for a single individual via smartphone multiple times per day over the course of many days. We then fit a temporal MNM to obtain representations of both the inter-temporal and contemporaneous relationships among the three primary outcomes (*Stress* is assumed to be the moderator)[1]. We can see both networks displayed in Figure 5.2.

These networks afford a detailed interpretation of the relationships among variables. Starting with the temporal network, we see self-directed arrows on each node. These represent *autoregressive effects*, also called *self-loops*, which show how each variable is related to itself at subsequent

[1]Faux network structures were created purely for illustration.

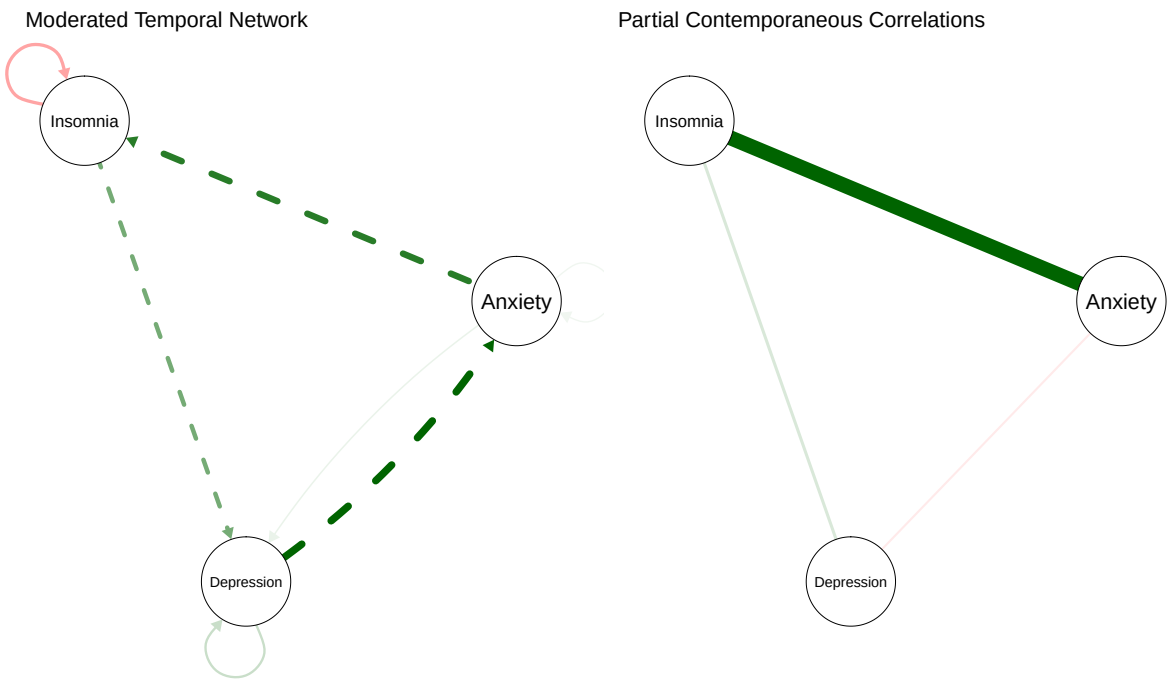

Figure 5.2: Faux temporal and contemporaneous networks illustrating a temporal MNM.

.

time points. The green self-loop for *Depression*, for instance, shows that when depression increases (decreases), it tends to continue increasing (decreasing) at the next time point. The negative self-loop on *Insomnia*, however, indicates that increases in insomnia are associated with subsequent decreases in insomnia (and vice versa). In most settings it is uncommon to find negative self-loops, although in this case it may indicate that increased insomnia often requires an individual to 'catch-up' on lost sleep.

Moving to the cross-lagged relationships (i.e., the arrows between nodes), we see that *Depression* predicts subsequent increases in *Anxiety*, which predicts subsequent increases in *Insomnia*, which itself predicts increases in *Depression*. Moreover, each of these edges are dashed, indicating that the effects are moderated by *Stress*. In this example, we may imagine that higher (lower) levels of *Stress* increase (decrease) the strength of these pairwise relationships.

Next, we see a different sort of pattern in the contemporaneous network. The contemporaneous network reflects the *partial correlations of the residuals* from the time-lagged model. This is obtained by standardizing the inverse of the estimated residual covariance matrix, which itself is

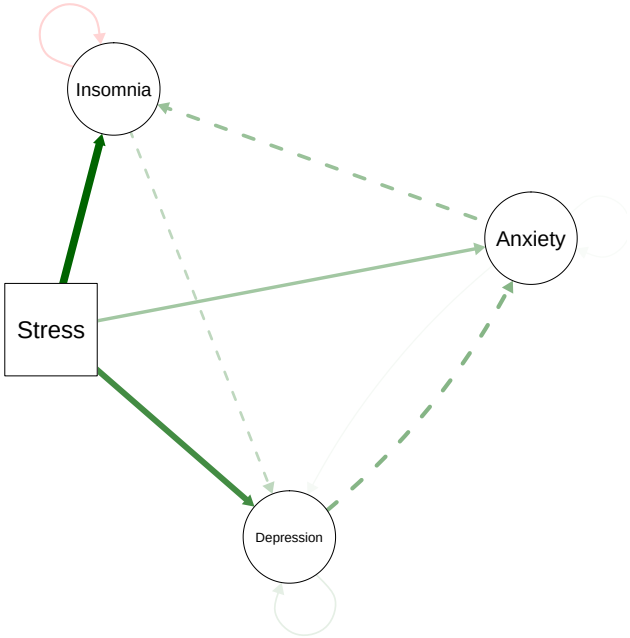

Figure 5.3: Visualizing the moderator in a temporal network.

used in the estimation of **B** (as discussed in the next section). Thus, these reflect the relationships observed, on average, at any given time point after controlling for the inter-temporal relationships.

The strongest contemporaneous relationship is between *Insomnia* and *Anxiety*, which tells us that these two often vary together within timepoints. The edge between *Depression* and *Anxiety* is much weaker, and also negative, indicating that the two are expected to be negatively related at any particular timepoint, even though they have positive time-lagged relationships *across* timepoints. Simply put, these networks show when the participant is experiencing anxiety, he is also likely experiencing insomnia, although not as likely to be experiencing depression. Instead, it appears that depression is the key driver of *subsequent* changes in anxiety.[2]

Lastly, in Figure 5.3 we see the same temporal network as in Figure 5.2, except here we also visualize the main effects of *Stress* on the variables constituting the network. All of these edges are green, indicating positive regression coefficients which signify that higher levels of *Stress*, in this example, predict subsequently higher levels of the other variables.

This fictional example is simply meant to illustrate how temporal MNMs may be used in a

[2]Causal relations, however, cannot be inferred without experimental design.

practical setting to make sense of complex, temporal relationships. And although these networks did not come from real data, we can see the value of investigating *both* contemporaneous and the time-lagged effects which, taken together, give us a more complete picture of the dynamics at hand. In the next section, I provide a description of how exactly we go about estimating these models.

5.4 Estimating Temporal and Contemporaneous Networks

Now that we have a visual sense of what temporal networks look like, it will be useful to have a formal sense of these models as well. We start with the same general framework as with the cross-sectional GGM, where the objective is to model a system of linear equations, although in this case we are using the VAR model with parameters that represent time-lagged, directed associations. Using the previous example to illustrate this, the equations corresponding to Figure 5.2 would be

$$\begin{aligned} Anx_t &= \beta_{0,1} + \beta_{1,1}Anx_{t-1} + \beta_{2,1}Dep_{t-1} + \beta_{3,1}Ins_{t-1} + \beta_{4,1}Stress_{t-1} + \cdots + \varepsilon_{t,1} \\ Dep_t &= \beta_{0,2} + \beta_{1,2}Anx_{t-1} + \beta_{2,2}Dep_{t-1} + \beta_{3,2}Ins_{t-1} + \beta_{4,2}Stress_{t-1} + \cdots + \varepsilon_{t,2} \\ Ins_t &= \beta_{0,3} + \beta_{1,3}Anx_{t-1} + \beta_{2,3}Dep_{t-1} + \beta_{3,3}Ins_{t-1} + \beta_{4,3}Stress_{t-1} + \cdots + \varepsilon_{t,3}. \end{aligned} \tag{5.5}$$

The ellipses indicate where we would write the interaction terms between the each predictor and $Stress_{t-1}$.[3] The reason I include this equation at all, however, is to show that for the saturated VAR, each equation in our model will have all the same predictors. As a result, each β coefficient (with the exception of the intercepts) will represent a separate parameter in the final network model. While with cross-sectional networks we must aggregate the regression coefficients post-hoc to obtain values for edges in the network, in this we can directly use the estimated values to create the network.

After estimating the temporal network, we can then construct the contemporaneous network by modeling the relationships among $\{\varepsilon_{t,1}, \varepsilon_{t,2}, \varepsilon_{t,3}\}$ as a cross-sectional GGM; i.e., using the same methods discussed in Chapter 2. That is, let $\hat{\mathbf{\Sigma}}_t$ be the estimated covariance structure of the error terms; we obtain the contemporaneous network ($\hat{\mathbf{\Omega}}_t$) by standardizing its inverse.

[3] I only omit writing the interaction terms because the full equations are very long and therefore hard to read.

5.4.1 Coping with Correlated Errors

Thus far, the only unpenalized estimator I've discussed in this paper has been OLS. However, this requires taking the assumption that the error terms are independent across equations, which we can be relatively sure is false in most cases. So, how can we incorporate this consideration into the estimation process? One method designed to handle these types of situations is *generalized least squares (GLS)* (Aitkin, 1935). For notational clarity, let $\mathbf{Y}$ be the outcome variables measured at t, and let $\mathbf{X}$ be the matrix of predictors (which are the same variables as in $\mathbf{Y}$ except time-lagged at $t-1$). With GLS, we obtain estimates of the autoregressive and cross-lagged parameters $\mathbf{B}$ with the following equation:

$$\hat{\mathbf{B}} = (\mathbf{X}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\mathbf{X})^{-1}\mathbf{X}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\mathbf{Y}, \tag{5.6}$$

where $\boldsymbol{\Sigma}^{-1}$ is the true inverse covariance matrix for the residuals across equations. This means that the relationships among error terms are being accounted for in the estimation of both the temporal network and the interaction terms.

One issue with this method, however, is that it assumes prior knowledge of the residual covariance matrix. Given that this is unlikely to be the case in most settings, we can use a variant referred to as *feasible generalized least squares (FGLS)*, where an iterative approach is taken to estimate both matrices. Specifically, this procedure involves first estimating $\mathbf{B}$ using OLS, taking the estimate of the covariance matrix $\boldsymbol{\Sigma}$, and then re-estimating $\mathbf{B}$ using Equation 5.6 although substituting the true inverse covariance matrix $\boldsymbol{\Sigma}^{-1}$ with the OLS estimate $\hat{\boldsymbol{\Sigma}}^{-1}$. A new estimate of the residual covariance matrix can then be obtained based on the FGLS estimate of $\mathbf{B}$, and this process is then repeated until $\hat{\mathbf{B}}$ and $\hat{\boldsymbol{\Sigma}}$ achieve a pre-specified level of convergence (Zellner & Theil, 1992).

The application of FGLS to a multivariate regression model is known as a *seemingly unrelated regression (SUR)* model (Zellner, 1962; Zellner & Huang, 1962), and has also been referred to as "multivariate regression with covariance estimation" (MRCE) (Abegaz & Wit, 2013; Rothman et al., 2010). The latter has been used in the psychological network literature (e.g., Epskamp et al., 2018c), and incorporates the GLASSO using two penalty parameters (one for each model).

However, given that this method is not capable of enforcing strong hierarchy (or even supporting interaction terms at all) it is not available for fitting temporal MNMs. The SUR method, however, is unpenalized and affords the inclusion of interaction terms via the `systemfit` package for R (Henningsen & Hamann, 2007). Moreover, estimation of both $\mathbf{B}$ and $\mathbf{\Sigma}^{-1}$ is performed in a simultaneous fashion rather than with the nodewise regression approach used with cross-sectional networks.

In general, this approach to modeling temporal networks with an accompanying contemporaneous network is termed *graphical vector autoregression (GVAR)*. And although the hierarchical LASSO has not yet been integrated with the SUR model, we still have the same variable selection techniques from Chapter 3 available to us via two-stage or resampling-based methods. Specifically, I have setup the `modnets` package to allow temporal MNMs to be fit using variable selection methods by performing sequential, nodewise regressions to select the predictors that will be included in each equation, and then that pattern of predictors is used to apply constraints in the multivariate SUR estimation approach used in `systemfit`. Next, I wanted to get a general sense of how the moderated GVAR performs in analyses of simulated data where true interactions are included in the network structure.

5.5 Simulated Example

In order highlight the potential utility of the moderated GVAR model, I conducted a brief simulation to illustrate what happens when we *fail* to include true moderator effects in a temporal model, as well as to see how the moderated GVAR performs in relatively simple settings. That is, imagine that in the ground truth there is a significant moderator in a temporal network; to what extent does *failing* to include this in the final model affect the accuracy of the parameter estimates? I investigated this under a variety of conditions similar to those tested in with cross-sectional networks in Chapter 4. However I tested a greater number of sample sizes[4] so as to get a more nuanced picture of how the tested models perform as sample size is progressively increased.

[4] Albeit with a smaller range, in order to mimic more realistic data scenarios.

For each of the conditions described below, the objective was to compare the accuracy of parameters from three models: (1) An unmoderated network where the true moderator is excluded from the analysis entirely, (2) an unmoderated network that includes that true moderator as a covariate in each model, and (3) a saturated MNM that includes all interaction terms. All three models were fit to each simulated dataset, and then correlations were computed between the estimated and true temporal networks, as well as between the estimated and true contemporaneous networks to see how each model performed across conditions. Correlations representing the accuracy of interaction estimates were also included for the third model, although these parameters were not included in the other two models.

In general, there were two network sizes $p \in \{5, 10\}$, three levels of interaction sparsity $m \in \{.05, .10, .25\}$ (which meant that there were either 5%, 10% or 25% of possible interactions included in the model, respectively), and thirteen sample sizes[5] that were chosen on a logarithmic scale ranging from $N = 33$ to $N = 500$. This method was used in order to over-sample from the smaller sample sizes, where I expected to see the greatest differences in performance across models. This led to 81 conditions in total, and 100 models with accompanying datasets were generated for each case.

The data generation procedure was based on an algorithm described in the supplementary materials of a paper by Epskamp et al. (2018c), where the only aspect which was changed was that models were adapted to include (and generate) interaction terms for a moderator variable in each case. Interaction effects were only included on temporal networks, and so the procedure for generating contemporaneous networks was identical to that described by Epskamp et al. (2018c).

5.5.1 Simulation Results

The results from the simulation are displayed in Figure 5.4. We see generally good performance across the board, and that correlations between the estimated and true temporal networks (lines

[5] A fourteenth sample size of $N = 25$ was simulated only for $p = 5$, as it was too small to be used for models with $p = 10$.

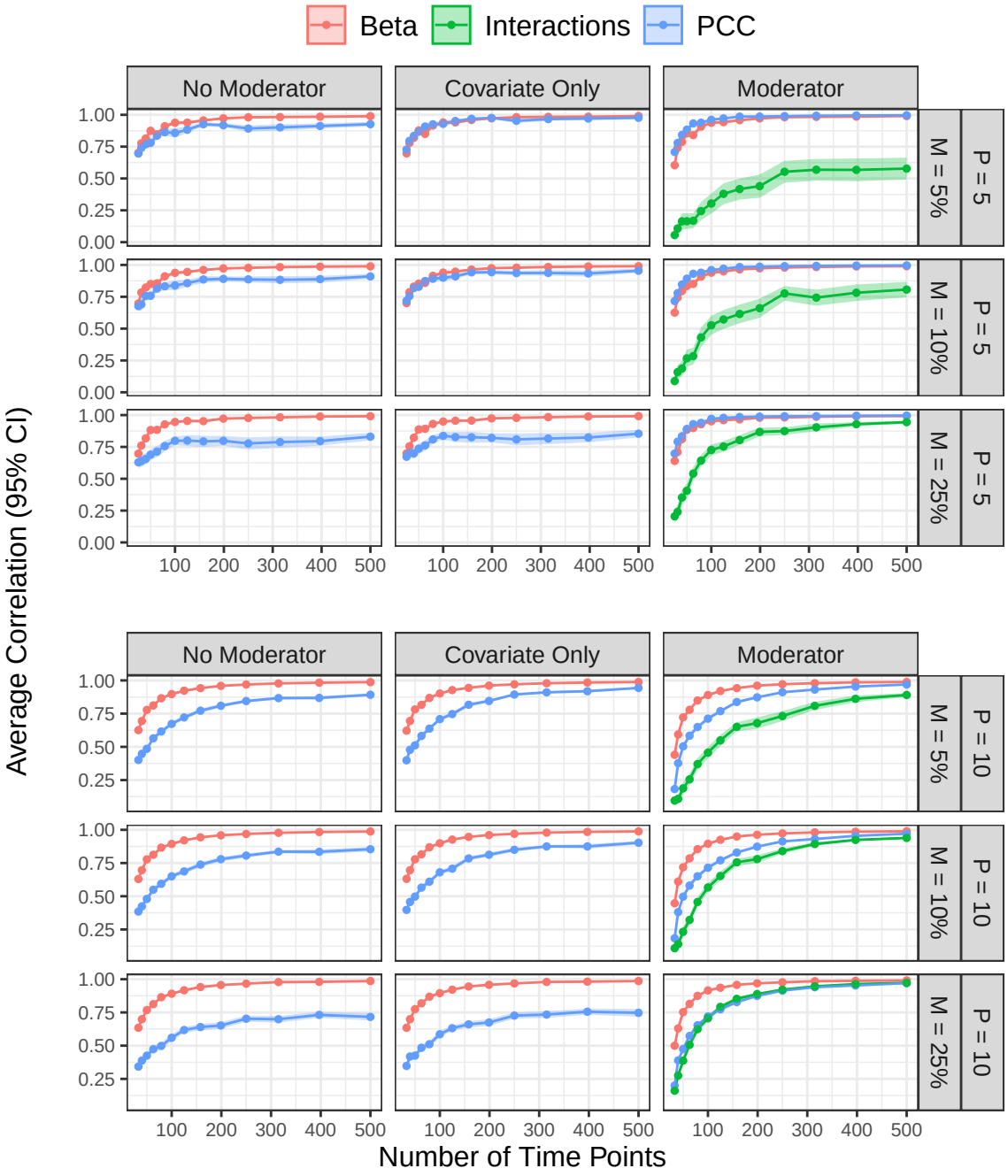

Figure 5.4: Average correlations for three models fit to 100 datasets generated for each of 13 sample sizes, three levels of interaction sparsity, and two network sizes. An additional sample size ($N = 25$) was included for $p = 5$, but was too small for when $p = 10$. 'No Moderator' reflects the unmoderated temporal network, 'Covariate Only' adds the moderator variable as a covariate, and 'Moderator' adds interaction terms. Values for the temporal networks are in red; blue for the partial contemporaneous correlation networks, and green for the interactions.

in red) quickly converged to the upper bound for all three models as sample size increased. This was a surprising result to me, as given that interaction terms were only specified for the temporal networks, I anticipated that excluding these parameters from the model would negatively impact the accuracy of the pairwise estimates. We see instead that the pairwise relationships are accurately estimated even when moderators are omitted from these models.

For the contemporaneous networks, however, we see a very different pattern of results. Specifically, correlations between the estimated and true contemporaneous networks were clearly affected when interactions were omitted, as we see that correlations worsen for the unmoderated models as both network size $(p = 10)$ and the proportion of true interactions (e.g., $m = .25$) increase. This essentially boils down to a simple conclusion: the more true interactions there are in a system, the worse estimates of contemporaneous relationships become when the moderator is not accounted for.

This result was somewhat unanticipated, and I essentially expected to see the opposite pattern; where failing to include interactions would impact estimates from the temporal network moreso than for the contemporaneous network. Thus, it would be beneficial to conduct further investigations in order to determine how exactly moderators influence the contemporaneous relationships. Still, for now the central message is that omitting moderators when they impact a temporal network can have unexpected, downstream consequences on other aspects of the model. It would therefore be interesting to extend this analysis into seeing how failing to include moderators impacts the stability of both the parameter values and centrality estimates (e.g., using the case-dropping bootstrap).

5.6 Empirical Example

To exemplify the temporal MNM using empirical data, I applied the model to idiographic ESM data originally studied by Wichers et al. (2016) and made publicly available by Kossakowski et al. (2017). The data consist of 1471 measurements taken over 239 days (1-10 $[Mdn = 6]$ per day) on psychological variables related to the experience of major depression in a single individual. The

participant was a 57-year-old man with a history of depression who decreased his intake of anti-depressants over the course of the study and experienced a relapse of major depression (Wichers et al., 2016).

The first published analysis of these data investigated whether changes in the variables' auto-correlation structure could predict sudden, non-linear changes in depression (Wichers et al., 2016), and a recent analysis studied the structure of symptoms during the *post-reduction phase*—when medication levels were no longer changed (Epskamp, 2020). In the present analysis, I include both the baseline measurements (when levels were highest) and post-reduction measurements in order to assess the extent to which high versus low medication levels moderated changes in mood and symptom structure. Specifically, I studied 14 continuous variables designed to measure mood states, self-esteem, and physical condition.[6]

The time frame spanned the first 42 days of the study (baseline; $N = 286$ measurements), as well as the 29-day assessment period following medication reduction ($N = 316$ measurements). I chose to omit the actual reduction period, as dosage levels decreased monotonically and with a unit root, which are strong violations of VAR model assumptions. I therefore instead chose to treat medication levels as a binary variable, wherein we are interested in comparing mood states and symptom structure when the participant is (not) taking anti-depressant medication (venlafaxine).

