The Theoretical and Statistical Ising Model: A Practical Guide in *R*

Adam Finnemann [1,2,*], Denny Borsboom [1], Sacha Epskamp [1,2] and Han L. J. van der Maas [1]

1 Department of Psychology, University of Amsterdam, 1105 Amsterdam, The Netherlands;
   D.borsboom@uva.nl (D.B.); mail@sachaepskamp.com (S.E.); H.L.J.vanderMaas@uva.nl (H.L.J.v.d.M.)
2 Center for Urban Mental Health, University of Amsterdam, 1105 Amsterdam, The Netherlands
* Correspondence: adam.finnemann@gmail.com

Abstract: The "Ising model" refers to both the *statistical* and the *theoretical* use of the same equation. In this article, we introduce both uses and contrast their differences. We accompany the conceptual introduction with a survey of Ising-related software packages in *R*. Since the model's different uses are best understood through simulations, we make this process easily accessible with fully reproducible examples. Using simulations, we show how the theoretical Ising model captures *local-alignment* dynamics. Subsequently, we present it statistically as a likelihood function for estimating empirical network models from binary data. In this process, we give recommendations on when to use traditional frequentist estimators as well as novel Bayesian options.

Keywords: ising model; alignment; dynamics; binary; networks; computational; estimation; Bayesian; frequentist

1. Introduction

The Ising model (named after Ernest Ising. Ironically, Ising concluded that the model was unfit for physics and left science. It was to his surprise that he had become famous through the work of other scientists on the same model [1]) has attracted widespread scientific attention for several decades [1–7]. Early fascination arose because it was not suspected that such a simple model could produce interesting behaviours. Once its surprisingly complex dynamics were established and used to model magnetic properties, it became one of the most taught and studied models within physics [4,6].

Since its origin in physics, it has spread to several other sciences [5,6,8–12]. Its applications roughly fall into one of two categories. First, it is used as a theoretical model of empirical phenomena. Second, it is used as a data analytic model that provides a statistical likelihood function for dependencies between binary variables. This two-fold use can be explained by its characteristic of being a maximum entropy distribution [12]. From thermodynamics, we know that physical processes maximise entropy [13], which motivates its modeling applications. Additionally, information theory motivates the maximization of entropy given incomplete information [14]. Thus, there is a correspondence between different interpretations of maximum entropy and the Ising model's applications.

In this paper we focus on its recent use in the expanding literature on network modeling and network psychometrics [15,16]. Here, we see its two-fold use both as a theoretical model of political beliefs, attitudes, and depression [17–19], and as a psychometric data analytic tool [7,20–23].

Both uses have been accompanied by software developments that make these techniques accessible to researchers. However, as of yet there exists no systematic treatise that illustrates the dual use of the Ising model in psychological applications. The current paper aims to fill this gap; in the spirit of "learning by simulating", we show readers how to computationally study both the theoretical and statistical properties of the model. We

focus on the freely accessible programming language *R*, where we survey relevant software packages and illustrate their use [24].

The paper proceeds as follows: In the first part we introduce the Ising model as a theoretical model of *alignment*. Afterwards we explicate the theoretical model by presenting an overview of software packages and employ these to simulate the *dynamics of alignment*. We finish the section on its theoretical use by discussing mean-field approximations, encoding and practical application.The fourth part of the paper surveys the statistical use of the Ising model by discussing software related to network psychometrics for binary data. Throughout the sections, text boxes exemplify how the software packages are used. We conclude the paper by discussing the practical differences between the statistical and the theoretical Ising model.

2. The Theoretical Ising Model

In this section, we introduce the Ising model as a theoretical model. Conceptually, we view it as a network governed by the process of *local alignment*. In the next part of the section, we introduce software relevant to deriving *local alignment dynamics* and show how we can summarise the dynamics as a cusp catastrophe model. Lastly, we discuss variable encodings and illustrate its extensive applications in climate and opinion research.

2.1. A Conceptual Introduction

We begin with Figure 1, which shows a general network consisting of five nodes (circle) and six edges (lines connecting the circles). It is general in the sense that nodes and edges can represent any suitable features. As our example, we follow [19], who interprets nodes as attitude elements (such as beliefs, feelings, and behaviours) and edges as their pairwise tendency to align. Following this, we can understand, e.g., a vegetarian attitude as the alignment of meat consumption with beliefs and feelings towards the meat producing industry, climate, health, etc. [19].

If two nodes are connected by an edge, we say they are *neighbours*. Furthermore, each node *i* is associated with a variable X_i. More importantly, the Ising model assumes that X_i is *dichotomous*. Most commonly, X_i is encoded as 1 and −1. For instance, we can code positive-attitude elements with 1 and a negative value with −1. We refer to the value of X_i with x_i. Alternative encodings and their implications are discussed later in this section.
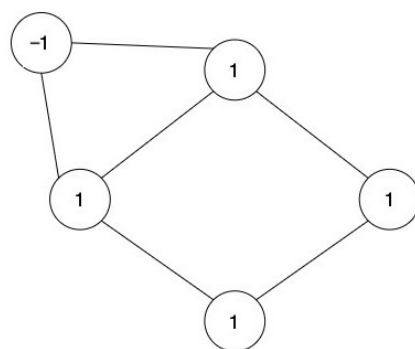

Figure 1. A simple network with five nodes and six edges. Its current configuration is $\boldsymbol{x} = [-1, 1, 1, 1, 1]$.