The following items were investigated: "irritated", "satisfied", "lonely", "anxious", "enthusiastic", "guilty", "strong", "restless", "agitated", "worried", "ashamed", "tired", "headache", and "sleepy".[7] At each timepoint the participant was asked to indicate the extent to which they were currently experiencing these. The items were all measured on 7-point Likert scales anchored from *Not at all* to *Very much*, and constituted the network under investigation. All variables were standardized prior to analysis, and the medication variable ("medication") was coded 1 for the baseline period (when medication levels were highest), and 0 during the post-reduction period (when no medication was taken).

[6]These were identified as having sufficient variability by Epskamp (2020).

[7]The actual wording of each item varied (e.g., "I am tired"; "I feel lonely"; "I worry"), but were all simple statements about the degree to which the participant was currently feeling X.

Considering only the baseline and post-reduction periods, there are a total of $N = 602$ complete observations. Given that the data violate the equal-interval assumption for time series data (i.e., no measurements are taken during sleep, of course), I systematically reduced the number of usable observations to 352 time-lagged pairs; i.e., 352 pairs of measurements separated by a single time point (t and $t-1$). This is common practice when analyzing ESM data (e.g., Haslbeck & Waldorp, 2015; Bringmann et al., 2017; Epskamp, 2020), as it is problematic to assume that the relation between the final measurement on any given day and the first measurement of the following day is the same as that for two consecutive measurements within a single day. Thus, when creating the lag-1 model matrices I removed all rows where measurements on a given day were regressed onto measurements from the preceding day. Moreover, only consecutive measurements were retained. This gives us a grand total of $N = 352$ intra-day, lag-1 relationships for the final analysis.

5.6.1 Model Estimation

The following procedure was used to estimate and evaluate the temporal MNM. First, three different models were fit to the data: (1) An unmoderated network that excluded the moderator entirely, (2) an unmoderated network that used the moderator as an exogenous covariate, and (3) a saturated MNM, wherein for each outcome interactions between medication and all predictors were included in the model. The model was fit using the SUR method with FGLS as the estimator; this allowed me to fit a single, multivariate model rather than use the sequential nodewise method. In the tables to follow these models are referred to as fit0, fit1, and fit2, respectively.

The above formed the initial part of the investigation, where the goal was to determine whether or not including medication as a moderator significantly improves the model fit over either of the former unmoderated models. The omnibus likelihood values and fit statistics, along with likelihood-ratio tests, are presented in Table 5.1, and the plot for fit2 (the MNM) is displayed in Figure 5.5.

Next, I applied the variable selection techniques described in the foregoing chapters to obtain a more parsimonious model and assess more specifically which interactions are important in the

Temporal MNM Fit Statistics										
Omnibus Fit Values					Likelihood-Ratio Tests					
Model	*LL*	*df*	*AIC*	*BIC*	M1	M2	χ^2	df_Δ	*p*-value	Decision
fit0	-5316.51	315	11263.01	13311.36	fit0	fit1	72.18	14	< .001	fit1
fit1	-5280.42	329	11218.84	13358.22	fit0	fit2	359.43	210	< .001	fit2
fit2	-5136.79	525	11323.58	14737.50	fit1	fit2	287.25	196	< .001	fit2

Table 5.1: Omnibus fit statistics and likelihood-ratio tests (LRT) associated with each of three models: fit0 is the model where the 'medication' variable is excluded entirely; fit1 includes 'medication' as a covariate, and then fit2 includes interaction terms.

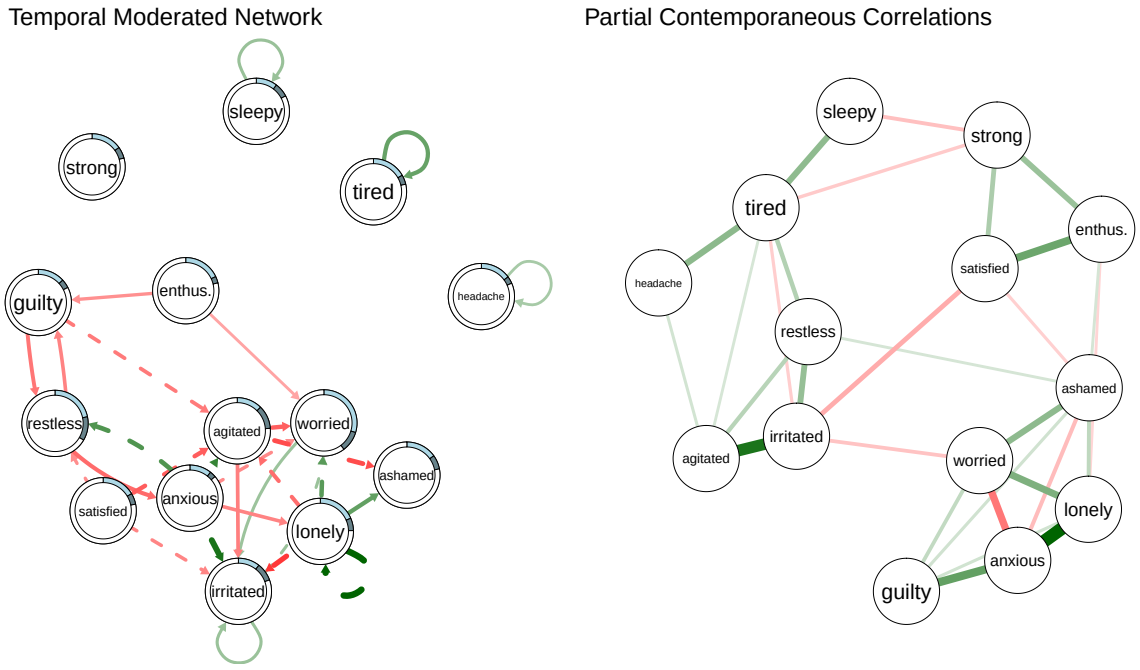

Figure 5.5: Saturated temporal MNM with *p*-value thresholding at $p < .05$ for both networks. Dashed edges reflect significant interaction effects (only specified in the temporal network). Light blue circles around the perimeter of each node reflect the R^2 value for each variable. The darker blue that can be seen for some nodes indicate the increase R^2_Δ that occurred by including interaction terms (fit2) in comparison with the unmoderated network (fit0).

model.

5.6.2 Variable Selection

Because this is intended to be an exploratory, demonstrative analysis, I tested a series of MNMs fit with the variable selection methods discussed in the previous chapter. Given the temporal dependency in the data, however, I used a variation on the bootstrapped selection procedure called the *fixed block bootstrap*. With this method, the bootstrapped version of Algorithm 1 was applied except at each iteration I sampled 'blocks' of fixed length l from the data (rather than single values, one at a time), and append them end-to-end until a new series of length N is created. This method is provided as an option within the `resample` function created for `modnets`. The length chosen for l was based on previous research showing that blocks of size $l = 3.15(N^{\frac{1}{3}})$ can be optimal under many conditions (c.f. Kunsch, 1989; Politis & Romano, 1994).[8] Thus, given $N = 352$ time-lagged observations, this size was set to $l = 22$. The other two resampling methods (multi-sample split and stability selection) were not used as I have yet to implement the block-resampling approach in those contexts.

In order to determine which model should be selected, I first assessed differences in likelihood using LRTs with Bonferroni corrections comparing every possible pair of models. The omnibus fit statistics for each model are shown in Table 5.2, where the right-most column reflects the number of times each model was selected by the LRTs. Given that there were 8 models in total, there were 7 LRTs comparing each model to all others.

In line with the simulation results from Chapter 4, we see that the bootstrap method was selected across all comparisons with the other models, while the two-stage AIC was selected in 6/7 comparisons, and the two-stage CV_{min} was selected in 5/7 comparisons. Moreover, we see that while the bootstrap minimized the AIC, the two-stage AIC and CV_{min} methods returned lower BIC values. This highlights a common situation where the researcher must decide which criterion to use for selecting the best model. In this case, given the exploratory nature of the analysis, I

[8] This is set as the default for the block bootstrap in the `resample` function, although can also be specified.

Model Fit Statistics							
Name	Model	Type	*LL*	*df*	*AIC*	*BIC*	LRT
bootstrap	Block Bootstrap	MNM	-5242.57	278	11041.14	12848.89	7
AIC	Two-Stage	MNM	-5262.51	268	11061.02	12803.74	6
CVmin	Two-Stage	MNM	-5262.66	270	11065.31	12821.04	5
fit2	Saturated	MNM	-5136.79	525	11323.58	14737.50	4
EBIC25	Two-Stage	MNM	-5367.83	168	11071.65	12164.11	3
BIC	Two-Stage	MNM	-5365.40	170	11070.80	12176.26	2
fit1	Saturated	Covariate	-5280.42	329	11218.84	13358.22	1
fit0	Saturated	Unmoderated	-5316.51	315	11263.01	13311.36	0

Table 5.2: All names reflect the same meanings as in Table 4.2, with the exception that $B = 1000$ iterations were used for the block bootstrap. The LRT column reflects the number of times that each model was selected in a series of likelihood-ratio tests comparing it with every other model.

selected the bootstrap model since it appeared to strike an optimal balance between parsimony and likelihood relative to the other models.

5.6.3 Analysis Results

The models obtained by the block-bootstrap selection method are displayed in Figure 5.6. Specifically, I've plotted here the contemporaneous network, as well as the two conditional temporal networks—the network on the left shows the pairwise relationships when medication = 1 (during the baseline phase), and the one on the right reflects when medication = 0. The first aspect of these plots that stands out is that there are substantially more interactions and pairwise effects that emerge after the participant ceased anti-depressant medication. That is, relationships among mood states and depression symptoms seem to become more dense and well-connected once medication was no longer being taken.

In addition to the plots showing the final model (i.e., the block-bootstrap results; Figure 5.6), centrality estimates for the model are displayed in Appendix F, along with results from the case-dropping bootstrap which I conducted to investigate the stability of the selected model. We can see

Correlation Stability $CS(\rho_{.95} \geq .70)$								
	Temporal Network					Contemporaneous		
Parameters	Edges	OutStr	InStr	OutEI	InEI	Edge	Strength	EI
Pairwise	.28	.28	.52	.28	.20	.75	.28	.67
Interactions	.28	.20	.28	.28	.28	-	-	-

Table 5.3: Correlation stability coefficients obtained via the case-dropping bootstrap.

in Table 5.3 that most correlation stability coefficients meet the recommended criterion (Epskamp et al., 2018a) of being larger than .25. Centrality estimates (i.e., strength and expected influence) in the case of temporal networks have to be amended, however, as we are now studying directed relationships rather than undirected connections. Thus, each centrality measure has an "outgoing" and "incoming" version, where *out-strength*, for instance, shows the overall magnitude of the extent to which one node predicts subsequent values of other nodes in the network, while *in-strength* tells us the extent to which a particular node is predicted by other nodes in the network.

Overall, we see that while the *CS* coefficients are not particularly high for these metrics, they still exceed an acceptable level for both pairwise and interaction effects.

5.7 Conclusion

In this chapter, I covered the foundation of temporal MNMs, as well as provided a number of demonstrations to show how these models can be constructed, interpreted, and applied to empirical data. The `modnets` package is designed to afford a wide array of flexible options for researchers to use when fitting these models, and there are still other techniques that must be covered in a future paper. For now, we have at least seen the basics of how these models work, as well as what they can contribute to a variety of different types of investigation. From cross-sectional to longitudinal data, and exploratory to confirmatory analyses, my goal was to provide options that would accommodate a number of different objectives for these types of models.

Moderation analysis has been a core statistical tool within psychology for a number of years, and it has certainly been recognized as a staple within our methodological toolkit. Psychologi-

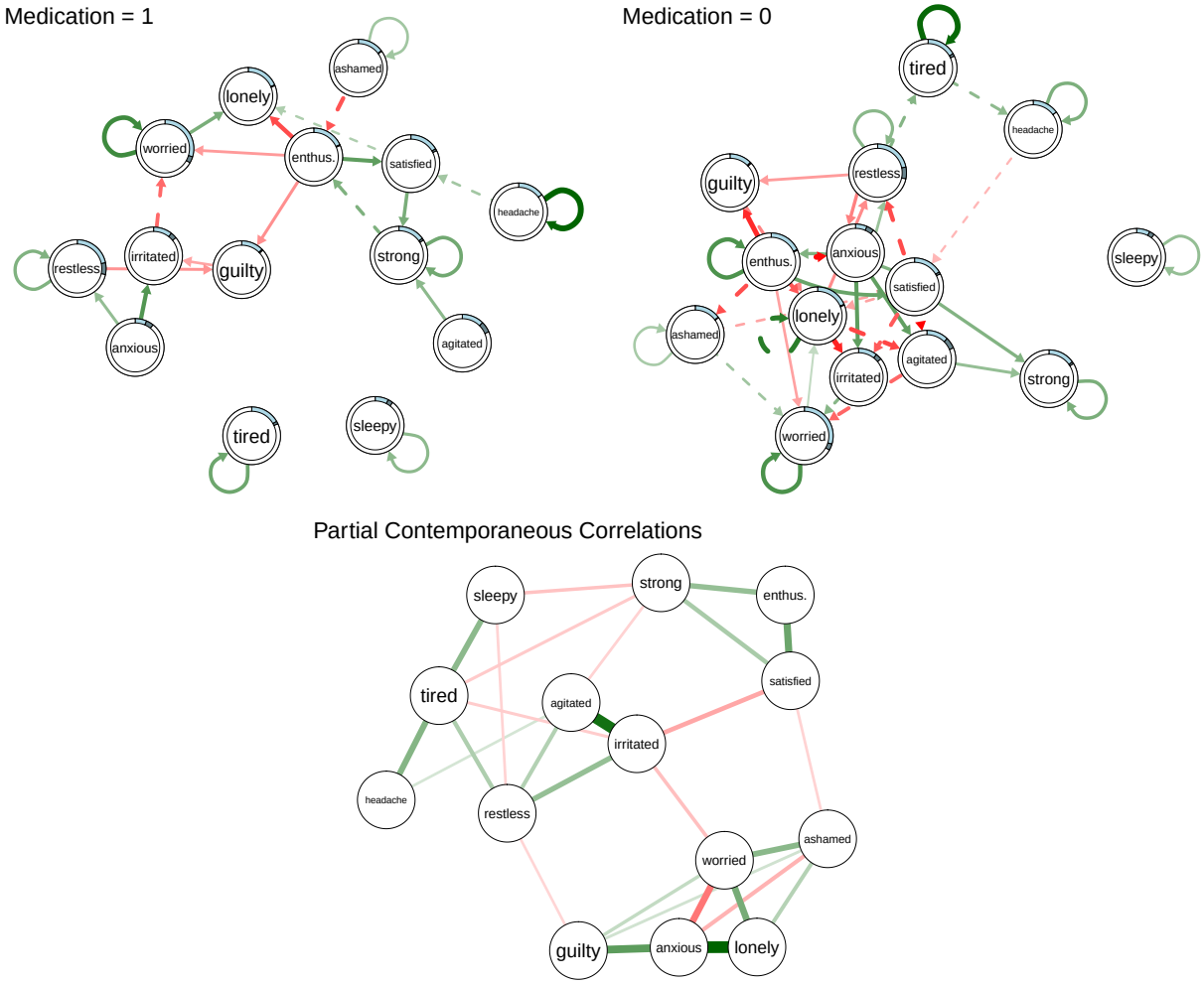

Figure 5.6: Conditional temporal networks for when the participant is taking versus not taking medication, as well as the contemporaneous network. These plots represent the results for the final model which was selected based on the block bootstrap with AIC selection of λ.

cal networks represent a newer class of methods, although have gained incredible popularity and widespread use in the short time since they have been developed. Moderated networks thus offer an integration of these two frameworks in a way that is both incremental (e.g., adding interaction terms to nodewise regression models; treating moderators as exogenous variables) and novel in its approach (e.g., visualizing conditional networks; performing global model comparison and goodness-of-fit tests).

Finally, the `modnets` package, which is the centerpiece of this project, is currently available online for free download and use via GitHub (`https://github.com/tswanson222/modnets`). It is not yet a compiled package, however, and so more work needs to be done in order to make it formally available for R. Still, it can be used quite easily in its current form, and is available for researchers to use for addressing a wide variety of interesting questions regarding the influence of moderators on both cross-sectional and temporal networks.

References

Abegaz, F. & Wit, E. (2013). Sparse time series chain graphical models for reconstructing genetic networks. *Biostatistics*, 14, 586–599.

Aiken, L. S. & West, S. G. (1991). *Multiple regression: Testing and interpreting interactions*. Newbury Park, CA: Sage Publications.

Aitkin, A. (1935). On least squares and linear combination of observations. *Proceedings of the Royal Society of Edinburgh*, 55, 42–48.

Aoshima, M. & Yata, K. (2011). Two-stage procedures for high-dimensional data. *Sequential Analysis*, 30, 356–399.

Barabási, A.-L. et al. (2016). *Network science*. Cambridge University Press.

Baron, R. M. & Kenny, D. A. (1986). The moderator–mediator variable distinction in social psychological research: Conceptual, strategic, and statistical considerations. *Journal of Personality and Social Psychology*, 51, 1173–1182.

Berk, R., Brown, L., & Zhao, L. (2010). Statistical inference after model selection. *Journal of Quantitative Criminology*, 26, 217–236.

Berry, W. D., DeMeritt, J. H., & Esarey, J. (2016). Bias and overconfidence in parametric models of interactive processes. *American Journal of Political Science*, 60, 521–539.

Bien, J., Taylor, J., & Tibshirani, R. (2013). A lasso for hierarchical interactions. *The Annals of Statistics*, 41, 1111–1141.

Borgatti, S. P., Mehra, A., Brass, D. J., & Labianca, G. (2009). Network analysis in the social sciences. *Science*, 323, 892–895.

Boschloo, L., Schoevers, R. A., van Borkulo, C. D., Borsboom, D., & Oldehinkel, A. J. (2016). The network structure of psychopathology in a community sample of preadolescents. *Journal of Abnormal Psychology*, 125, 599.

Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons.

Brambor, T., Clark, W. R., & Golder, M. (2006). Understanding interaction models: Improving empirical analyses. *Political Analysis*, 14, 63–82.

Braumoeller, B. F. (2004). Hypothesis testing and multiplicative interaction terms. *International Organization*, 58, 807–820.

Bringmann, L. F., Elmer, T., Epskamp, S., Krause, R. W., Schoch, D., Wichers, M., Wigman, J. T., & Snippe, E. (2019). What do centrality measures measure in psychological networks? *Journal of Abnormal Psychology*, 128, 892.

Bringmann, L. F., Hamaker, E. L., Vigo, D. E., Aubert, A., Borsboom, D., & Tuerlinckx, F. (2017). Changing dynamics: Time-varying autoregressive models using generalized additive modeling. *Psychological Methods*, 22, 409–425.

Bringmann, L. F., Vissers, N., Wichers, M., Geschwind, N., Kuppens, P., Peeters, F., Borsboom, D., & Tuerlinckx, F. (2013). A network approach to psychopathology: new insights into clinical longitudinal data. *PLOS ONE*, 8, e60188.

Casella, G. & George, E. I. (1992). Explaining the gibbs sampler. *The American Statistician*, 46, 167–174.

Chen, S., Witten, D. M., & Shojaie, A. (2014). Selection and estimation for mixed graphical models. *Biometrika*, 102, 47–64.

Costantini, G. & Epskamp, S. (2017). Estimategroupnetwork: Perform the joint graphical lasso and selects tuning parameters. *R package version 0.1*, 2, 2017.

Costantini, G., Epskamp, S., Borsboom, D., Perugini, M., Mõttus, R., Waldorp, L. J., & Cramer, A. O. (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.

Costantini, G., Richetin, J., Preti, E., Casini, E., Epskamp, S., & Perugini, M. (2019). Stability and variability of personality networks. a tutorial on recent developments in network psychometrics. *Personality and Individual Differences*, 136, 68–78.

Costenbader, E. & Valente, T. W. (2003). The stability of centrality measures when networks are sampled. *Social Networks*, 25, 283–307.

Cramer, A. O. J., van Borkulo, C. D., Giltay, E. J., van der Maas, H. L. J., Kendler, K. S., Scheffer, M., & Borsboom, D. (2016). Major depression as a complex dynamic system. *PLOS ONE*, 11, e0167490.

Cramer, A. O. J., Waldorp, L. J., van der Maas, H. L. J., & Borsboom, D. (2010). Comorbidity: A network perspective. *Behavioral and Brain Sciences*, 33, 137–150.

Dablander, F. & Hinne, M. (2019). Node centrality measures are a poor substitute for causal inference. *Scientific Reports*, 9, 6846.

Danaher, P., Wang, P., & Witten, D. M. (2014). The joint graphical lasso for inverse covariance estimation across multiple classes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76, 373–397.

De Deyne, S. & Storms, G. (2008). Word associations: Network and semantic properties. *Behavior Research Methods*, 40, 213–231.

Dezeure, R., Bühlmann, P., Meier, L., & Meinshausen, N. (2015). High-dimensional inference: Confidence intervals, p-values and r-software hdi. *Statistical Science*, (pp. 533–558).

Drton, M. & Maathuis, M. H. (2017). Structure learning in graphical modeling. *Annual Review of Statistics and Its Application*, 4, 365–393.

Drton, M. & Perlman, M. D. (2004). Model selection for gaussian concentration graphs. *Biometrika*, 91, 591–602.

Drton, M. & Perlman, M. D. (2007). Multiple testing and error control in gaussian graphical model selection. *Statistical Science*, 22, 430–449.

Epskamp, S. (2020). Psychometric network models from time-series and panel data. *Psychometrika*, (pp. 1–26).

Epskamp, S., Borsboom, D., & Fried, E. I. (2018a). Estimating psychological networks and their accuracy: A tutorial paper. *Behavior Research Methods*, 50, 195–212.

Epskamp, S. & Fried, E. I. (2018a). Package 'bootnet'. R package version 1.3.

Epskamp, S. & Fried, E. I. (2018b). A tutorial on regularized partial correlation networks. *Psychological Methods*, 23, 617–634.

Epskamp, S., Rhemtulla, M., & Borsboom, D. (2017). Generalized network psychometrics: Combining network and latent variable models. *Psychometrika*, 82, 904–927.

Epskamp, S., van Borkulo, C. D., van der Veen, D. C., Servaas, M. N., Isvoranu, A.-M., Riese, H., & Cramer, A. O. J. (2018b). Personalized network modeling in psychopathology: The importance of contemporaneous and temporal connections. *Clinical Psychological Science*, 6, 416–427.

Epskamp, S., Waldorp, L. J., Mõttus, R., & Borsboom, D. (2018c). The gaussian graphical model in cross-sectional and time-series data. *Multivariate Behavioral Research*, 53, 453–480.

Falkenström, F., Solomonov, N., & Rubel, J. (2020). Using time-lagged panel data analysis to study mechanisms of change in psychotherapy research: Methodological recommendations. *Counselling and Psychotherapy Research*.

Forbush, K., Siew, C. S. Q., & Vitevitch, M. (2016). Application of network analysis to identify interactive systems of eating disorder psychopathology. *Psychological Medicine*, 46, 2667–2677.

Fried, E. I. & Cramer, A. O. J. (2017). Moving forward: Challenges and directions for psychopathological network theory and methodology. *Perspectives on Psychological Science*, 12, 999–1020.

Fried, E. I., van Borkulo, C. D., Cramer, A. O. J., Boschloo, L., Schoevers, R. A., & Borsboom, D. (2017). Mental disorders as networks of problems: A review of recent insights. *Social Psychiatry and Psychiatric Epidemiology*, 52, 1–10.

Friedman, J., Hastie, T., & Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9, 432–441.

Friedman, J., Hastie, T., & Tibshirani, R. (2010b). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33, 1–22.

Friedrich, R. J. (1982). In defense of multiplicative terms in multiple regression equations. *American Journal of Political Science*, (pp. 797–833).

Galaskiewicz, J. & Wasserman, S. (1993). Social network analysis: Concepts, methodology, and directions for the 1990s. *Sociological Methods & Research*, 22, 3–22.

Gelman, A. & Su, Y.-S. (2018). *arm: Data analysis using regression and multilevel/hierarchical models*. R package version 1.10-1.

Hainmueller, J., Mummolo, J., & Xu, Y. (2019). How much should we trust estimates from multiplicative interaction models? simple tools to improve empirical practice. *Political Analysis*, 27, 163–192.

Hall, J. & Sammons, P. (2013). Mediation, moderation & interaction. In T. Teo (Ed.), *Handbook of quantitative methods for educational research* (pp. 267–286). Rotterdam: Sense Publishers.

Hamaker, E. L., Asparouhov, T., Brose, A., Schmiedek, F., & Muthén, B. (2018). At the frontiers of modeling intensive longitudinal data: Dynamic structural equation models for the affective measurements from the cogito study. *Multivariate Behavioral Research*, 53, 820–841.

Hamaker, E. L. & Dolan, C. V. (2009). Idiographic data analysis: Quantitative methods—from simple to advanced. In J. Valsiner, P. C. M. Molenaar, M. Lyra, & N. Chaudhary (Eds.), *Dynamic process methodology in the social and developmental sciences* (pp. 191–216). New York, NY: Springer-Verlag.

Haslbeck, J. M. B., Borsboom, D., & Waldorp, L. J. (2018). *Moderated network models*. Retrieved from https://arxiv.org/abs/1807.02877.

Haslbeck, J. M. B., Bringmann, L. F., & Waldorp, L. J. (2017). *How to estimate time-varying vector autoregressive models? A comparison of two methods*. Retrieved from https://arxiv.org/abs/1711.05204.

Haslbeck, J. M. B. & Fried, E. I. (2017). How predictable are symptoms in psychopathological networks? a reanalysis of 18 published datasets. *Psychological Medicine*, 47, 2767–2776.

Haslbeck, J. M. B. & Waldorp, L. J. (2015). *mgm: Structure estimation for time-varying mixed graphical models in high-dimensional data*. Retrieved from https://arxiv.org/abs/1510.06871v6.

Henningsen, A. & Hamann, J. D. (2007). systemfit: A package for estimating systems of simultaneous equations in r. *Journal of statistical software*, 23, 1–40.

Hills, T. T., Maouene, J., Riordan, B., & Smith, L. B. (2010). The associative structure of language: Contextual diversity in early word learning. *Journal of Memory and Language*, 63, 259–273.

Hinshaw, S. P. (2002). Intervention research, theoretical mechanisms, and causal processes related to externalizing behavior patterns. *Development and Psychopathology*, 14, 789–818.

Kindermann, R. P. & Snell, J. L. (1980). On the relation between markov random fields and social networks. *Journal of Mathematical Sociology*, 7, 1–13.

Korkmaz, S., Goksuluk, D., & Zararsiz, G. (2014). Mvn: An r package for assessing multivariate normality. *The R Journal*, 6(2), 151–162.

Kossakowski, J., Groot, P., Haslbeck, J., Borsboom, D., & Wichers, M. (2017). Data from 'critical slowing down as a personalized early warning signal for depression'. *Journal of Open Psychology Data*, 5.

Kunsch, H. R. (1989). The jackknife and the bootstrap for general stationary observations. *The Annals of Statistics*, 17, 1217–1241.

Lance, C. E. (1988). Residual centering, exploratory and confirmatory moderator analysis, and decomposition of effects in path models containing interactions. *Applied Psychological Measurement*, 12, 163–175.

Lauritzen, S. L. (1996). *Graphical models*, volume 17. Oxford: Clarendon Press.

Lim, M. & Hastie, T. (2015). Learning interactions via hierarchical group-lasso regularization. *Journal of Computational and Graphical Statistics*, 24, 627–654.

Little, T. D., Bovaird, J. A., & Widaman, K. F. (2006). On the merits of orthogonalizing powered and product terms: Implications for modeling interactions among latent variables. *Structural Equation Modeling*, 13, 497–519.

Marsman, M., Borsboom, D., Kruis, J., Epskamp, S., van Bork, R., Waldorp, L. J., Maas, H. L. J. v. d., & Maris, G. (2018). An introduction to network psychometrics: Relating ising network models to item response theory models. *Multivariate Behavioral Research*, 53, 15–35.

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, 836–849.

Meinshausen, N. & Bühlmann, P. (2006). High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34, 1436–1462.

Meinshausen, N. & Bühlmann, P. (2010). Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72, 417–473.

Meinshausen, N., Meier, L., & Bühlmann, P. (2009). P-values for high-dimensional regression. *Journal of the American Statistical Association*, 104, 1671–1681.

Molenaar, P. C. M. (2004). A manifesto on psychology as idiographic science: Bringing the person back into scientific psychology, this time forever. *Measurement*, 2, 201–218.

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, 1533–1547.

Nielson, D. M., Smith, T. A., Sreekumar, V., Dennis, S., & Sederberg, P. B. (2015). Human hippocampus represents space and time during retrieval of real-world memories. *Proceedings of the National Academy of Sciences*, 112, 11078–11083.

Politis, D. N. & Romano, J. P. (1994). The stationary bootstrap. *Journal of the American Statistical association*, 89, 1303–1313.

Powers, D. M. (2011). Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 2, 37–63.

Read, S. J., Monroe, B. M., Brownstein, A. L., Yang, Y., Chopra, G., & Miller, L. C. (2010). A neural network model of the structure and dynamics of human personality. *Psychological Review*, 117, 61–92.

Revelle, W. (2018). *psych: Procedures for Psychological, Psychometric, and Personality Research*. Northwestern University, Evanston, Illinois. R package version 1.8.12.

Robinaugh, D. J., Millner, A. J., & McNally, R. J. (2016). Identifying highly influential nodes in the complicated grief network. *Journal of Abnormal Psychology*, 125, 747.

Rothman, A. J., Levina, E., & Zhu, J. (2010). Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics*, 19, 947–962.

Ryan, O., Bringmann, L., & Schuurman, N. K. (2019). *The challenge of generating causal hypotheses using network models*. Retrieved from https://psyarxiv.com/ryg69/.

Saunders, D. R. (1955). The "moderator variable" as a useful tool in prediction. In *Proceedings of the 1954 Invitational Conference on Testing Problems* (pp. 54–58). Princeton, NJ: Educational Testing Service.

Siew, C. S. Q., McCartney, M. J., & Vitevitch, M. S. (2019). Using network science to understand statistics anxiety among college students. *Scholarship of Teaching and Learning in Psychology*.

Tibshirani, R. (1996). Regression selection and shrinkage via the lasso. *Journal of the Royal Statistical Society Series B*, 58, 267–288.

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.

van Borkulo, C. D., Boschloo, L., Borsboom, D., Penninx, B. W., Waldorp, L. J., & Schoevers, R. A. (2015). Association of symptom network structure with the course of depression. *JAMA Psychiatry*, 72, 1219–1226.

van Borkulo, C. D., Boschloo, L., Kossakowski, J., Tio, P., Schoevers, R. A., Borsboom, D., & Waldorp, L. J. (2017). Comparing network structures on three aspects: A permutation test. *Manuscript submitted for publication*.

Vitevitch, M. S. & Luce, P. A. (2016). Phonological neighborhood effects in spoken word perception and production. *Annual Review of Linguistics*, 2, 75–94.

Wasserman, L. & Roeder, K. (2009). High dimensional variable selection. *Annals of Statistics*, 37, 2178–2201.

West, D. B. (1996). *Introduction to graph theory*. Upper Saddle River, NJ: Prentice Hall, 2 edition.

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, 114–116.

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, 28.

Williams, D. R., Rhemtulla, M., Wysocki, A. C., & Rast, P. (2019). On nonregularized estimation of psychological networks. *Multivariate Behavioral Research*, (pp. 1–23).

Wright Jr, G. C. (1976). Linear models for evaluating conditional relationships. *American Journal of Political Science*, (pp. 349–373).

Yang, E., Baker, Y., Ravikumar, P., Allen, G., & Liu, Z. (2014). Mixed graphical models via exponential families. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics* (pp. 1042–1050).

Yang, Y. (2005). Can the strengths of aic and bic be shared? a conflict between model identification and regression estimation. *Biometrika*, 92, 937–950.

Zellner, A. (1962). An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. *Journal of the American Statistical Association*, 57, 348–368.

Zellner, A. & Huang, D. S. (1962). Further properties of efficient estimators for seemingly unrelated regression equations. *International Economic Review*, 3, 300–313.

Zellner, A. & Theil, H. (1992). Three-stage least squares: Simultaneous estimation of simultaneous equations. In *Henri Theil's Contributions to Economics and Econometrics* (pp. 147–178). Springer.

Appendix A

Gibbs Sampling Procedure for Simulation Study

At the start of the procedure, a random value is drawn from $\mathcal{N}(0,1)$ to define the mean of the moderator μ_m. At the first iteration, initial values for the p variables are then drawn from $\mathcal{N}(0,1)$, and the moderator is initialized by a single draw from $\mathcal{N}(\mu_m,1)$. Then, the procedure outlined in Algorithm 3 is applied. Overall, this is a standard Gibbs sampling procedure, with the exception being that the order of the variables for which values are generated is randomized at each iteration. The reason for this is because of a bias caused by the presence of interaction effects. By randomizing the order of the variables, it will never be the case that certain interactions are always defined by values from the preceding iteration, while others are always defined by values from the current iteration. Instead, each variable will be defined by a mixture of the two iterations. ω is the number of 'burn-in' iterations, which are removed from the final data (set to $\omega = 250$ for the current study).

Algorithm 3 Gibbs sampling procedure

For each iteration $t = 2,\ldots,(\omega+N)$:

1. Draw a value for the moderator $m^{(t)} \sim \mathcal{N}(\mu_m,1)$.
2. Define $k^{(t)}$ as a random order of the values $\{1,\ldots,p\}$.
3. For $i = k_1^{(t)},\ldots,k_p^{(t)}$:
   (a) Take the sum of the products of the most recent draws for each variable (either t or $t-1$) and the parameter values in row i of the pairwise effects matrix.
   (b) Take the sum of the products of the most recent draws for each variable, the corresponding draws for m, and the parameter values in row i of the interaction effects matrix.
   (c) Define $\mu_i^{(t)}$ as the sum of the two preceding sums (e.g., when there are no interactions between any variable $\times$ m on i, the second sum will be zero. The same applies to the first sum but only with regards to pairwise effects).
   (d) Draw a single value for $i^{(t)} \sim \mathcal{N}(\mu_i^{(t)},1)$.

Appendix B

Simulation Study—Specificity

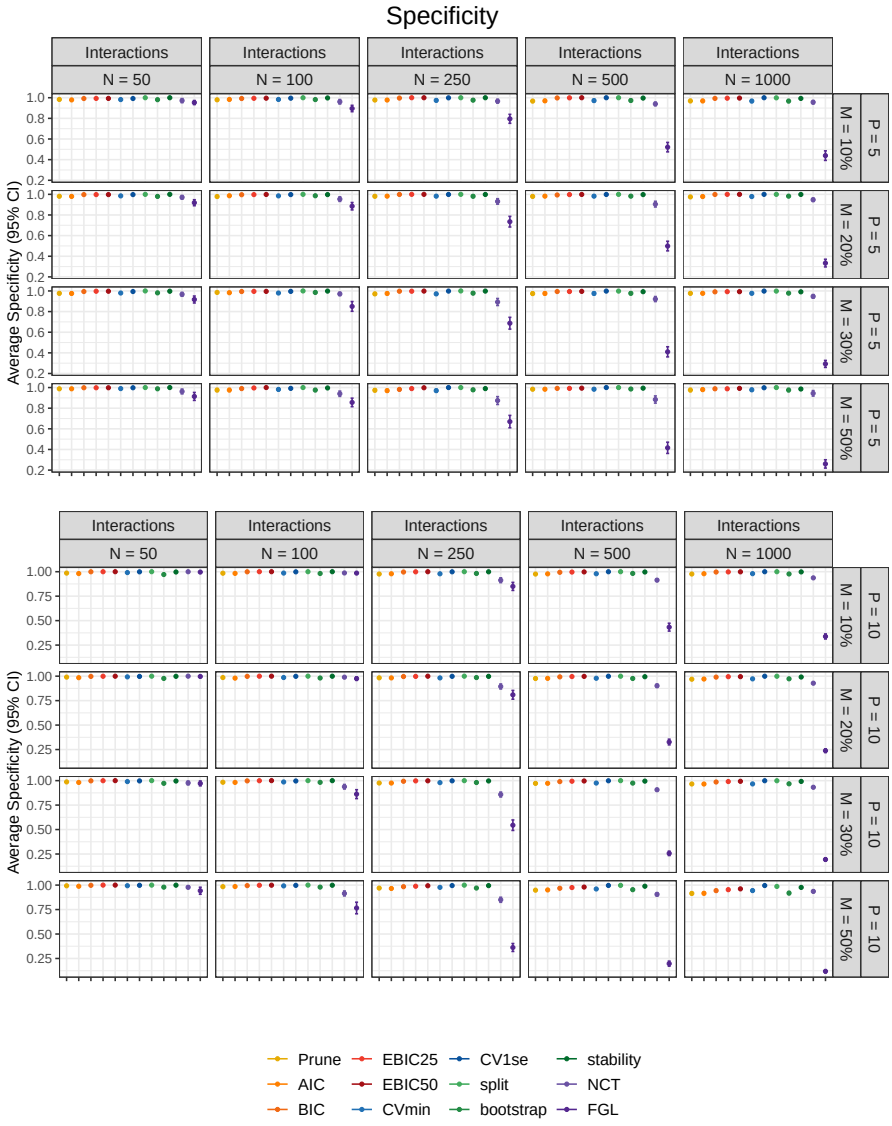

Figure B.1: Average specificity across models and conditions for the *continuous* version of the simulation study. Each plotted point is based on 100 simulated datasets

Appendix C

Simulation Study—Primary Analysis Means

C.1 Grand Means

Sensitivity					**FDR**				
Model	Type	*M*	*SE*	*Mdn*	Model	Type	*M*	*SE*	*Mdn*
FGL	Sample-split	.997	.001	1.000	bootstrap	MNM	.212	.005	.000
bootstrap	MNM	.624	.006	.778	AIC	MNM	.219	.006	.000
AIC	MNM	.618	.006	.778	CVmin	MNM	.220	.006	.000
CVmin	MNM	.609	.006	.739	Prune	MNM	.251	.006	.000
Prune	MNM	.600	.006	.739	BIC	MNM	.265	.007	.000
BIC	MNM	.538	.007	.600	CV1se	MNM	.289	.007	.000
NCT	Sample-split	.531	.007	.652	stability	MNM	.311	.007	.000
stability	MNM	.518	.007	.565	EBIC25	MNM	.327	.007	.000
EBIC25	MNM	.494	.007	.500	FGL	Sample-split	.362	.005	.400
CV1se	MNM	.464	.006	.400	EBIC50	MNM	.376	.008	.000
EBIC50	MNM	.451	.007	.333	split	MNM	.437	.008	.000
split	MNM	.400	.007	.200	NCT	Sample-split	.481	.007	.357
MCC					**Specificity**				
Model	Type	*M*	*SE*	*Mdn*	Model	Type	*M*	*SE*	*Mdn*
bootstrap	MNM	.631	.006	.756	split	MNM	.999	.000	1.000
AIC	MNM	.623	.006	.744	CV1se	MNM	.998	.000	1.000
CVmin	MNM	.620	.006	.735	EBIC50	MNM	.996	.000	1.000
FGL	Sample-split	.617	.005	.574	stability	MNM	.996	.000	1.000
Prune	MNM	.603	.006	.705	EBIC25	MNM	.995	.000	1.000
BIC	MNM	.572	.006	.667	BIC	MNM	.992	.000	1.000
stability	MNM	.550	.007	.667	CVmin	MNM	.979	.001	1.000
EBIC25	MNM	.527	.007	.610	bootstrap	MNM	.977	.001	1.000
CV1se	MNM	.516	.006	.543	Prune	MNM	.976	.001	1.000
EBIC50	MNM	.485	.007	.500	AIC	MNM	.976	.001	1.000
NCT	Sample-split	.459	.006	.509	NCT	Sample-split	.937	.002	1.000
split	MNM	.436	.007	.398	FGL	Sample-split	.635	.005	.667

Table C.1: Grand means, standard errors, and medians for each model after collapsing across conditions. Rows differ across facets to reflect rank-ordered performance on each outcome. Each statistic is based on 4000 observations (100 iterations for each of 40 conditions).

C.2 Sensitivity (Interactions)

Average Sensitivity (SE) – Interactions

P = 5

Model	M = 10% N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	M = 30% N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.15 (.04)	.33 (.05)	.90 (.03)	.98 (.01)	1.00 (.00)	.67	.13 (.02)	.28 (.02)	.81 (.03)	.98 (.01)	1.00 (.00)	.64	Prune
AIC	.17 (.04)	.38 (.05)	.90 (.03)	1.00 (.00)	1.00 (.00)	.69	.15 (.02)	.30 (.03)	.82 (.03)	.99 (.01)	1.00 (.00)	.65	AIC
BIC	.12 (.03)	.18 (.04)	.67 (.05)	.97 (.02)	1.00 (.00)	.59	.10 (.02)	.16 (.02)	.64 (.03)	.94 (.01)	1.00 (.00)	.57	BIC
EBIC25	.10 (.03)	.12 (.03)	.50 (.05)	.95 (.02)	1.00 (.00)	.53	.08 (.02)	.11 (.02)	.53 (.04)	.91 (.02)	1.00 (.00)	.53	EBIC25
EBIC50	.06 (.02)	.09 (.03)	.38 (.05)	.90 (.03)	.98 (.01)	.48	.06 (.01)	.06 (.01)	.42 (.03)	.86 (.02)	1.00 (.00)	.48	EBIC50
CVmin	.17 (.04)	.38 (.05)	.90 (.03)	.99 (.01)	1.00 (.00)	.69	.15 (.02)	.29 (.03)	.81 (.03)	.99 (.01)	1.00 (.00)	.65	CVmin
CV1se	.10 (.03)	.15 (.04)	.32 (.05)	.62 (.05)	.91 (.03)	.42	.09 (.02)	.12 (.02)	.43 (.03)	.68 (.03)	.97 (.01)	.46	CV1se
split	.00 (.00)	.01 (.01)	.25 (.04)	.92 (.03)	1.00 (.00)	.44	.00 (.00)	.02 (.01)	.33 (.03)	.80 (.02)	1.00 (.00)	.43	split
bootstrap	.18 (.04)	.35 (.05)	.89 (.03)	1.00 (.00)	1.00 (.00)	.68	.15 (.02)	.30 (.03)	.82 (.02)	.98 (.01)	1.00 (.00)	.65	bootstrap
stability	.01 (.01)	.08 (.03)	.60 (.05)	.98 (.01)	1.00 (.00)	.53	.02 (.01)	.07 (.01)	.63 (.03)	.95 (.01)	1.00 (.00)	.53	stability
NCT	.05 (.02)	.16 (.04)	.41 (.05)	.92 (.03)	1.00 (.00)	.51	.08 (.02)	.18 (.03)	.57 (.04)	.90 (.02)	.99 (.00)	.54	NCT
FGL	.99 (.01)	.99 (.01)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	.99 (.00)	.99 (.01)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	FGL

Model	M = 20% N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	M = 50% N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.16 (.02)	.37 (.03)	.76 (.03)	.98 (.01)	1.00 (.00)	.65	.12 (.02)	.30 (.02)	.75 (.02)	.98 (.01)	1.00 (.00)	.63	Prune
AIC	.20 (.03)	.38 (.03)	.76 (.03)	1.00 (.00)	1.00 (.00)	.67	.14 (.02)	.30 (.02)	.75 (.02)	.99 (.01)	1.00 (.00)	.64	AIC
BIC	.12 (.02)	.13 (.02)	.57 (.04)	.93 (.02)	1.00 (.00)	.55	.07 (.01)	.16 (.02)	.60 (.03)	.95 (.01)	1.00 (.00)	.56	BIC
EBIC25	.09 (.02)	.10 (.02)	.48 (.04)	.90 (.02)	1.00 (.00)	.51	.05 (.01)	.11 (.02)	.50 (.03)	.91 (.02)	1.00 (.00)	.51	EBIC25
EBIC50	.08 (.02)	.06 (.02)	.40 (.04)	.84 (.03)	1.00 (.00)	.48	.05 (.01)	.08 (.01)	.40 (.03)	.86 (.02)	1.00 (.00)	.48	EBIC50
CVmin	.20 (.03)	.34 (.03)	.76 (.03)	.99 (.01)	1.00 (.00)	.66	.12 (.02)	.28 (.02)	.75 (.02)	.99 (.01)	1.00 (.00)	.63	CVmin
CV1se	.12 (.02)	.12 (.02)	.40 (.03)	.69 (.04)	.96 (.01)	.46	.07 (.01)	.14 (.02)	.44 (.02)	.79 (.02)	.97 (.01)	.48	CV1se
split	.00 (.00)	.04 (.01)	.26 (.03)	.84 (.03)	1.00 (.00)	.43	.00 (.00)	.02 (.01)	.26 (.02)	.79 (.02)	.99 (.00)	.41	split
bootstrap	.20 (.03)	.34 (.03)	.77 (.03)	.98 (.01)	1.00 (.00)	.66	.13 (.01)	.31 (.02)	.75 (.02)	.98 (.01)	1.00 (.00)	.63	bootstrap
stability	.04 (.01)	.10 (.02)	.51 (.04)	.96 (.01)	1.00 (.00)	.52	.01 (.00)	.09 (.01)	.60 (.02)	.97 (.01)	1.00 (.00)	.53	stability
NCT	.12 (.03)	.18 (.03)	.52 (.04)	.89 (.03)	1.00 (.00)	.54	.08 (.02)	.25 (.03)	.59 (.03)	.88 (.02)	1.00 (.00)	.56	NCT
FGL	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	.98 (.01)	.99 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99	FGL

P = 10

Model	M = 10% N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	M = 30% N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.06 (.01)	.26 (.02)	.74 (.02)	.98 (.01)	1.00 (.00)	.61	.04 (.01)	.18 (.01)	.59 (.01)	.87 (.01)	.97 (.01)	.53	Prune
AIC	.09 (.01)	.33 (.02)	.79 (.02)	.98 (.01)	1.00 (.00)	.64	.09 (.01)	.22 (.01)	.61 (.02)	.88 (.01)	.97 (.01)	.55	AIC
BIC	.05 (.01)	.13 (.01)	.61 (.02)	.95 (.01)	1.00 (.00)	.55	.05 (.01)	.11 (.01)	.52 (.02)	.85 (.01)	.97 (.01)	.50	BIC
EBIC25	.02 (.01)	.05 (.01)	.48 (.02)	.93 (.01)	1.00 (.00)	.50	.02 (.00)	.05 (.01)	.41 (.02)	.82 (.02)	.96 (.01)	.45	EBIC25
EBIC50	.01 (.01)	.02 (.01)	.32 (.02)	.89 (.02)	1.00 (.00)	.45	.01 (.00)	.02 (.00)	.30 (.02)	.78 (.02)	.96 (.01)	.42	EBIC50
CVmin	.09 (.01)	.33 (.02)	.78 (.02)	.98 (.01)	1.00 (.00)	.64	.07 (.01)	.20 (.01)	.60 (.02)	.87 (.01)	.97 (.01)	.54	CVmin
CV1se	.04 (.01)	.15 (.02)	.57 (.02)	.90 (.01)	.99 (.00)	.53	.04 (.00)	.13 (.01)	.46 (.02)	.74 (.02)	.94 (.01)	.46	CV1se
split	.00 (.00)	.01 (.00)	.25 (.02)	.83 (.02)	1.00 (.00)	.42	.00 (.00)	.01 (.00)	.20 (.01)	.61 (.02)	.94 (.01)	.35	split
bootstrap	.14 (.01)	.34 (.02)	.80 (.02)	.99 (.01)	1.00 (.00)	.66	.11 (.01)	.25 (.01)	.63 (.01)	.89 (.01)	.97 (.01)	.57	bootstrap
stability	.04 (.01)	.09 (.01)	.62 (.02)	.97 (.01)	1.00 (.00)	.54	.02 (.00)	.10 (.01)	.52 (.02)	.86 (.01)	.97 (.01)	.49	stability
NCT	.00 (.00)	.06 (.02)	.45 (.04)	.91 (.02)	1.00 (.00)	.48	.03 (.01)	.18 (.02)	.65 (.02)	.93 (.01)	1.00 (.00)	.56	NCT
FGL	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	1.00 (.00)	.99 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	FGL

Model	M = 20% N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	M = 50% N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.06 (.01)	.24 (.01)	.69 (.01)	.93 (.01)	.99 (.00)	.58	.04 (.00)	.16 (.01)	.51 (.01)	.81 (.01)	.91 (.01)	.49	Prune
AIC	.10 (.01)	.26 (.01)	.73 (.02)	.93 (.01)	.99 (.00)	.60	.07 (.01)	.20 (.01)	.52 (.01)	.82 (.01)	.90 (.01)	.50	AIC
BIC	.04 (.01)	.11 (.01)	.61 (.02)	.91 (.01)	.98 (.01)	.53	.04 (.00)	.11 (.01)	.46 (.01)	.80 (.01)	.90 (.01)	.46	BIC
EBIC25	.02 (.00)	.05 (.01)	.48 (.02)	.88 (.02)	.98 (.01)	.48	.02 (.00)	.06 (.01)	.40 (.01)	.78 (.01)	.89 (.01)	.43	EBIC25
EBIC50	.01 (.00)	.03 (.01)	.31 (.02)	.83 (.02)	.98 (.01)	.43	.01 (.00)	.04 (.00)	.29 (.01)	.73 (.01)	.89 (.01)	.39	EBIC50
CVmin	.08 (.01)	.27 (.01)	.73 (.02)	.93 (.01)	.98 (.00)	.60	.06 (.00)	.18 (.01)	.50 (.01)	.79 (.02)	.86 (.02)	.48	CVmin
CV1se	.04 (.01)	.13 (.01)	.59 (.02)	.83 (.02)	.97 (.01)	.51	.04 (.00)	.11 (.01)	.37 (.01)	.65 (.02)	.76 (.02)	.38	CV1se
split	.00 (.00)	.01 (.00)	.27 (.01)	.73 (.02)	.96 (.01)	.39	.00 (.00)	.01 (.00)	.19 (.01)	.57 (.01)	.87 (.01)	.33	split
bootstrap	.12 (.01)	.29 (.01)	.74 (.01)	.93 (.01)	.99 (.00)	.61	.10 (.01)	.23 (.01)	.54 (.01)	.83 (.01)	.92 (.01)	.52	bootstrap
stability	.03 (.01)	.08 (.01)	.62 (.02)	.92 (.01)	.98 (.01)	.53	.02 (.00)	.09 (.01)	.48 (.01)	.79 (.01)	.89 (.01)	.45	stability
NCT	.00 (.00)	.04 (.01)	.50 (.03)	.93 (.01)	1.00 (.00)	.49	.05 (.01)	.25 (.02)	.66 (.02)	.88 (.01)	.96 (.01)	.56	NCT
FGL	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	.99 (.00)	.99 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99	FGL

Table C.2: Means and standard errors across models and conditions for *sensitivity*. The values essentially reflect the same information shown in Figure 4.1.

C.3 False-Discovery Rate (Interactions)

Average MAE (*SE*) – Interactions

P = 5

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.86 (.03)	.71 (.04)	.18 (.03)	.14 (.03)	.12 (.02)	.40
AIC	.84 (.04)	.66 (.05)	.18 (.03)	.11 (.02)	.12 (.02)	.38
BIC	.88 (.03)	.82 (.04)	.34 (.05)	.04 (.02)	.03 (.01)	.42
EBIC25	.90 (.03)	.88 (.03)	.50 (.05)	.06 (.02)	.02 (.01)	.47
EBIC50	.94 (.02)	.91 (.03)	.62 (.05)	.10 (.03)	.03 (.02)	.52
CVmin	.84 (.04)	.66 (.05)	.19 (.03)	.11 (.02)	.12 (.02)	.39
CV1se	.90 (.03)	.85 (.04)	.68 (.05)	.38 (.05)	.09 (.03)	.58
split	1.00 (.00)	.99 (.01)	.75 (.04)	.08 (.03)	.00 (.00)	.56
bootstrap	.82 (.04)	.68 (.05)	.18 (.03)	.11 (.02)	.12 (.02)	.38
stability	.99 (.01)	.92 (.03)	.40 (.05)	.04 (.02)	.03 (.01)	.47
NCT	.98 (.01)	.88 (.03)	.66 (.04)	.30 (.03)	.16 (.03)	.59
FGL	.16 (.03)	.28 (.03)	.42 (.04)	.73 (.02)	.78 (.02)	.47

M = 30%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.67 (.05)	.39 (.05)	.08 (.02)	.04 (.01)	.04 (.01)	.24
AIC	.65 (.05)	.36 (.05)	.07 (.02)	.04 (.01)	.04 (.01)	.23
BIC	.75 (.04)	.59 (.05)	.12 (.03)	.01 (.00)	.02 (.01)	.30
EBIC25	.77 (.04)	.69 (.05)	.19 (.04)	.01 (.00)	.01 (.01)	.33
EBIC50	.82 (.04)	.81 (.04)	.26 (.04)	.02 (.01)	.01 (.01)	.38
CVmin	.64 (.05)	.39 (.05)	.08 (.02)	.04 (.01)	.04 (.01)	.24
CV1se	.75 (.04)	.66 (.05)	.23 (.04)	.05 (.02)	.00 (.00)	.34
split	1.00 (.00)	.94 (.02)	.31 (.05)	.00 (.00)	.00 (.00)	.45
bootstrap	.64 (.05)	.35 (.05)	.06 (.02)	.04 (.01)	.04 (.01)	.22
stability	.94 (.02)	.79 (.04)	.08 (.03)	.01 (.01)	.02 (.01)	.37
NCT	.89 (.03)	.73 (.04)	.39 (.04)	.13 (.02)	.09 (.01)	.44
FGL	.10 (.02)	.17 (.02)	.33 (.03)	.54 (.02)	.61 (.01)	.35

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.72 (.04)	.44 (.05)	.11 (.03)	.05 (.01)	.06 (.01)	.28
AIC	.64 (.05)	.41 (.05)	.11 (.03)	.05 (.01)	.06 (.01)	.25
BIC	.76 (.04)	.74 (.04)	.22 (.04)	.03 (.01)	.00 (.00)	.35
EBIC25	.82 (.04)	.80 (.04)	.30 (.05)	.03 (.01)	.00 (.00)	.39
EBIC50	.84 (.04)	.87 (.03)	.39 (.05)	.06 (.02)	.00 (.00)	.43
CVmin	.64 (.05)	.44 (.05)	.11 (.03)	.05 (.01)	.06 (.01)	.26
CV1se	.78 (.04)	.78 (.04)	.34 (.05)	.17 (.04)	.00 (.00)	.41
split	1.00 (.00)	.93 (.03)	.54 (.05)	.02 (.01)	.00 (.00)	.50
bootstrap	.64 (.05)	.47 (.05)	.11 (.03)	.04 (.01)	.05 (.01)	.26
stability	.92 (.03)	.80 (.04)	.28 (.04)	.01 (.00)	.00 (.00)	.40
NCT	.90 (.03)	.80 (.04)	.46 (.04)	.23 (.03)	.13 (.02)	.50
FGL	.15 (.02)	.20 (.03)	.38 (.03)	.61 (.02)	.71 (.01)	.41

M = 50%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.57 (.05)	.24 (.04)	.03 (.01)	.01 (.00)	.02 (.01)	.17
AIC	.50 (.05)	.24 (.04)	.03 (.01)	.01 (.00)	.02 (.01)	.16
BIC	.71 (.05)	.43 (.05)	.04 (.02)	.01 (.00)	.01 (.00)	.24
EBIC25	.79 (.04)	.58 (.05)	.05 (.02)	.01 (.00)	.01 (.00)	.29
EBIC50	.80 (.04)	.69 (.05)	.12 (.03)	.01 (.00)	.01 (.00)	.33
CVmin	.52 (.05)	.25 (.04)	.03 (.01)	.01 (.00)	.02 (.01)	.17
CV1se	.71 (.05)	.49 (.05)	.05 (.02)	.01 (.01)	.00 (.00)	.25
split	.99 (.01)	.89 (.03)	.20 (.04)	.00 (.00)	.00 (.00)	.42
bootstrap	.50 (.05)	.23 (.04)	.02 (.01)	.01 (.00)	.02 (.01)	.16
stability	.95 (.02)	.66 (.05)	.01 (.01)	.00 (.00)	.01 (.00)	.33
NCT	.85 (.03)	.55 (.05)	.21 (.03)	.10 (.02)	.04 (.01)	.35
FGL	.06 (.01)	.10 (.01)	.21 (.02)	.35 (.01)	.42 (.01)	.23

P = 10

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.80 (.04)	.41 (.04)	.18 (.02)	.14 (.01)	.13 (.01)	.33
AIC	.73 (.04)	.31 (.03)	.15 (.02)	.13 (.01)	.12 (.01)	.29
BIC	.78 (.04)	.48 (.05)	.03 (.01)	.04 (.01)	.02 (.01)	.27
EBIC25	.92 (.03)	.79 (.04)	.05 (.02)	.02 (.01)	.01 (.00)	.36
EBIC50	.93 (.03)	.91 (.03)	.17 (.04)	.02 (.00)	.01 (.00)	.41
CVmin	.70 (.04)	.26 (.03)	.15 (.02)	.13 (.01)	.12 (.01)	.27
CV1se	.81 (.04)	.43 (.05)	.03 (.02)	.00 (.00)	.00 (.00)	.25
split	1.00 (.00)	.96 (.02)	.20 (.04)	.00 (.00)	.00 (.00)	.43
bootstrap	.63 (.04)	.31 (.03)	.13 (.01)	.11 (.01)	.13 (.01)	.26
stability	.84 (.04)	.63 (.05)	.02 (.01)	.02 (.01)	.01 (.00)	.30
NCT	.98 (.01)	.90 (.03)	.60 (.03)	.39 (.02)	.30 (.01)	.64
FGL	.02 (.01)	.07 (.01)	.33 (.03)	.77 (.02)	.83 (.01)	.40

M = 30%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.64 (.04)	.23 (.03)	.08 (.01)	.06 (.01)	.07 (.01)	.22
AIC	.39 (.04)	.14 (.02)	.08 (.01)	.06 (.01)	.07 (.01)	.15
BIC	.52 (.05)	.25 (.04)	.03 (.01)	.03 (.01)	.03 (.00)	.17
EBIC25	.74 (.04)	.49 (.05)	.01 (.00)	.02 (.00)	.02 (.00)	.26
EBIC50	.82 (.04)	.74 (.04)	.04 (.02)	.01 (.00)	.02 (.00)	.33
CVmin	.44 (.05)	.13 (.02)	.06 (.01)	.06 (.01)	.07 (.01)	.15
CV1se	.52 (.05)	.17 (.03)	.02 (.01)	.00 (.00)	.00 (.00)	.14
split	1.00 (.00)	.86 (.03)	.04 (.02)	.00 (.00)	.00 (.00)	.38
bootstrap	.40 (.04)	.13 (.02)	.06 (.01)	.06 (.01)	.06 (.01)	.14
stability	.74 (.04)	.20 (.04)	.01 (.00)	.01 (.00)	.02 (.00)	.20
NCT	.94 (.02)	.70 (.04)	.31 (.02)	.17 (.01)	.12 (.01)	.45
FGL	.03 (.01)	.15 (.02)	.44 (.02)	.62 (.00)	.64 (.00)	.38

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.63 (.05)	.22 (.03)	.08 (.01)	.09 (.01)	.10 (.01)	.23
AIC	.48 (.04)	.21 (.03)	.08 (.01)	.08 (.01)	.09 (.01)	.19
BIC	.71 (.05)	.40 (.05)	.02 (.01)	.03 (.01)	.03 (.01)	.24
EBIC25	.86 (.03)	.64 (.05)	.04 (.02)	.02 (.00)	.02 (.01)	.32
EBIC50	.92 (.03)	.81 (.04)	.06 (.02)	.01 (.00)	.02 (.01)	.36
CVmin	.52 (.05)	.17 (.03)	.08 (.01)	.08 (.01)	.09 (.01)	.19
CV1se	.68 (.05)	.38 (.05)	.01 (.00)	.00 (.00)	.00 (.00)	.22
split	1.00 (.00)	.89 (.03)	.05 (.02)	.00 (.00)	.00 (.00)	.39
bootstrap	.44 (.04)	.19 (.03)	.07 (.01)	.09 (.01)	.09 (.01)	.17
stability	.74 (.04)	.51 (.05)	.01 (.00)	.02 (.01)	.03 (.01)	.26
NCT	.96 (.02)	.88 (.03)	.43 (.03)	.26 (.01)	.20 (.01)	.55
FGL	.01 (.00)	.05 (.01)	.30 (.03)	.72 (.01)	.75 (.00)	.37

M = 50%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.40 (.05)	.10 (.02)	.05 (.01)	.06 (.01)	.08 (.01)	.14
AIC	.24 (.04)	.06 (.01)	.06 (.01)	.05 (.01)	.08 (.01)	.10
BIC	.46 (.05)	.09 (.02)	.04 (.01)	.04 (.00)	.06 (.01)	.13
EBIC25	.62 (.05)	.26 (.04)	.03 (.01)	.03 (.00)	.05 (.01)	.20
EBIC50	.75 (.04)	.40 (.05)	.03 (.01)	.03 (.00)	.04 (.01)	.25
CVmin	.30 (.04)	.05 (.01)	.05 (.01)	.05 (.01)	.06 (.01)	.10
CV1se	.44 (.05)	.06 (.02)	.02 (.01)	.02 (.01)	.01 (.00)	.11
split	1.00 (.00)	.78 (.04)	.01 (.01)	.00 (.00)	.02 (.00)	.36
bootstrap	.18 (.03)	.07 (.01)	.05 (.01)	.05 (.01)	.08 (.01)	.09
stability	.61 (.05)	.07 (.03)	.02 (.01)	.01 (.00)	.02 (.00)	.15
NCT	.87 (.03)	.42 (.04)	.16 (.01)	.09 (.01)	.06 (.01)	.32
FGL	.04 (.01)	.14 (.02)	.37 (.01)	.43 (.00)	.46 (.00)	.29

Table C.3: Means and standard errors across models and conditions for *FDR*. The values essentially reflect the same information shown in Figure 4.2.

C.4 Matthew's Correlation Coefficient (Interactions)

Average MCC (*SE*) – Interactions

P = 5

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.13 (.03)	.29 (.04)	.85 (.03)	.89 (.02)	.92 (.02)	.62
AIC	.14 (.04)	.35 (.05)	.84 (.03)	.92 (.02)	.92 (.02)	.63
BIC	.11 (.03)	.17 (.04)	.66 (.05)	.96 (.02)	.98 (.01)	.58
EBIC25	.09 (.03)	.11 (.03)	.50 (.05)	.95 (.02)	.99 (.01)	.53
EBIC50	.05 (.02)	.09 (.03)	.38 (.05)	.90 (.03)	.97 (.02)	.48
CVmin	.15 (.04)	.35 (.05)	.84 (.03)	.92 (.02)	.92 (.02)	.63
CV1se	.09 (.03)	.15 (.04)	.32 (.05)	.62 (.05)	.91 (.03)	.42
split	.00 (.00)	.01 (.01)	.25 (.04)	.92 (.03)	1.00 (.00)	.44
bootstrap	.16 (.04)	.32 (.05)	.83 (.03)	.93 (.01)	.92 (.02)	.63
stability	.01 (.01)	.08 (.03)	.60 (.05)	.97 (.02)	.98 (.01)	.53
NCT	.02 (.02)	.11 (.03)	.35 (.05)	.77 (.03)	.89 (.02)	.43
FGL	.88 (.02)	.79 (.03)	.67 (.03)	.36 (.02)	.31 (.02)	.60

M = 30%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.15 (.03)	.35 (.03)	.81 (.02)	.95 (.01)	.97 (.01)	.65	Prune
AIC	.17 (.03)	.37 (.03)	.83 (.02)	.96 (.01)	.97 (.01)	.66	AIC
BIC	.13 (.03)	.22 (.03)	.70 (.03)	.95 (.01)	.99 (.00)	.60	BIC
EBIC25	.12 (.02)	.16 (.03)	.60 (.03)	.93 (.01)	.99 (.00)	.56	EBIC25
EBIC50	.09 (.02)	.09 (.02)	.50 (.03)	.89 (.02)	.99 (.00)	.51	EBIC50
CVmin	.17 (.03)	.35 (.03)	.82 (.02)	.96 (.01)	.97 (.01)	.65	CVmin
CV1se	.12 (.02)	.17 (.03)	.52 (.03)	.76 (.02)	.97 (.01)	.51	CV1se
split	.00 (.00)	.03 (.01)	.43 (.03)	.86 (.02)	1.00 (.00)	.46	split
bootstrap	.18 (.03)	.38 (.03)	.83 (.02)	.96 (.01)	.97 (.01)	.66	bootstrap
stability	.03 (.01)	.10 (.02)	.71 (.03)	.95 (.01)	.99 (.00)	.56	stability
NCT	.06 (.02)	.18 (.03)	.49 (.03)	.82 (.02)	.92 (.01)	.49	NCT
FGL	.90 (.02)	.82 (.03)	.68 (.03)	.42 (.02)	.32 (.02)	.63	FGL

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.18 (.03)	.41 (.04)	.78 (.03)	.95 (.01)	.95 (.01)	.66
AIC	.22 (.03)	.44 (.04)	.79 (.03)	.96 (.01)	.96 (.01)	.67
BIC	.16 (.03)	.17 (.03)	.64 (.04)	.94 (.02)	1.00 (.00)	.58
EBIC25	.12 (.03)	.13 (.03)	.55 (.04)	.92 (.02)	1.00 (.00)	.54
EBIC50	.11 (.02)	.08 (.02)	.47 (.04)	.87 (.03)	1.00 (.00)	.51
CVmin	.23 (.03)	.40 (.04)	.78 (.03)	.96 (.01)	.96 (.01)	.67
CV1se	.15 (.03)	.15 (.03)	.49 (.04)	.73 (.04)	.97 (.01)	.50
split	.00 (.00)	.05 (.02)	.33 (.04)	.88 (.02)	1.00 (.00)	.45
bootstrap	.23 (.03)	.38 (.04)	.79 (.03)	.96 (.01)	.97 (.01)	.67
stability	.05 (.02)	.13 (.03)	.58 (.04)	.97 (.01)	1.00 (.00)	.55
NCT	.09 (.02)	.15 (.03)	.47 (.04)	.77 (.03)	.90 (.01)	.47
FGL	.88 (.02)	.84 (.02)	.67 (.03)	.43 (.02)	.30 (.02)	.62

M = 50%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.15 (.02)	.34 (.03)	.76 (.02)	.97 (.01)	.98 (.01)	.64	Prune
AIC	.18 (.02)	.35 (.03)	.75 (.02)	.97 (.01)	.98 (.01)	.65	AIC
BIC	.10 (.02)	.22 (.02)	.64 (.02)	.94 (.01)	.99 (.00)	.58	BIC
EBIC25	.07 (.02)	.16 (.02)	.56 (.02)	.91 (.01)	.99 (.00)	.54	EBIC25
EBIC50	.07 (.02)	.12 (.02)	.47 (.03)	.88 (.02)	.99 (.00)	.50	EBIC50
CVmin	.17 (.02)	.33 (.03)	.75 (.02)	.97 (.01)	.98 (.01)	.64	CVmin
CV1se	.10 (.02)	.19 (.02)	.52 (.02)	.81 (.02)	.97 (.01)	.52	CV1se
split	.00 (.00)	.04 (.01)	.35 (.02)	.82 (.01)	.99 (.00)	.44	split
bootstrap	.17 (.02)	.35 (.03)	.76 (.02)	.97 (.01)	.98 (.01)	.65	bootstrap
stability	.02 (.01)	.12 (.02)	.65 (.02)	.97 (.01)	.99 (.01)	.55	stability
NCT	.05 (.02)	.22 (.03)	.51 (.03)	.79 (.02)	.95 (.01)	.50	NCT
FGL	.91 (.02)	.87 (.02)	.71 (.03)	.49 (.03)	.33 (.02)	.66	FGL

P = 10

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.08 (.02)	.35 (.03)	.74 (.02)	.90 (.01)	.92 (.01)	.60
AIC	.12 (.02)	.43 (.02)	.79 (.01)	.91 (.01)	.93 (.01)	.63
BIC	.10 (.02)	.24 (.02)	.74 (.02)	.95 (.01)	.99 (.00)	.60
EBIC25	.03 (.01)	.10 (.02)	.64 (.02)	.94 (.01)	.99 (.00)	.54
EBIC50	.03 (.01)	.04 (.01)	.49 (.03)	.92 (.01)	1.00 (.00)	.49
CVmin	.14 (.02)	.44 (.02)	.78 (.01)	.91 (.01)	.93 (.01)	.64
CV1se	.08 (.02)	.28 (.03)	.71 (.02)	.94 (.01)	.99 (.00)	.60
split	.00 (.00)	.02 (.01)	.42 (.02)	.89 (.01)	1.00 (.00)	.47
bootstrap	.18 (.02)	.44 (.02)	.81 (.01)	.93 (.01)	.92 (.01)	.65
stability	.07 (.02)	.17 (.02)	.75 (.02)	.97 (.01)	.99 (.00)	.59
NCT	.01 (.01)	.06 (.02)	.35 (.03)	.69 (.02)	.80 (.01)	.38
FGL	.99 (.00)	.95 (.01)	.74 (.03)	.31 (.02)	.24 (.01)	.65

M = 30%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.07 (.01)	.29 (.02)	.66 (.01)	.86 (.01)	.93 (.01)	.56	Prune
AIC	.15 (.01)	.34 (.01)	.67 (.01)	.87 (.01)	.93 (.01)	.59	AIC
BIC	.12 (.01)	.23 (.02)	.63 (.01)	.88 (.01)	.95 (.01)	.56	BIC
EBIC25	.06 (.01)	.13 (.01)	.56 (.01)	.86 (.01)	.96 (.01)	.51	EBIC25
EBIC50	.04 (.01)	.06 (.01)	.46 (.02)	.83 (.01)	.96 (.01)	.47	EBIC50
CVmin	.14 (.01)	.33 (.01)	.67 (.01)	.87 (.01)	.93 (.01)	.59	CVmin
CV1se	.11 (.01)	.27 (.01)	.59 (.01)	.81 (.01)	.95 (.01)	.55	CV1se
split	.00 (.00)	.03 (.01)	.37 (.01)	.72 (.01)	.95 (.01)	.41	split
bootstrap	.16 (.02)	.37 (.01)	.70 (.01)	.88 (.01)	.93 (.01)	.61	bootstrap
stability	.06 (.01)	.23 (.01)	.64 (.01)	.89 (.01)	.96 (.01)	.56	stability
NCT	.01 (.00)	.14 (.02)	.53 (.02)	.81 (.01)	.90 (.01)	.48	NCT
FGL	.97 (.02)	.84 (.03)	.55 (.03)	.31 (.01)	.26 (.01)	.58	FGL

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.11 (.02)	.36 (.02)	.75 (.01)	.90 (.01)	.92 (.01)	.61
AIC	.17 (.02)	.38 (.02)	.78 (.01)	.90 (.01)	.93 (.01)	.63
BIC	.09 (.02)	.22 (.02)	.73 (.01)	.92 (.01)	.97 (.01)	.59
EBIC25	.04 (.01)	.12 (.02)	.63 (.02)	.91 (.01)	.97 (.01)	.54
EBIC50	.02 (.01)	.06 (.01)	.49 (.02)	.89 (.01)	.97 (.01)	.49
CVmin	.16 (.02)	.40 (.02)	.77 (.01)	.90 (.01)	.93 (.01)	.63
CV1se	.10 (.02)	.25 (.02)	.72 (.01)	.89 (.01)	.98 (.01)	.59
split	.00 (.00)	.03 (.01)	.45 (.02)	.82 (.01)	.98 (.01)	.46
bootstrap	.19 (.02)	.41 (.02)	.80 (.01)	.90 (.01)	.93 (.01)	.65
stability	.08 (.01)	.18 (.02)	.74 (.01)	.94 (.01)	.97 (.01)	.58
NCT	.01 (.01)	.05 (.01)	.42 (.03)	.77 (.01)	.86 (.01)	.42
FGL	.99 (.00)	.96 (.01)	.75 (.03)	.30 (.01)	.24 (.01)	.65

M = 50%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.09 (.01)	.25 (.01)	.53 (.01)	.77 (.01)	.83 (.01)	.49	Prune
AIC	.13 (.01)	.29 (.01)	.54 (.01)	.77 (.01)	.82 (.02)	.51	AIC
BIC	.09 (.01)	.21 (.01)	.52 (.01)	.78 (.01)	.85 (.02)	.49	BIC
EBIC25	.06 (.01)	.14 (.01)	.47 (.01)	.77 (.01)	.85 (.02)	.46	EBIC25
EBIC50	.04 (.01)	.10 (.01)	.39 (.01)	.74 (.01)	.86 (.02)	.42	EBIC50
CVmin	.12 (.01)	.28 (.01)	.54 (.01)	.76 (.01)	.82 (.02)	.50	CVmin
CV1se	.09 (.01)	.23 (.01)	.45 (.01)	.69 (.02)	.78 (.02)	.45	CV1se
split	.00 (.00)	.03 (.01)	.31 (.01)	.62 (.01)	.86 (.01)	.37	split
bootstrap	.16 (.01)	.31 (.01)	.56 (.01)	.79 (.01)	.84 (.01)	.53	bootstrap
stability	.07 (.01)	.20 (.01)	.55 (.01)	.80 (.01)	.87 (.01)	.49	stability
NCT	.04 (.01)	.21 (.02)	.54 (.01)	.79 (.01)	.90 (.01)	.49	NCT
FGL	.93 (.02)	.78 (.03)	.46 (.02)	.31 (.02)	.22 (.01)	.54	FGL

Table C.4: Means and standard errors across models and conditions for *MCC*. The values essentially reflect the same information shown in Figure 4.3.

C.5 Specificity (Interactions)

Average Specificity (*SE*) – Interactions

P = 5

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.98 (.00)	.98 (.00)	.98 (.01)	.97 (.01)	.97 (.01)	.98
AIC	.98 (.00)	.98 (.00)	.98 (.01)	.97 (.01)	.97 (.01)	.98
BIC	.99 (.00)	.99 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	.99
EBIC25	.99 (.00)	.99 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
EBIC50	.99 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
CVmin	.98 (.00)	.98 (.00)	.97 (.01)	.97 (.01)	.97 (.01)	.98
CV1se	.99 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
split	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
bootstrap	.98 (.00)	.98 (.00)	.98 (.01)	.97 (.01)	.97 (.01)	.98
stability	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	1.00
NCT	.97 (.01)	.96 (.01)	.97 (.01)	.94 (.01)	.96 (.01)	.96
FGL	.95 (.01)	.90 (.02)	.80 (.02)	.52 (.02)	.44 (.02)	.72

M = 30%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.98 (.01)	.99 (.00)	.97 (.01)	.97 (.01)	.98 (.01)	.98	Prune
AIC	.98 (.01)	.98 (.01)	.98 (.01)	.98 (.01)	.98 (.01)	.98	AIC
BIC	.99 (.00)	.99 (.00)	1.00 (.00)	.99 (.00)	.99 (.00)	.99	BIC
EBIC25	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	.99 (.00)	1.00	EBIC25
EBIC50	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	1.00	EBIC50
CVmin	.98 (.01)	.98 (.01)	.97 (.01)	.98 (.01)	.98 (.01)	.98	CVmin
CV1se	.99 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	CV1se
split	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	split
bootstrap	.98 (.01)	.99 (.00)	.98 (.01)	.98 (.01)	.98 (.01)	.98	bootstrap
stability	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	.99 (.00)	1.00	stability
NCT	.97 (.01)	.97 (.01)	.89 (.02)	.92 (.01)	.95 (.01)	.94	NCT
FGL	.92 (.02)	.85 (.02)	.69 (.03)	.41 (.03)	.29 (.02)	.63	FGL

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.98 (.00)	.98 (.01)	.98 (.01)	.98 (.01)	.98 (.01)	.98
AIC	.98 (.00)	.99 (.00)	.98 (.01)	.98 (.01)	.98 (.01)	.98
BIC	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	1.00 (.00)	1.00
EBIC25	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
EBIC50	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
CVmin	.98 (.00)	.98 (.00)	.98 (.01)	.98 (.01)	.98 (.00)	.98
CV1se	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
split	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
bootstrap	.98 (.00)	.99 (.00)	.98 (.01)	.98 (.01)	.98 (.00)	.98
stability	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
NCT	.97 (.01)	.95 (.01)	.93 (.01)	.90 (.01)	.95 (.01)	.94
FGL	.92 (.01)	.88 (.02)	.74 (.03)	.50 (.02)	.33 (.02)	.67

M = 50%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.99 (.00)	.98 (.01)	.97 (.01)	.98 (.01)	.98 (.01)	.98	Prune
AIC	.99 (.00)	.98 (.01)	.97 (.01)	.98 (.01)	.98 (.01)	.98	AIC
BIC	1.00 (.00)	.99 (.00)	.98 (.01)	.99 (.00)	.99 (.01)	.99	BIC
EBIC25	1.00 (.00)	1.00 (.00)	.99 (.00)	.99 (.00)	.99 (.01)	.99	EBIC25
EBIC50	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	.99 (.00)	1.00	EBIC50
CVmin	.99 (.00)	.98 (.01)	.97 (.01)	.98 (.01)	.98 (.01)	.98	CVmin
CV1se	1.00 (.00)	.99 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	CV1se
split	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	split
bootstrap	.99 (.00)	.98 (.01)	.98 (.01)	.99 (.01)	.98 (.01)	.98	bootstrap
stability	1.00 (.00)	1.00 (.00)	.99 (.00)	.99 (.00)	.99 (.01)	.99	stability
NCT	.96 (.01)	.94 (.01)	.87 (.02)	.88 (.02)	.94 (.01)	.92	NCT
FGL	.91 (.02)	.86 (.02)	.67 (.03)	.42 (.03)	.26 (.02)	.62	FGL

P = 10

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.99 (.00)	.98 (.00)	.98 (.00)	.98 (.00)	.98 (.00)	.98
AIC	.98 (.00)	.98 (.00)	.98 (.00)	.98 (.00)	.98 (.00)	.98
BIC	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	1.00 (.00)	1.00
EBIC25	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
EBIC50	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
CVmin	.99 (.00)	.99 (.00)	.98 (.00)	.98 (.00)	.98 (.00)	.98
CV1se	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
split	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
bootstrap	.97 (.00)	.98 (.00)	.98 (.00)	.98 (.00)	.98 (.00)	.98
stability	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
NCT	1.00 (.00)	.99 (.01)	.91 (.01)	.91 (.01)	.94 (.00)	.95
FGL	1.00 (.00)	.99 (.00)	.85 (.02)	.43 (.02)	.34 (.01)	.72

M = 30%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.99 (.00)	.98 (.00)	.98 (.00)	.97 (.00)	.97 (.00)	.98	Prune
AIC	.98 (.00)	.98 (.00)	.98 (.00)	.97 (.00)	.97 (.00)	.98	AIC
BIC	1.00 (.00)	1.00 (.00)	.99 (.00)	.99 (.00)	.99 (.00)	.99	BIC
EBIC25	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	.99 (.00)	1.00	EBIC25
EBIC50	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	1.00	EBIC50
CVmin	.99 (.00)	.99 (.00)	.98 (.00)	.98 (.00)	.97 (.00)	.98	CVmin
CV1se	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	CV1se
split	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	split
bootstrap	.97 (.00)	.98 (.00)	.98 (.00)	.97 (.00)	.97 (.00)	.98	bootstrap
stability	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	1.00	stability
NCT	.98 (.01)	.94 (.01)	.86 (.01)	.91 (.01)	.93 (.01)	.92	NCT
FGL	.97 (.01)	.86 (.02)	.54 (.03)	.26 (.01)	.19 (.01)	.57	FGL

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.99 (.00)	.99 (.00)	.98 (.00)	.98 (.00)	.97 (.00)	.98
AIC	.98 (.00)	.98 (.00)	.98 (.00)	.98 (.00)	.97 (.00)	.98
BIC	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	.99 (.00)	1.00
EBIC25	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	1.00
EBIC50	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
CVmin	.99 (.00)	.99 (.00)	.98 (.00)	.98 (.00)	.97 (.00)	.98
CV1se	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
split	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00
bootstrap	.98 (.00)	.98 (.00)	.99 (.00)	.98 (.00)	.97 (.00)	.98
stability	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	.99 (.00)	1.00
NCT	1.00 (.00)	.99 (.00)	.89 (.01)	.90 (.01)	.93 (.01)	.94
FGL	1.00 (.00)	.98 (.01)	.81 (.02)	.33 (.01)	.24 (.01)	.67

M = 50%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.99 (.00)	.98 (.00)	.97 (.00)	.95 (.01)	.92 (.01)	.96	Prune
AIC	.99 (.00)	.99 (.00)	.97 (.00)	.95 (.01)	.92 (.01)	.96	AIC
BIC	1.00 (.00)	.99 (.00)	.98 (.00)	.97 (.00)	.94 (.01)	.98	BIC
EBIC25	1.00 (.00)	1.00 (.00)	.99 (.00)	.98 (.00)	.95 (.01)	.98	EBIC25
EBIC50	1.00 (.00)	1.00 (.00)	.99 (.00)	.98 (.00)	.96 (.01)	.99	EBIC50
CVmin	.99 (.00)	.99 (.00)	.98 (.00)	.96 (.00)	.94 (.01)	.97	CVmin
CV1se	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00	CV1se
split	1.00 (.00)	1.00 (.00)	1.00 (.00)	1.00 (.00)	.99 (.00)	1.00	split
bootstrap	.98 (.00)	.98 (.00)	.97 (.00)	.95 (.00)	.92 (.01)	.96	bootstrap
stability	1.00 (.00)	1.00 (.00)	.99 (.00)	.99 (.00)	.98 (.00)	.99	stability
NCT	.98 (.01)	.91 (.01)	.85 (.01)	.91 (.01)	.94 (.01)	.92	NCT
FGL	.94 (.02)	.77 (.03)	.36 (.02)	.20 (.01)	.12 (.01)	.48	FGL

Table C.5: Means and standard errors across models and conditions for *specificity*. The values essentially reflect the same information shown in Figure B.1.

Appendix D

Simulation Study—Secondary Analysis Means

D.1 Grand Means

Correlation							
Pairwise				Interactions			
Model	*M*	*SE*	*Mdn*	Model	*M*	*SE*	*Mdn*
bootstrap	.726	.004	.805	bootstrap	.680	.006	.834
AIC	.725	.004	.806	AIC	.672	.006	.830
CVmin	.721	.004	.803	CVmin	.664	.006	.819
BIC	.716	.004	.789	Prune	.650	.006	.810
Prune	.711	.004	.806	BIC	.598	.006	.745
stability	.706	.004	.782	stability	.569	.007	.745
EBIC25	.705	.004	.768	EBIC25	.549	.007	.673
EBIC50	.693	.004	.752	CV1se	.542	.006	.647
CV1se	.688	.004	.750	EBIC50	.504	.007	.565
split	.520	.006	.607	split	.453	.007	.455
MAE							
Pairwise				Interactions			
Model	*M*	*SE*	*Mdn*	Model	*M*	*SE*	*Mdn*
CVmin	.063	.001	.058	CVmin	.033	.000	.022
BIC	.063	.001	.059	AIC	.033	.000	.022
AIC	.064	.001	.057	bootstrap	.033	.000	.023
EBIC25	.064	.001	.062	BIC	.034	.000	.022
bootstrap	.065	.001	.059	stability	.034	.000	.022
stability	.065	.001	.060	Prune	.034	.000	.023
Prune	.065	.001	.058	EBIC25	.035	.000	.024
EBIC50	.066	.001	.065	CV1se	.036	.000	.027
CV1se	.066	.001	.065	EBIC50	.037	.000	.026
split	.072	.001	.082	split	.038	.000	.031

Table D.1: Grand means, standard errors, and medians for each model after collapsing across conditions. Rows differ across facets to reflect rank-ordered performance on each outcome. Each statistic is based on 4000 observations (100 iterations for each of 40 conditions).

D.2 Correlation (Pairwise)

Average Correlations (*SE*) – Pairwise

P = 5

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.38 (.03)	.58 (.02)	.85 (.01)	.91 (.01)	.92 (.01)	.73
AIC	.43 (.03)	.58 (.03)	.85 (.01)	.91 (.01)	.92 (.01)	.74
BIC	.43 (.03)	.56 (.02)	.83 (.01)	.91 (.01)	.92 (.01)	.73
EBIC25	.43 (.03)	.55 (.02)	.82 (.01)	.91 (.01)	.92 (.01)	.72
EBIC50	.43 (.03)	.53 (.02)	.80 (.01)	.90 (.01)	.92 (.01)	.72
CVmin	.42 (.03)	.57 (.02)	.85 (.01)	.91 (.01)	.92 (.01)	.73
CV1se	.42 (.02)	.49 (.02)	.71 (.02)	.84 (.01)	.91 (.01)	.67
split	.05 (.01)	.17 (.02)	.54 (.03)	.86 (.01)	.92 (.01)	.51
bootstrap	.40 (.03)	.58 (.02)	.85 (.01)	.91 (.01)	.92 (.01)	.73
stability	.35 (.03)	.48 (.03)	.79 (.02)	.91 (.01)	.92 (.01)	.69

M = 30%

N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
.36 (.03)	.55 (.03)	.78 (.02)	.86 (.01)	.88 (.01)	.69	Prune
.39 (.03)	.56 (.02)	.78 (.02)	.86 (.01)	.88 (.01)	.69	AIC
.40 (.03)	.55 (.02)	.76 (.02)	.86 (.01)	.88 (.01)	.69	BIC
.41 (.02)	.54 (.02)	.74 (.02)	.86 (.01)	.88 (.01)	.69	EBIC25
.41 (.02)	.53 (.02)	.72 (.02)	.86 (.01)	.88 (.01)	.68	EBIC50
.38 (.03)	.56 (.02)	.78 (.02)	.86 (.01)	.88 (.01)	.69	CVmin
.38 (.03)	.53 (.02)	.70 (.02)	.82 (.01)	.88 (.01)	.66	CV1se
.06 (.02)	.20 (.03)	.56 (.03)	.82 (.02)	.88 (.01)	.50	split
.41 (.03)	.55 (.02)	.77 (.02)	.86 (.01)	.88 (.01)	.69	bootstrap
.35 (.03)	.52 (.02)	.75 (.02)	.86 (.01)	.88 (.01)	.67	stability

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.37 (.03)	.60 (.02)	.81 (.01)	.88 (.01)	.89 (.01)	.71
AIC	.42 (.03)	.61 (.02)	.82 (.01)	.88 (.01)	.89 (.01)	.72
BIC	.41 (.03)	.58 (.02)	.80 (.02)	.88 (.01)	.89 (.01)	.71
EBIC25	.42 (.03)	.57 (.02)	.78 (.02)	.87 (.01)	.89 (.01)	.71
EBIC50	.42 (.03)	.55 (.02)	.76 (.02)	.87 (.01)	.89 (.01)	.70
CVmin	.41 (.02)	.60 (.02)	.82 (.01)	.88 (.01)	.89 (.01)	.72
CV1se	.43 (.03)	.52 (.02)	.69 (.02)	.82 (.02)	.89 (.01)	.67
split	.04 (.01)	.18 (.03)	.56 (.03)	.83 (.02)	.89 (.01)	.50
bootstrap	.42 (.03)	.59 (.02)	.82 (.01)	.88 (.01)	.89 (.01)	.72
stability	.37 (.03)	.51 (.03)	.77 (.02)	.87 (.01)	.89 (.01)	.68

M = 50%

N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
.34 (.03)	.54 (.02)	.80 (.02)	.85 (.02)	.82 (.02)	.67	Prune
.36 (.03)	.56 (.02)	.80 (.02)	.85 (.02)	.82 (.02)	.68	AIC
.36 (.03)	.55 (.02)	.79 (.02)	.85 (.02)	.83 (.02)	.68	BIC
.35 (.03)	.53 (.02)	.77 (.02)	.85 (.02)	.83 (.02)	.67	EBIC25
.35 (.03)	.51 (.02)	.75 (.02)	.85 (.02)	.83 (.02)	.66	EBIC50
.35 (.03)	.55 (.02)	.80 (.02)	.85 (.02)	.82 (.02)	.68	CVmin
.35 (.03)	.53 (.02)	.74 (.02)	.83 (.02)	.83 (.02)	.65	CV1se
.07 (.02)	.21 (.02)	.58 (.02)	.82 (.02)	.83 (.02)	.50	split
.36 (.03)	.56 (.02)	.80 (.02)	.85 (.02)	.83 (.02)	.68	bootstrap
.32 (.03)	.51 (.02)	.79 (.02)	.85 (.02)	.83 (.02)	.66	stability

P = 10

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.33 (.01)	.58 (.01)	.85 (.01)	.94 (.00)	.96 (.00)	.73
AIC	.41 (.01)	.60 (.01)	.86 (.01)	.94 (.00)	.96 (.00)	.75
BIC	.40 (.01)	.56 (.01)	.83 (.01)	.94 (.00)	.96 (.00)	.74
EBIC25	.38 (.01)	.52 (.01)	.79 (.01)	.94 (.00)	.96 (.00)	.72
EBIC50	.37 (.01)	.48 (.01)	.75 (.01)	.94 (.00)	.96 (.00)	.70
CVmin	.40 (.01)	.59 (.01)	.86 (.01)	.94 (.00)	.96 (.00)	.75
CV1se	.38 (.01)	.54 (.01)	.80 (.01)	.93 (.00)	.96 (.00)	.72
split	.00 (.00)	.23 (.01)	.59 (.01)	.90 (.01)	.96 (.00)	.54
bootstrap	.41 (.01)	.61 (.01)	.86 (.01)	.95 (.00)	.96 (.00)	.76
stability	.41 (.01)	.53 (.01)	.83 (.01)	.95 (.00)	.96 (.00)	.74

M = 30%

N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
.34 (.01)	.58 (.01)	.83 (.01)	.90 (.01)	.93 (.01)	.72	Prune
.40 (.01)	.60 (.01)	.83 (.01)	.90 (.01)	.93 (.01)	.73	AIC
.40 (.01)	.56 (.01)	.81 (.01)	.91 (.01)	.94 (.01)	.72	BIC
.39 (.01)	.53 (.01)	.78 (.01)	.90 (.01)	.94 (.01)	.71	EBIC25
.39 (.01)	.50 (.01)	.75 (.01)	.90 (.01)	.94 (.01)	.69	EBIC50
.40 (.01)	.59 (.01)	.83 (.01)	.90 (.01)	.93 (.01)	.73	CVmin
.38 (.01)	.54 (.01)	.79 (.01)	.89 (.01)	.93 (.01)	.71	CV1se
.02 (.01)	.25 (.01)	.62 (.01)	.87 (.01)	.94 (.01)	.54	split
.42 (.01)	.60 (.01)	.83 (.01)	.91 (.01)	.93 (.01)	.74	bootstrap
.42 (.01)	.56 (.01)	.82 (.01)	.91 (.01)	.94 (.01)	.73	stability

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.32 (.01)	.57 (.01)	.85 (.01)	.92 (.01)	.93 (.01)	.72
AIC	.40 (.01)	.60 (.01)	.86 (.01)	.92 (.01)	.93 (.01)	.74
BIC	.41 (.01)	.55 (.01)	.84 (.01)	.93 (.01)	.93 (.01)	.73
EBIC25	.40 (.01)	.52 (.01)	.80 (.01)	.93 (.01)	.93 (.01)	.71
EBIC50	.38 (.01)	.49 (.01)	.75 (.01)	.92 (.01)	.93 (.01)	.69
CVmin	.40 (.01)	.59 (.01)	.86 (.01)	.92 (.01)	.93 (.01)	.74
CV1se	.39 (.01)	.54 (.01)	.81 (.01)	.91 (.01)	.93 (.01)	.72
split	.01 (.00)	.21 (.01)	.60 (.01)	.88 (.01)	.94 (.01)	.53
bootstrap	.40 (.01)	.61 (.01)	.86 (.01)	.93 (.01)	.93 (.01)	.74
stability	.41 (.01)	.55 (.01)	.84 (.01)	.93 (.01)	.93 (.01)	.73

M = 50%

N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
.34 (.01)	.60 (.01)	.82 (.01)	.91 (.01)	.93 (.01)	.72	Prune
.40 (.01)	.61 (.01)	.82 (.01)	.91 (.01)	.93 (.01)	.73	AIC
.41 (.01)	.56 (.01)	.81 (.01)	.91 (.01)	.93 (.01)	.72	BIC
.41 (.01)	.53 (.01)	.79 (.01)	.91 (.01)	.93 (.01)	.71	EBIC25
.41 (.01)	.52 (.01)	.75 (.01)	.90 (.01)	.93 (.01)	.70	EBIC50
.39 (.01)	.60 (.01)	.81 (.01)	.91 (.01)	.92 (.01)	.73	CVmin
.39 (.01)	.56 (.01)	.77 (.01)	.89 (.01)	.90 (.01)	.70	CV1se
.03 (.01)	.26 (.01)	.64 (.01)	.88 (.01)	.93 (.01)	.55	split
.43 (.01)	.62 (.01)	.82 (.01)	.91 (.01)	.93 (.01)	.74	bootstrap
.45 (.01)	.60 (.01)	.81 (.01)	.91 (.01)	.93 (.01)	.74	stability

Table D.2: Mean *pairwise correlations* and their associated standard errors across models and conditions. The values essentially reflect the same information shown in the left-side column of plots in Figure 4.4.

D.3 Correlation (Interactions)

Average Correlations (*SE*) – Interactions

P = 5

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.15 (.03)	.31 (.05)	.86 (.03)	.94 (.02)	.98 (.00)	.65
AIC	.17 (.04)	.36 (.05)	.87 (.03)	.97 (.01)	.98 (.00)	.67
BIC	.12 (.03)	.18 (.04)	.67 (.05)	.97 (.02)	.99 (.00)	.59
EBIC25	.10 (.03)	.12 (.03)	.50 (.05)	.95 (.02)	1.00 (.00)	.53
EBIC50	.06 (.02)	.09 (.03)	.38 (.05)	.90 (.03)	.98 (.01)	.48
CVmin	.17 (.04)	.36 (.05)	.86 (.03)	.96 (.01)	.98 (.00)	.67
CV1se	.10 (.03)	.15 (.04)	.32 (.05)	.62 (.05)	.91 (.03)	.42
split	.00 (.00)	.01 (.01)	.25 (.04)	.92 (.03)	1.00 (.00)	.44
bootstrap	.18 (.04)	.34 (.05)	.86 (.03)	.97 (.01)	.98 (.00)	.67
stability	.01 (.01)	.08 (.03)	.60 (.05)	.97 (.01)	.99 (.00)	.53

M = 30%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.20 (.03)	.41 (.03)	.85 (.02)	.96 (.01)	.99 (.00)	.68	Prune
AIC	.23 (.03)	.42 (.03)	.86 (.02)	.97 (.00)	.99 (.00)	.69	AIC
BIC	.15 (.03)	.25 (.03)	.73 (.03)	.96 (.01)	.99 (.00)	.62	BIC
EBIC25	.14 (.03)	.18 (.03)	.64 (.03)	.93 (.01)	.99 (.00)	.58	EBIC25
EBIC50	.11 (.02)	.10 (.02)	.54 (.04)	.90 (.02)	.99 (.00)	.53	EBIC50
CVmin	.22 (.03)	.40 (.03)	.85 (.02)	.97 (.00)	.99 (.00)	.69	CVmin
CV1se	.15 (.03)	.20 (.03)	.56 (.03)	.79 (.02)	.97 (.01)	.53	CV1se
split	.00 (.00)	.03 (.01)	.46 (.03)	.88 (.01)	.99 (.00)	.47	split
bootstrap	.23 (.03)	.43 (.03)	.87 (.02)	.97 (.00)	.99 (.00)	.70	bootstrap
stability	.03 (.01)	.12 (.02)	.74 (.03)	.96 (.01)	.99 (.00)	.57	stability

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.21 (.03)	.44 (.04)	.80 (.03)	.97 (.01)	.99 (.00)	.68
AIC	.26 (.03)	.47 (.04)	.81 (.03)	.98 (.00)	.99 (.00)	.70
BIC	.17 (.03)	.18 (.03)	.66 (.04)	.95 (.01)	.99 (.00)	.59
EBIC25	.12 (.03)	.14 (.03)	.57 (.04)	.93 (.02)	.99 (.00)	.55
EBIC50	.12 (.03)	.09 (.02)	.49 (.04)	.87 (.03)	.99 (.00)	.51
CVmin	.25 (.03)	.43 (.04)	.81 (.03)	.98 (.01)	.99 (.00)	.69
CV1se	.16 (.03)	.16 (.03)	.51 (.04)	.75 (.04)	.97 (.01)	.51
split	.00 (.00)	.05 (.02)	.35 (.04)	.89 (.02)	.99 (.00)	.46
bootstrap	.26 (.03)	.41 (.04)	.82 (.02)	.97 (.01)	.99 (.00)	.69
stability	.06 (.02)	.14 (.03)	.59 (.04)	.96 (.01)	.99 (.00)	.55

M = 50%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.21 (.03)	.46 (.03)	.83 (.01)	.97 (.00)	.99 (.00)	.69	Prune
AIC	.25 (.03)	.46 (.03)	.83 (.01)	.97 (.00)	.99 (.00)	.70	AIC
BIC	.13 (.02)	.29 (.03)	.73 (.02)	.96 (.01)	.99 (.00)	.62	BIC
EBIC25	.09 (.02)	.20 (.02)	.67 (.02)	.94 (.01)	.99 (.00)	.58	EBIC25
EBIC50	.09 (.02)	.15 (.02)	.57 (.03)	.91 (.01)	.99 (.00)	.54	EBIC50
CVmin	.23 (.03)	.44 (.03)	.83 (.01)	.97 (.00)	.99 (.00)	.69	CVmin
CV1se	.14 (.02)	.25 (.03)	.62 (.02)	.86 (.02)	.97 (.00)	.57	CV1se
split	.00 (.00)	.05 (.02)	.45 (.03)	.87 (.01)	.98 (.00)	.47	split
bootstrap	.23 (.02)	.47 (.03)	.83 (.01)	.97 (.00)	.99 (.00)	.70	bootstrap
stability	.02 (.01)	.16 (.02)	.74 (.02)	.97 (.00)	.99 (.00)	.58	stability

P = 10

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.11 (.02)	.38 (.03)	.78 (.01)	.94 (.00)	.97 (.00)	.64
AIC	.14 (.02)	.46 (.02)	.82 (.01)	.95 (.00)	.98 (.00)	.67
BIC	.10 (.02)	.25 (.02)	.75 (.01)	.96 (.01)	.99 (.00)	.61
EBIC25	.04 (.01)	.10 (.02)	.66 (.02)	.94 (.01)	.99 (.00)	.55
EBIC50	.03 (.01)	.04 (.01)	.50 (.03)	.92 (.01)	.99 (.00)	.50
CVmin	.15 (.02)	.47 (.02)	.82 (.01)	.95 (.00)	.98 (.00)	.67
CV1se	.09 (.02)	.29 (.03)	.73 (.02)	.93 (.01)	.99 (.00)	.61
split	.00 (.00)	.02 (.01)	.44 (.02)	.90 (.01)	.99 (.00)	.47
bootstrap	.22 (.02)	.47 (.02)	.84 (.01)	.96 (.00)	.98 (.00)	.69
stability	.07 (.02)	.18 (.02)	.76 (.02)	.97 (.00)	.99 (.00)	.59

M = 30%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.12 (.02)	.37 (.02)	.73 (.01)	.90 (.01)	.96 (.00)	.61	Prune
AIC	.22 (.02)	.43 (.01)	.74 (.01)	.90 (.01)	.96 (.00)	.65	AIC
BIC	.15 (.02)	.28 (.02)	.69 (.01)	.89 (.01)	.96 (.00)	.59	BIC
EBIC25	.08 (.01)	.16 (.02)	.61 (.01)	.88 (.01)	.96 (.00)	.54	EBIC25
EBIC50	.05 (.01)	.07 (.01)	.51 (.02)	.85 (.01)	.96 (.00)	.49	EBIC50
CVmin	.19 (.02)	.41 (.01)	.73 (.01)	.90 (.01)	.95 (.00)	.64	CVmin
CV1se	.14 (.01)	.32 (.02)	.64 (.01)	.84 (.01)	.95 (.01)	.58	CV1se
split	.00 (.00)	.04 (.01)	.43 (.01)	.76 (.01)	.95 (.00)	.44	split
bootstrap	.24 (.02)	.46 (.01)	.76 (.01)	.91 (.01)	.96 (.00)	.67	bootstrap
stability	.08 (.01)	.27 (.02)	.69 (.01)	.90 (.01)	.96 (.00)	.58	stability

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.14 (.02)	.41 (.02)	.79 (.01)	.92 (.01)	.96 (.00)	.65
AIC	.22 (.02)	.44 (.02)	.81 (.01)	.93 (.01)	.97 (.00)	.67
BIC	.10 (.02)	.25 (.02)	.76 (.01)	.92 (.01)	.97 (.00)	.60
EBIC25	.05 (.01)	.13 (.02)	.65 (.02)	.91 (.01)	.97 (.00)	.54
EBIC50	.03 (.01)	.07 (.01)	.52 (.02)	.89 (.01)	.97 (.00)	.49
CVmin	.19 (.02)	.45 (.02)	.81 (.01)	.92 (.01)	.97 (.00)	.67
CV1se	.11 (.02)	.27 (.02)	.75 (.01)	.89 (.01)	.97 (.01)	.60
split	.00 (.00)	.04 (.01)	.48 (.02)	.84 (.01)	.97 (.00)	.47
bootstrap	.24 (.02)	.47 (.02)	.82 (.01)	.93 (.01)	.97 (.00)	.69
stability	.09 (.02)	.19 (.02)	.76 (.01)	.94 (.01)	.97 (.00)	.59

M = 50%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.15 (.01)	.37 (.01)	.68 (.01)	.86 (.01)	.91 (.01)	.59	Prune
AIC	.22 (.01)	.42 (.01)	.69 (.01)	.86 (.01)	.91 (.01)	.62	AIC
BIC	.13 (.01)	.30 (.01)	.65 (.01)	.86 (.01)	.91 (.01)	.57	BIC
EBIC25	.08 (.01)	.20 (.01)	.60 (.01)	.85 (.01)	.91 (.01)	.53	EBIC25
EBIC50	.05 (.01)	.14 (.01)	.51 (.01)	.82 (.01)	.91 (.01)	.49	EBIC50
CVmin	.19 (.01)	.40 (.01)	.67 (.01)	.85 (.01)	.89 (.01)	.60	CVmin
CV1se	.13 (.01)	.32 (.01)	.58 (.01)	.77 (.01)	.83 (.02)	.53	CV1se
split	.00 (.00)	.05 (.01)	.42 (.01)	.73 (.01)	.90 (.01)	.42	split
bootstrap	.27 (.01)	.45 (.01)	.70 (.01)	.87 (.01)	.91 (.01)	.64	bootstrap
stability	.09 (.01)	.27 (.01)	.66 (.01)	.86 (.01)	.91 (.01)	.56	stability

Table D.3: Mean *interaction correlations* and their associated standard errors across models and conditions. The values essentially reflect the same information shown in the right-side column of plots in Figure 4.4.

D.4 Mean Absolute Error (Pairwise)

Average MAE (*SE*) – Pairwise

P = 5

	M = 10%						M = 30%						
Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.11 (.00)	.09 (.00)	.04 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.05 (.00)	.07	Prune
AIC	.11 (.00)	.09 (.00)	.04 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.05 (.00)	.07	AIC
BIC	.11 (.00)	.09 (.00)	.04 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.04 (.00)	.07	BIC
EBIC25	.11 (.00)	.09 (.00)	.05 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.04 (.00)	.07	EBIC25
EBIC50	.11 (.00)	.09 (.00)	.05 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.10 (.00)	.07 (.00)	.04 (.00)	.04 (.00)	.07	EBIC50
CVmin	.11 (.00)	.09 (.00)	.04 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.04 (.00)	.07	CVmin
CV1se	.11 (.00)	.09 (.00)	.06 (.00)	.04 (.00)	.03 (.00)	.07	.11 (.00)	.10 (.00)	.07 (.00)	.05 (.00)	.04 (.00)	.08	CV1se
split	.10 (.00)	.11 (.00)	.08 (.00)	.04 (.00)	.03 (.00)	.07	.10 (.00)	.11 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.08	split
bootstrap	.11 (.00)	.09 (.00)	.04 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.04 (.00)	.07	bootstrap
stability	.11 (.00)	.09 (.00)	.05 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.04 (.00)	.07	stability

	M = 20%						M = 50%						
Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.12 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.04 (.00)	.07	.12 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.05 (.00)	.08	Prune
AIC	.12 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.04 (.00)	.07	.12 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.05 (.00)	.08	AIC
BIC	.11 (.00)	.09 (.00)	.06 (.00)	.04 (.00)	.04 (.00)	.07	.12 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.05 (.00)	.08	BIC
EBIC25	.11 (.00)	.09 (.00)	.06 (.00)	.04 (.00)	.04 (.00)	.07	.11 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.05 (.00)	.08	EBIC25
EBIC50	.11 (.00)	.09 (.00)	.06 (.00)	.04 (.00)	.03 (.00)	.07	.12 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.05 (.00)	.08	EBIC50
CVmin	.11 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.04 (.00)	.07	.11 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.05 (.00)	.07	CVmin
CV1se	.11 (.00)	.10 (.00)	.07 (.00)	.05 (.00)	.04 (.00)	.07	.11 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.05 (.00)	.08	CV1se
split	.11 (.00)	.11 (.00)	.09 (.00)	.05 (.00)	.03 (.00)	.08	.10 (.00)	.11 (.00)	.08 (.00)	.05 (.00)	.05 (.00)	.08	split
bootstrap	.12 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.04 (.00)	.07	.12 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.05 (.00)	.08	bootstrap
stability	.11 (.00)	.10 (.00)	.06 (.00)	.04 (.00)	.03 (.00)	.07	.12 (.00)	.10 (.00)	.06 (.00)	.05 (.00)	.05 (.00)	.08	stability

P = 10

	M = 10%						M = 30%						
Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.11 (.00)	.09 (.00)	.04 (.00)	.03 (.00)	.02 (.00)	.06	.11 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	Prune
AIC	.11 (.00)	.08 (.00)	.04 (.00)	.03 (.00)	.02 (.00)	.06	.11 (.00)	.08 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	AIC
BIC	.10 (.00)	.08 (.00)	.04 (.00)	.02 (.00)	.02 (.00)	.05	.10 (.00)	.08 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	BIC
EBIC25	.10 (.00)	.09 (.00)	.05 (.00)	.02 (.00)	.02 (.00)	.06	.10 (.00)	.09 (.00)	.05 (.00)	.03 (.00)	.03 (.00)	.06	EBIC25
EBIC50	.10 (.00)	.09 (.00)	.05 (.00)	.02 (.00)	.02 (.00)	.06	.10 (.00)	.09 (.00)	.06 (.00)	.04 (.00)	.03 (.00)	.06	EBIC50
CVmin	.10 (.00)	.08 (.00)	.04 (.00)	.03 (.00)	.02 (.00)	.05	.10 (.00)	.08 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	CVmin
CV1se	.10 (.00)	.09 (.00)	.05 (.00)	.03 (.00)	.02 (.00)	.06	.10 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	CV1se
split	.10 (.00)	.10 (.00)	.08 (.00)	.03 (.00)	.02 (.00)	.06	.10 (.00)	.10 (.00)	.08 (.00)	.04 (.00)	.03 (.00)	.07	split
bootstrap	.11 (.00)	.09 (.00)	.04 (.00)	.02 (.00)	.02 (.00)	.06	.11 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	bootstrap
stability	.11 (.00)	.09 (.00)	.04 (.00)	.02 (.00)	.02 (.00)	.06	.11 (.00)	.09 (.00)	.05 (.00)	.03 (.00)	.03 (.00)	.06	stability

	M = 20%						M = 50%						
Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
Prune	.11 (.00)	.09 (.00)	.04 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	Prune
AIC	.11 (.00)	.08 (.00)	.04 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	AIC
BIC	.10 (.00)	.08 (.00)	.04 (.00)	.03 (.00)	.03 (.00)	.06	.10 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	BIC
EBIC25	.10 (.00)	.09 (.00)	.05 (.00)	.03 (.00)	.03 (.00)	.06	.10 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	EBIC25
EBIC50	.10 (.00)	.09 (.00)	.05 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.09 (.00)	.06 (.00)	.04 (.00)	.03 (.00)	.07	EBIC50
CVmin	.10 (.00)	.08 (.00)	.04 (.00)	.03 (.00)	.03 (.00)	.06	.10 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	CVmin
CV1se	.10 (.00)	.08 (.00)	.05 (.00)	.03 (.00)	.03 (.00)	.06	.10 (.00)	.09 (.00)	.06 (.00)	.04 (.00)	.04 (.00)	.06	CV1se
split	.10 (.00)	.10 (.00)	.08 (.00)	.03 (.00)	.03 (.00)	.07	.10 (.00)	.10 (.00)	.08 (.00)	.04 (.00)	.03 (.00)	.07	split
bootstrap	.11 (.00)	.08 (.00)	.04 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	bootstrap
stability	.10 (.00)	.09 (.00)	.04 (.00)	.03 (.00)	.03 (.00)	.06	.11 (.00)	.09 (.00)	.05 (.00)	.04 (.00)	.03 (.00)	.06	stability

Table D.4: Mean *pairwise MAEs* and their associated standard errors across models and conditions. The values essentially reflect the same information shown in the left-side column of plots in Figure 4.5.

D.5 Mean Absolute Error (Interactions)

Average MAE (*SE*) – Interactions

P = 5

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.03 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.00 (.00)	.01
AIC	.03 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.00 (.00)	.01
BIC	.02 (.00)	.02 (.00)	.01 (.00)	.00 (.00)	.00 (.00)	.01
EBIC25	.02 (.00)	.02 (.00)	.01 (.00)	.00 (.00)	.00 (.00)	.01
EBIC50	.02 (.00)	.02 (.00)	.01 (.00)	.00 (.00)	.00 (.00)	.01
CVmin	.03 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.00 (.00)	.01
CV1se	.02 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.00 (.00)	.01
split	.02 (.00)	.02 (.00)	.02 (.00)	.00 (.00)	.00 (.00)	.01
bootstrap	.03 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.00 (.00)	.01
stability	.02 (.00)	.02 (.00)	.01 (.00)	.00 (.00)	.00 (.00)	.01

M = 30%

N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
.07 (.00)	.05 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.03	Prune
.07 (.00)	.05 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.03	AIC
.06 (.00)	.05 (.00)	.03 (.00)	.01 (.00)	.01 (.00)	.03	BIC
.06 (.00)	.06 (.00)	.03 (.00)	.01 (.00)	.01 (.00)	.03	EBIC25
.06 (.00)	.06 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.04	EBIC50
.06 (.00)	.05 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.03	CVmin
.06 (.00)	.06 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.04	CV1se
.06 (.00)	.06 (.00)	.05 (.00)	.02 (.00)	.01 (.00)	.04	split
.06 (.00)	.05 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.03	bootstrap
.06 (.00)	.06 (.00)	.03 (.00)	.01 (.00)	.01 (.00)	.03	stability

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.05 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.02
AIC	.04 (.00)	.03 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.02
BIC	.04 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.00 (.00)	.02
EBIC25	.04 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.00 (.00)	.02
EBIC50	.04 (.00)	.04 (.00)	.03 (.00)	.01 (.00)	.00 (.00)	.02
CVmin	.04 (.00)	.03 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.02
CV1se	.04 (.00)	.04 (.00)	.03 (.00)	.02 (.00)	.01 (.00)	.03
split	.04 (.00)	.04 (.00)	.03 (.00)	.01 (.00)	.00 (.00)	.03
bootstrap	.05 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.02
stability	.04 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.00 (.00)	.02

M = 50%

N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
.10 (.00)	.09 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.05	Prune
.10 (.00)	.08 (.00)	.04 (.00)	.02 (.00)	.02 (.00)	.05	AIC
.10 (.00)	.09 (.00)	.05 (.00)	.02 (.00)	.01 (.00)	.06	BIC
.10 (.00)	.09 (.00)	.06 (.00)	.02 (.00)	.01 (.00)	.06	EBIC25
.10 (.00)	.10 (.00)	.07 (.00)	.03 (.00)	.01 (.00)	.06	EBIC50
.10 (.00)	.09 (.00)	.04 (.00)	.02 (.00)	.02 (.00)	.05	CVmin
.10 (.00)	.09 (.00)	.06 (.00)	.03 (.00)	.02 (.00)	.06	CV1se
.10 (.00)	.10 (.00)	.08 (.00)	.03 (.00)	.01 (.00)	.07	split
.10 (.00)	.08 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.05	bootstrap
.10 (.00)	.10 (.00)	.05 (.00)	.02 (.00)	.01 (.00)	.06	stability

P = 10

M = 10%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.03 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.00 (.00)	.02
AIC	.03 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.00 (.00)	.01
BIC	.02 (.00)	.02 (.00)	.01 (.00)	.00 (.00)	.00 (.00)	.01
EBIC25	.02 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.00 (.00)	.01
EBIC50	.02 (.00)	.02 (.00)	.02 (.00)	.01 (.00)	.00 (.00)	.01
CVmin	.02 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.00 (.00)	.01
CV1se	.02 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.00 (.00)	.01
split	.02 (.00)	.02 (.00)	.02 (.00)	.01 (.00)	.00 (.00)	.01
bootstrap	.03 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.00 (.00)	.02
stability	.02 (.00)	.02 (.00)	.01 (.00)	.00 (.00)	.00 (.00)	.01

M = 30%

N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
.07 (.00)	.06 (.00)	.03 (.00)	.02 (.00)	.02 (.00)	.04	Prune
.07 (.00)	.06 (.00)	.03 (.00)	.02 (.00)	.02 (.00)	.04	AIC
.06 (.00)	.06 (.00)	.04 (.00)	.02 (.00)	.02 (.00)	.04	BIC
.06 (.00)	.06 (.00)	.04 (.00)	.02 (.00)	.02 (.00)	.04	EBIC25
.06 (.00)	.06 (.00)	.05 (.00)	.03 (.00)	.02 (.00)	.04	EBIC50
.06 (.00)	.06 (.00)	.03 (.00)	.02 (.00)	.02 (.00)	.04	CVmin
.06 (.00)	.06 (.00)	.04 (.00)	.03 (.00)	.02 (.00)	.04	CV1se
.06 (.00)	.06 (.00)	.05 (.00)	.03 (.00)	.02 (.00)	.05	split
.07 (.00)	.06 (.00)	.03 (.00)	.02 (.00)	.02 (.00)	.04	bootstrap
.06 (.00)	.06 (.00)	.04 (.00)	.02 (.00)	.02 (.00)	.04	stability

M = 20%

Model	N = 50	N = 100	N = 250	N = 500	N = 1000	M_N
Prune	.05 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.03
AIC	.05 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.02
BIC	.04 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.02
EBIC25	.04 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.03
EBIC50	.04 (.00)	.04 (.00)	.03 (.00)	.01 (.00)	.01 (.00)	.03
CVmin	.04 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.02
CV1se	.04 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.02
split	.04 (.00)	.04 (.00)	.03 (.00)	.02 (.00)	.01 (.00)	.03
bootstrap	.05 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.02
stability	.04 (.00)	.04 (.00)	.02 (.00)	.01 (.00)	.01 (.00)	.02

M = 50%

N = 50	N = 100	N = 250	N = 500	N = 1000	M_N	Model
.11 (.00)	.10 (.00)	.06 (.00)	.04 (.00)	.04 (.00)	.07	Prune
.10 (.00)	.09 (.00)	.06 (.00)	.04 (.00)	.04 (.00)	.07	AIC
.10 (.00)	.09 (.00)	.07 (.00)	.04 (.00)	.04 (.00)	.07	BIC
.10 (.00)	.10 (.00)	.07 (.00)	.05 (.00)	.04 (.00)	.07	EBIC25
.10 (.00)	.10 (.00)	.08 (.00)	.05 (.00)	.04 (.00)	.07	EBIC50
.10 (.00)	.09 (.00)	.06 (.00)	.05 (.00)	.04 (.00)	.07	CVmin
.10 (.00)	.09 (.00)	.07 (.00)	.05 (.00)	.05 (.00)	.07	CV1se
.10 (.00)	.10 (.00)	.09 (.00)	.06 (.00)	.04 (.00)	.08	split
.11 (.00)	.09 (.00)	.06 (.00)	.04 (.00)	.04 (.00)	.07	bootstrap
.10 (.00)	.10 (.00)	.06 (.00)	.04 (.00)	.04 (.00)	.07	stability

Table D.5: Mean *interaction MAEs* and their associated standard errors across models and conditions. The values essentially reflect the same information shown in the right-side column of plots in Figure 4.5.

Appendix E

Simulation Study—Ordinalized Variables

The following pages contain the same plots as shown for the simulation study, except that in these cases we see the results for the ordinalized versions of the datasets. That is, I transformed all variables in each case to become ordinal variables with four levels. Additionally, I wanted to add some variability to the shapes of the distributions. Because the data were originally generated from univariate normal distributions, the example is rather idealized since researchers may often be working with non-normal variables. Moreover, rank-ordered scales are overwhelmingly used in clinical psychological research, where measures are commonly taken with 0–3 or 1–5 point scales. Thus, the objective was to try and mimic a scenario that was potentially more realistic.

To do this, for each variable in each dataset (including the moderators) I selected 3 random cut points from $\mathcal{N}(0,1)$ and grouped values accordingly. And although the cut points were drawn from normal distributions, they ubiquitously induced non-normality to the distributions of the variables, as well as the multivariate distributions of the datasets (as shown in Table 4.1. This provides a much stronger test of the models, and is perhaps a much more realistic scenario than the original data.

For the results, we see more or less the same patterns as observed in the continuous-variable analyses, however across the board there are reductions in both sensitivity and accuracy, increases in false positives, and a general increase in the variability of the estimates. Specificity remains equally as high as in the original study. Despite an overall reduction in performance, which is to be expected, we see that it remains satisfactory for many models at larger sample sizes ($N \geq 500$). The general conclusion that can be drawn here is that larger samples will typically be necessary to achieve good performance when items are measured on ordinal scales and the data are skewed.

E.1 Sensitivity—Ordinal (Interactions)

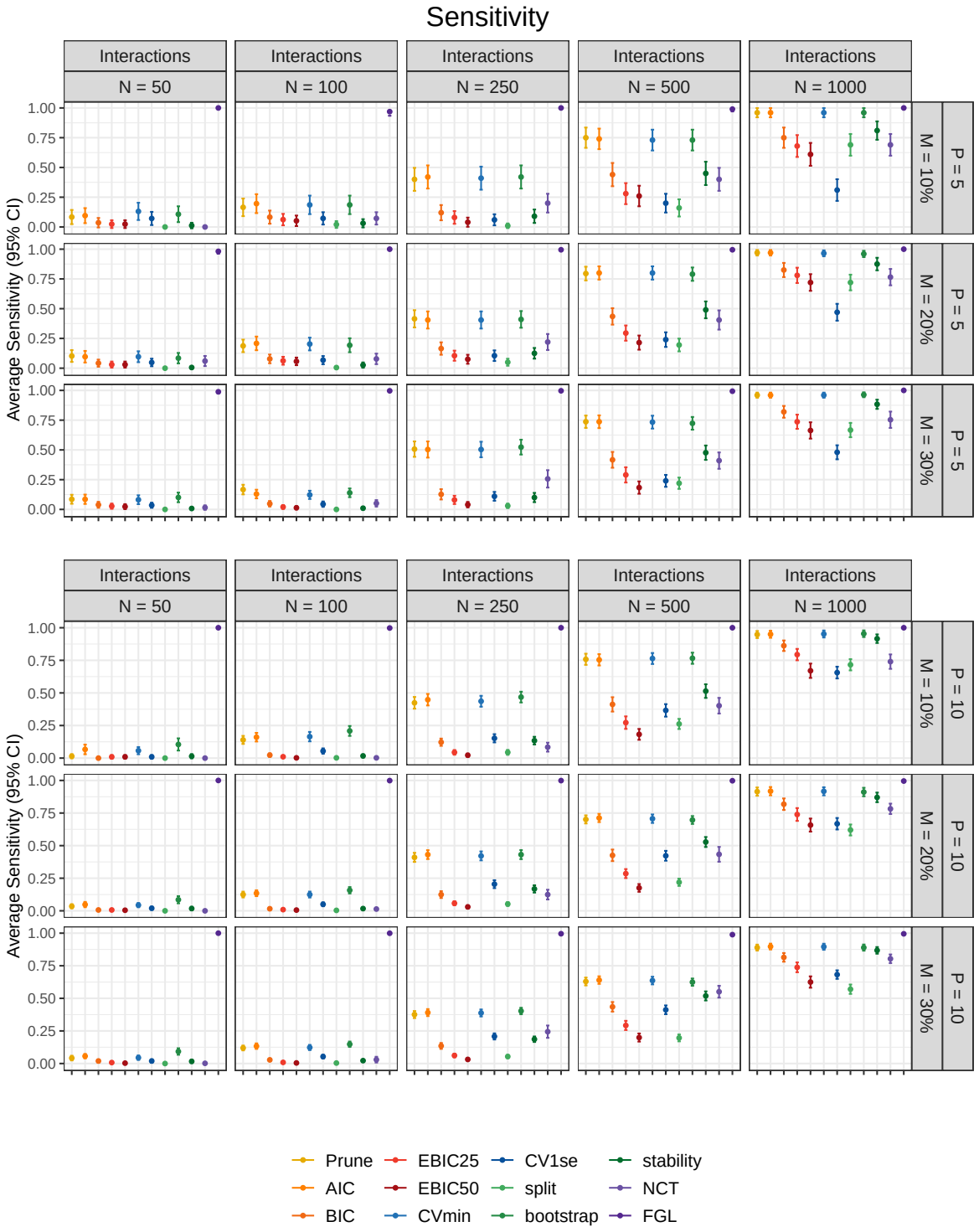

Figure E.1: Average sensitivity across models and conditions for the ordinalized version of the simulation study. Each plotted point is based on 100 simulated datasets

E.2 False-Discovery Rate—Ordinal (Interactions)

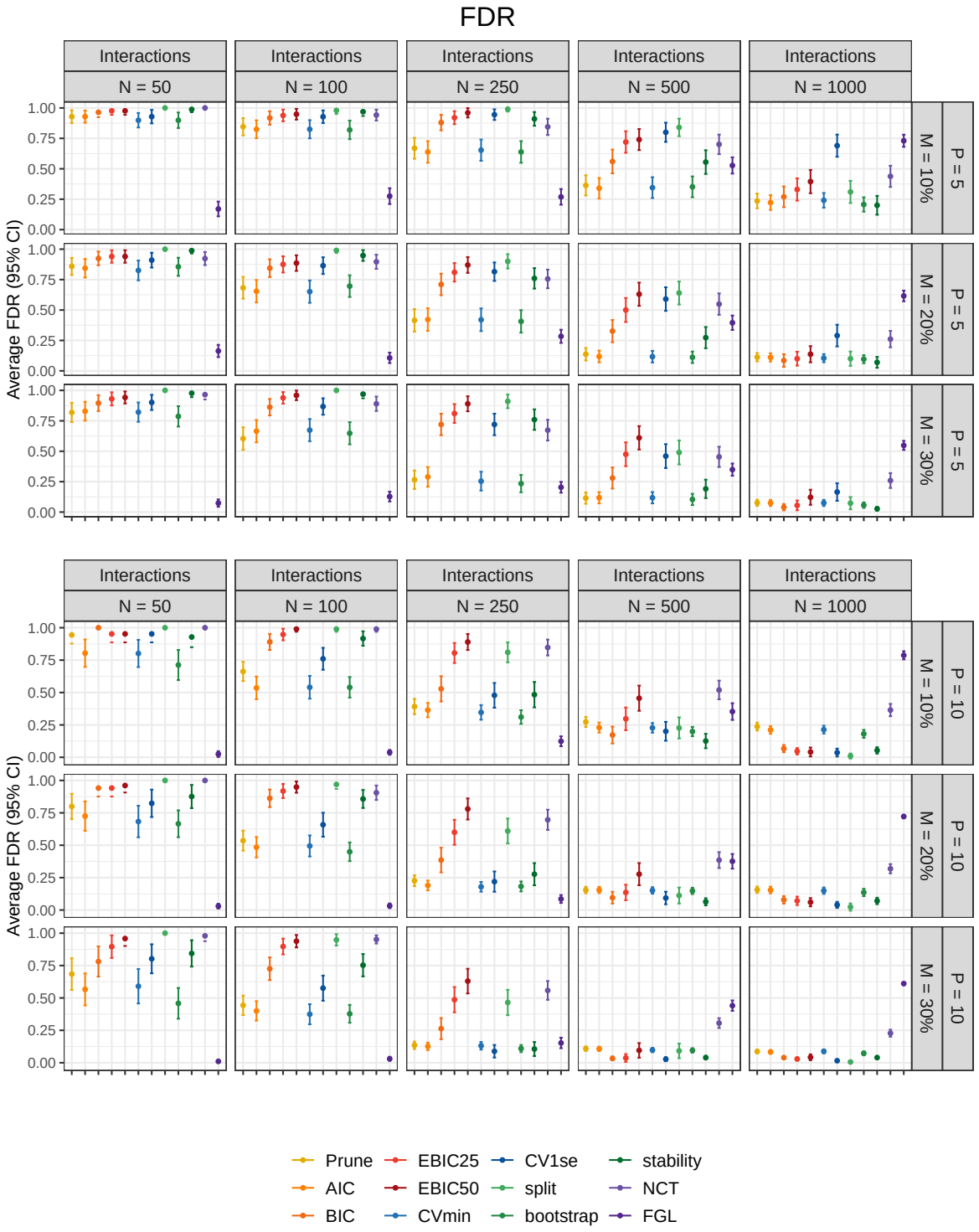

Figure E.2: Average false-discovery rate across models and conditions for the ordinalized version of the simulation study. Each plotted point is based on 100 simulated datasets

E.3 Matthew's Correlation Coefficient—Ordinal (Interactions)

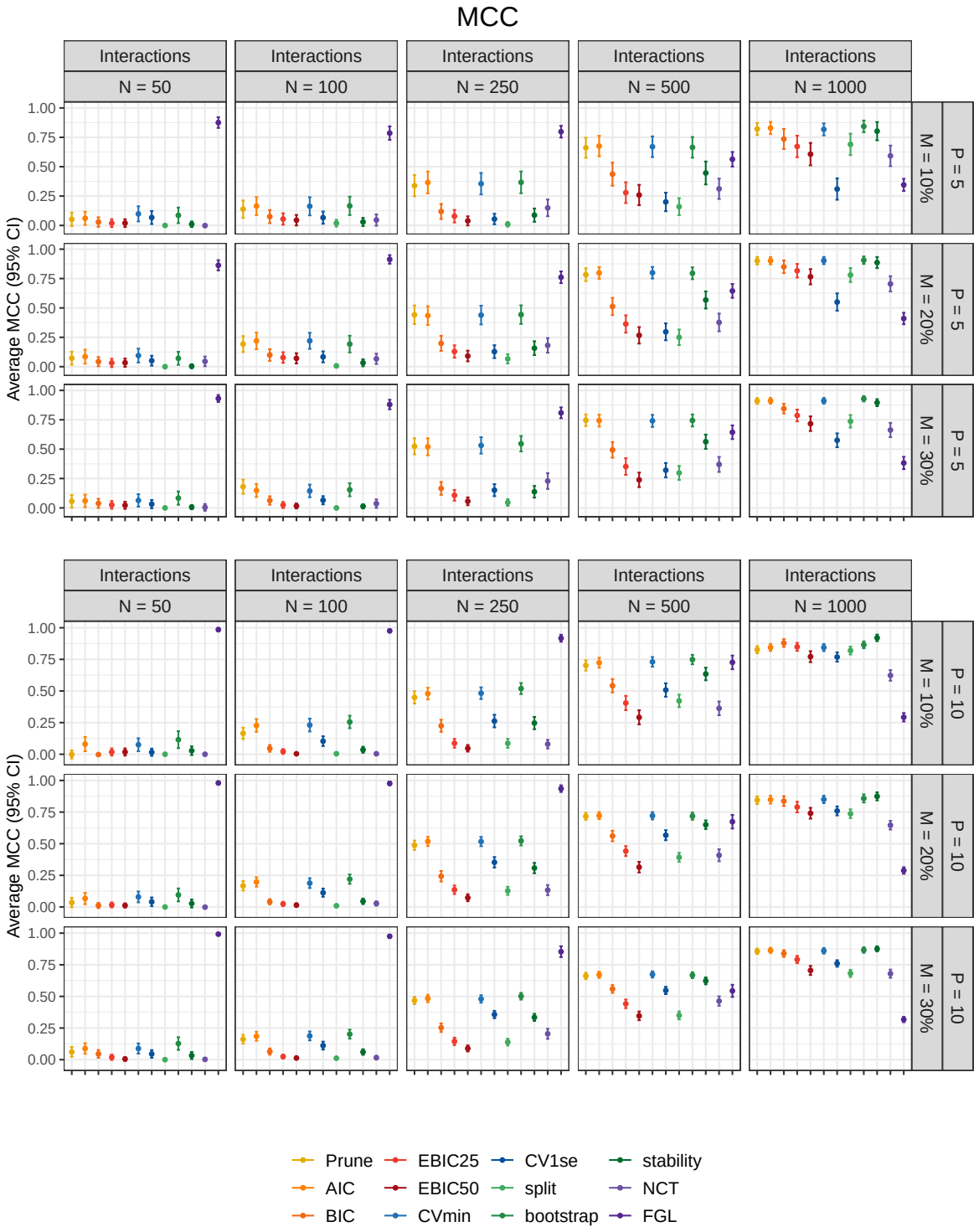

Figure E.3: Average MCC across models and conditions for the ordinalized version of the simulation study. Each plotted point is based on 100 simulated datasets

E.4 Specificity—Ordinal (Interactions)

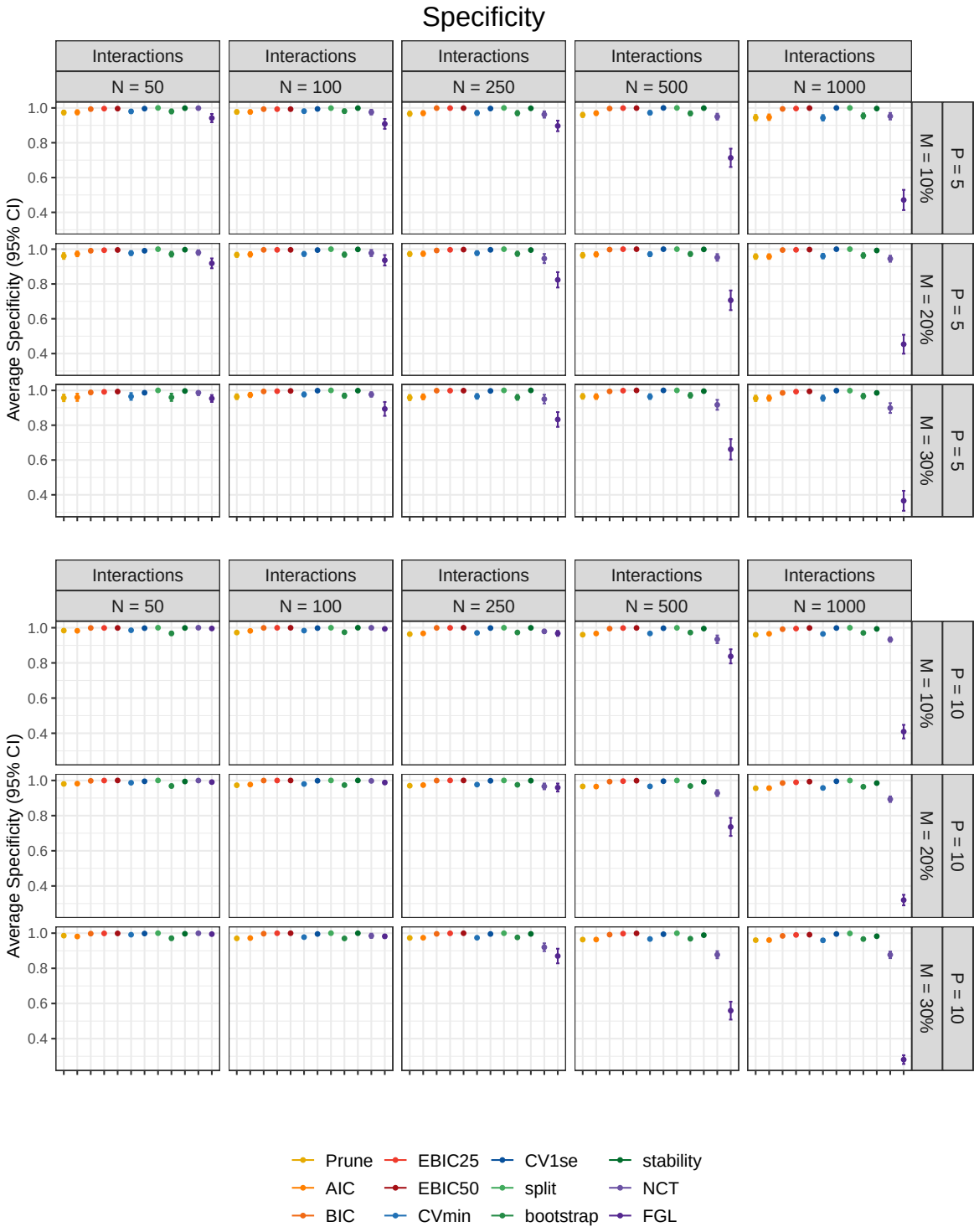

Figure E.4: Average specificity across models and conditions for the ordinalized version of the simulation study. Each plotted point is based on 100 simulated datasets

E.5 Correlation—Ordinal

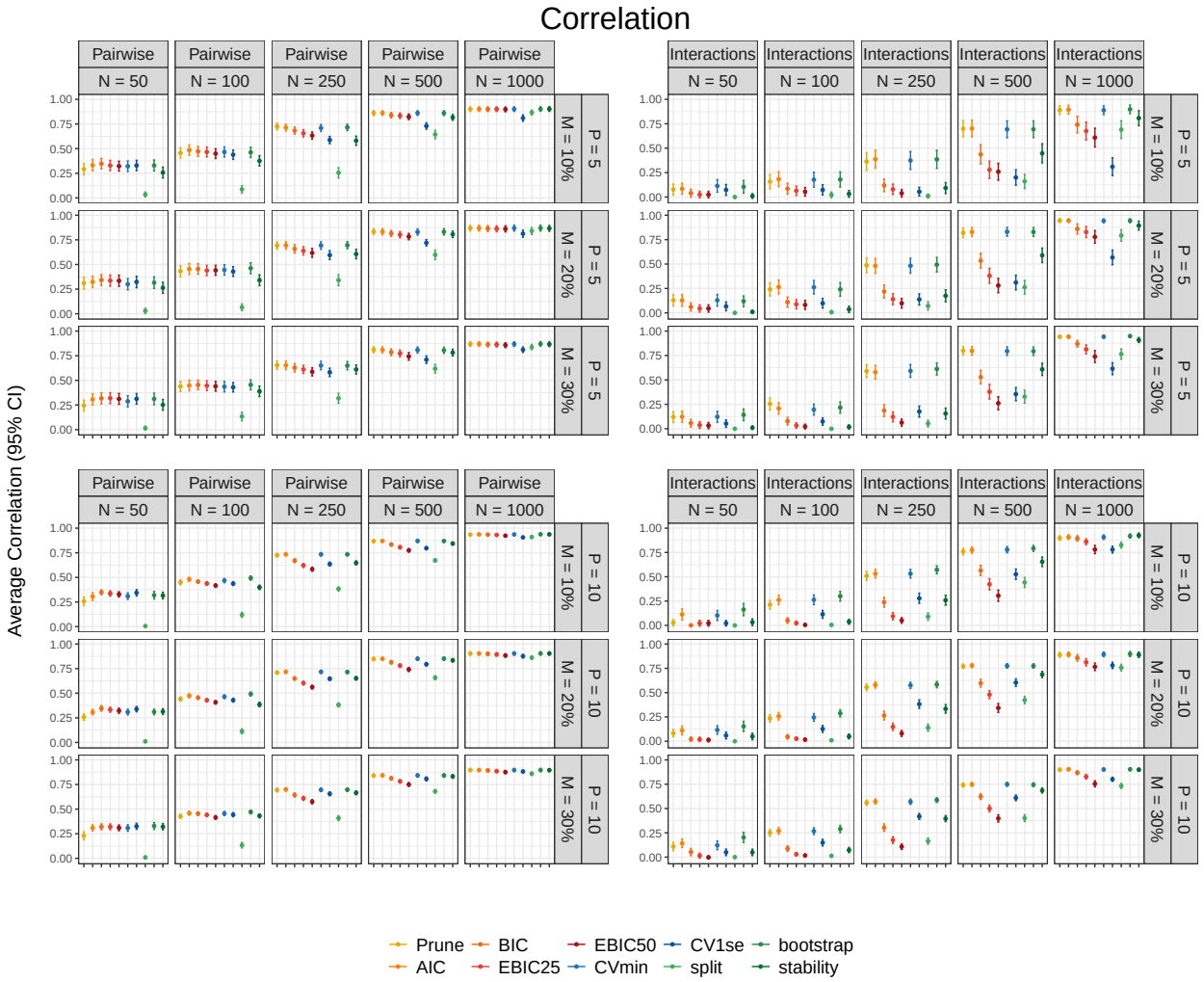

Figure E.5: Average correlations across MNMs and conditions for the ordinalized version of the simulation study. Each plotted point is based on 100 simulated datasets.

E.6 Mean Absolute Error—Ordinal

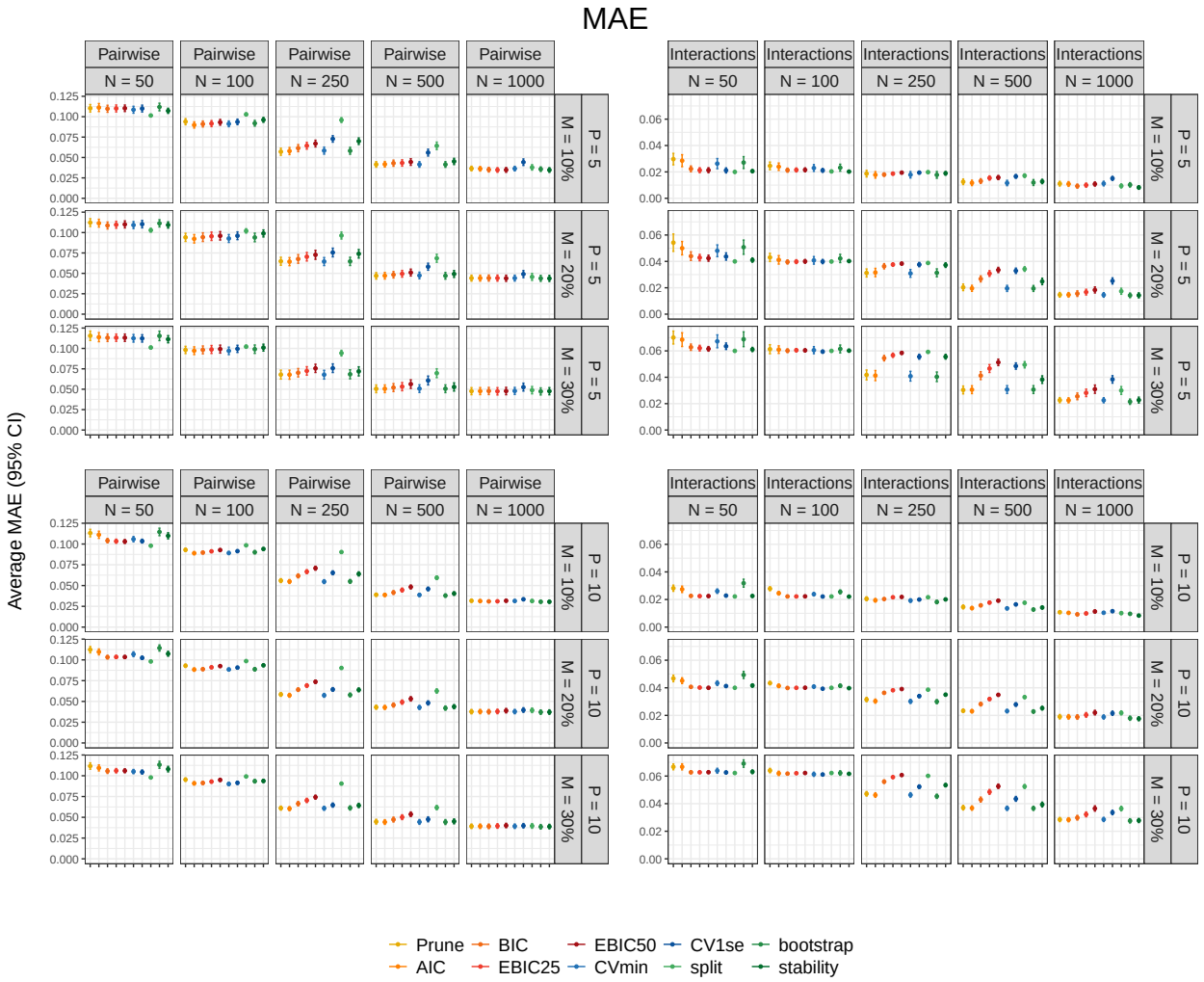

Figure E.6: Average MAE across MNMs and conditions for the ordinalized version of the simulation study. Each plotted point is based on 100 simulated datasets.

Appendix F

Temporal MNM Empirical Example—Correlation Stability

F.1 Centrality Estimates from Full Sample

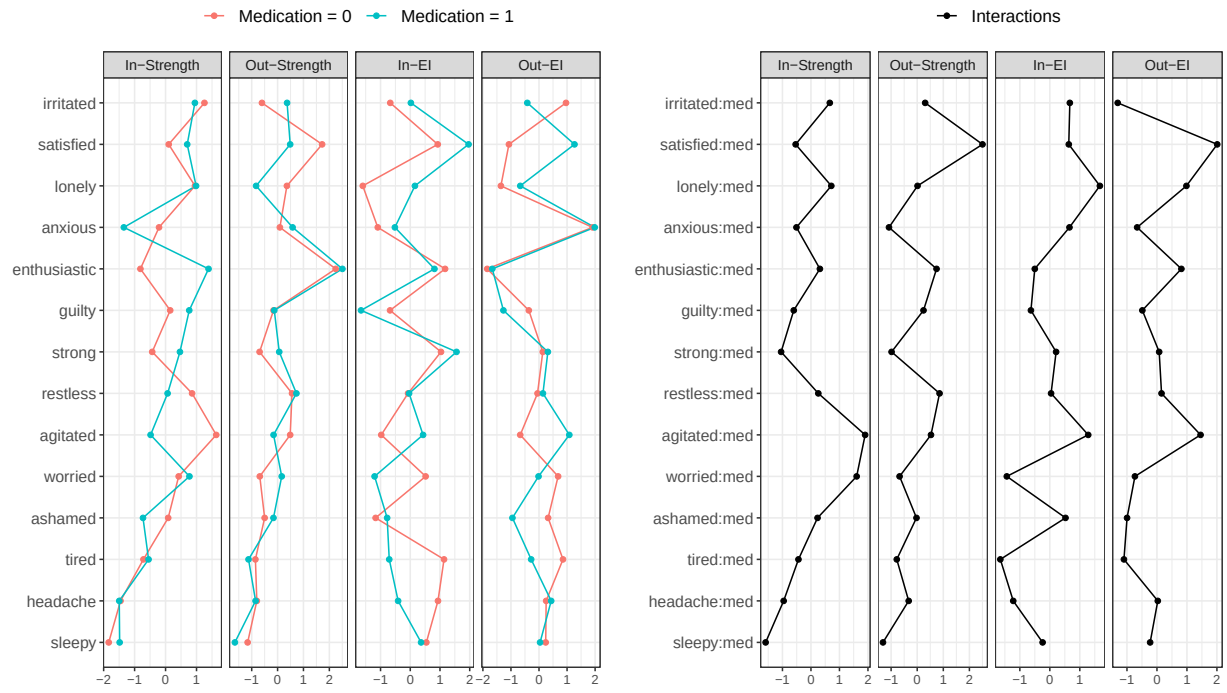

Figure F.1: Centrality estimates based on the conditional temporal networks (left) and the interaction terms (right) using the full sample. The raw centrality values have been standardized to represent z-scores in the plot.

F.2 Correlation Stability—Edge Weights

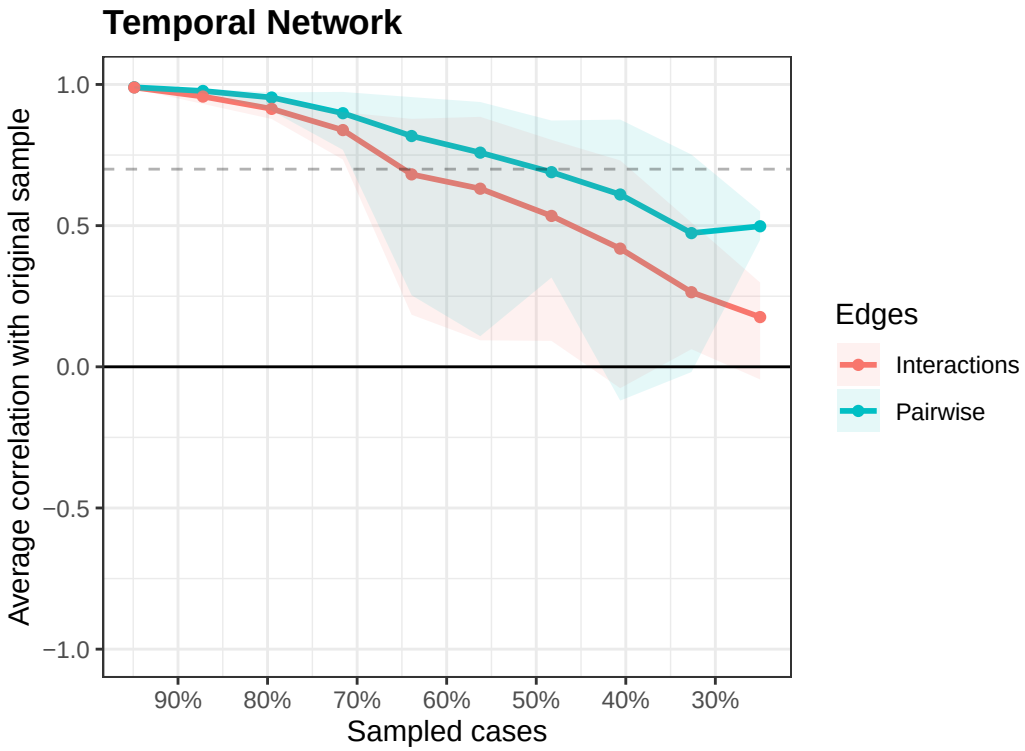

Figure F.2: Case-dropping bootstrap results with $B = 1000$ for edge weights in the temporal network (specifically, when 'medication' = 0).

F.3 Correlation Stability—Centrality Measures

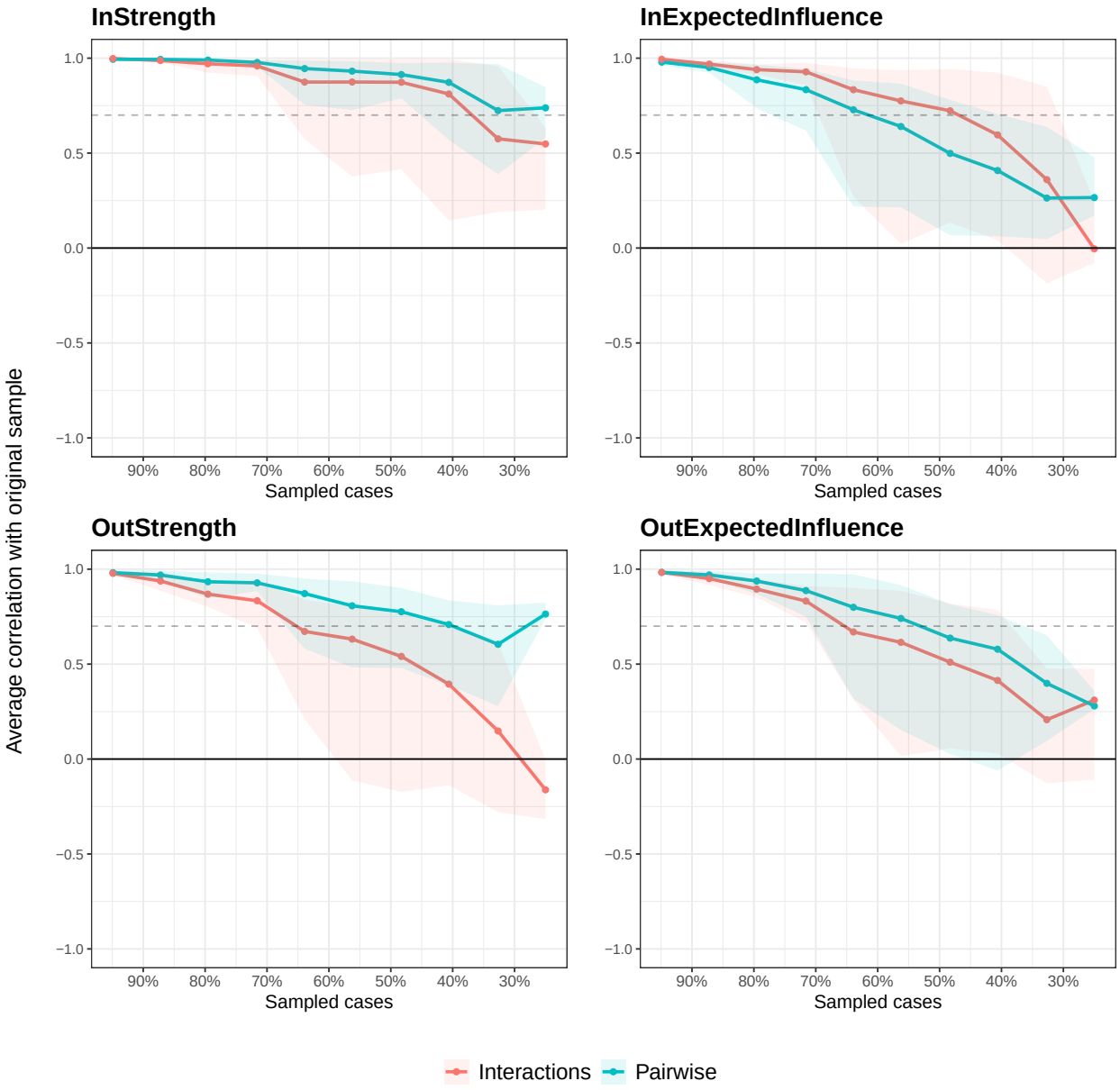

Figure F.3: Case-dropping bootstrap results with $B = 1000$ for ingoing and outgoing variants of strength centrality and expected influence for nodes and interaction terms in the temporal network (specifically, when 'medication' = 0).