In Figure 1, the current *configuration* is $\boldsymbol{x} = [-1, 1, 1, 1, 1]$. This is 1 out of $2^5 = 32$ possible configurations. In general, there are 2^n possible configurations for *n* nodes. The Ising model specifies the probability distribution over all network configurations $P(\boldsymbol{X} = \boldsymbol{x})$. Where $\boldsymbol{X}$ is a vector representing all possible configurations $\boldsymbol{X} = [X_1 \ \ldots \ X_p]$.

2.2. A Model of Alignment

The central assumption of the theoretical Ising model is: the *alignment* of the current configuration determines how likely it is to occur. Alignment refers to cases where

neighbours have the same values (i.e., 1, 1 or -1, -1), whereas divergence happens when neighbours have different values (i.e., -1, 1 or 1, -1). In our simple example, we observe four aligned cases and two that are non-aligned.

The central term of the Ising model is the potential function H, which models alignment as follows (The negative signs in H is the standard notation for the Ising model. It comes from a thermodynamic interpretation where states with the lowest energy/H are the most likely. Note that it is cancelled out by the additional minus sign we see in the exponent of Equation (2)):

$$H(\boldsymbol{x}) = -\sum_i \alpha x_i - \sum_{<i,j>} \omega_{i,j} x_i x_j \tag{1}$$

where ω encodes the network structure. If $\omega_{i,j} = 0$, then i and j are unconnected. If $\omega_{i,j} \neq 0$, they are connected with the value of $\omega_{i,j}$, reflecting the edge strength. Hence, $\sum_{<i,j>} \omega_{i,j}$ means that we sum up all neighbours (six pairs in our case). Finally, α is the *external field* parameter, which can also be interpreted as an intercept. It gives nodes a directional preference regardless of their neighbours. We see that aligned neighbours and alignment with the external fields make positive contributions to H (assuming all ω parameters are positive). From the full Ising equation, we see that H drives up the overall probability of the configuration:

$$P(\boldsymbol{X} = \boldsymbol{x}) = \frac{e^{-\beta H(\boldsymbol{x})}}{Z} \tag{2}$$

The numerator in Equation (2), Z, sums up all configurations, making sure that the probabilities sum up to 1:

$$Z = \sum_{\boldsymbol{X}} e^{-\beta H(\boldsymbol{x})}$$

This term is called the normalizing constant or the partition function. With more than 20 nodes, Z becomes intractable to calculate due to the 2^n computations. Many of the later mathematical tricks that we encounter—sampling methods, expected values, and pseudo-likelihoods—are essentially ways of avoiding the computation of Z.

Equation (2) also contains the β parameter, which we refer to as the *alignment weight* because it weighs the importance of H on $P(\boldsymbol{X} = \boldsymbol{x})$. We show this in the next section. The β parameter is often referred to as the inverse temperature due to its original ferromagnetic interpretation [1], and sometimes as the density parameter [25].

The Ising model enables us to understand how alignment processes and external influences interact. To do so, we must study which network configurations arise when the external field α and alignment weight β are varied. We encourage readers to construct their own simulations and insights. In the next section, we will make this process easy and available. To assist us, Table 1 gives an overview of relevant functions in R.

Table 1. Software packages in *R* relevant in the simulation of Ising Dynamics. We relate functions to their packages using the ':::' notation from *R*. To access a function, we first install its package with *install.package("Example")* then load the package with *library("Example")*.

Package::Function()	Description
`IsingSampler::Isingsampler()`	Flexible Ising state sampler
`bayess:isinghm()`	Metropolis–Hastings Sampler
`igraph::make_lattice()`	N dimensional lattice structures
`igraph::sample_small_world()`	Watts–Strogatz model
`parSim::parSim()`	Easy simulations and multi-core
`ggplot2::ggplot()`	Visualisation
`set.seed()`	Reproduces random numbers

3. Simulating Ising Dynamics

In this section we simulate dynamical processes using an Ising model. This requires two elements: a network structure and the dynamics that determine the progression of the system over time. The network structure is encoded as an adjacency matrix in ω from Equation (2).

3.1. Network Structures

Choosing a network structure is an empirical question that depends on what is being modelled. Ref. [26] discusses commonly found network structures in physics, biology, and social science. Here, we will focus on two simple but general network structures: *lattices* and the *Watts–Strogatz* model [27]. Lattices are *n*-dimensional grid structures (see Figure 2 for a 2*D* lattice) often used in physics to model atoms. Social and biological networks have been argued to satisfy small-world properties, having both high local clustering and low average distance between nodes. We are not aware of any characteristic structure for psychological networks, although small-world networks have been observed [28,29]. Due to its popularity across disciplines we use it for our simulations. We can obtain small-world networks starting with a ring lattice (see Figure 2) followed by randomly rewiring edges. If this process is continued, we obtain a random graph. Between the ring lattice and random graph, we find the small-world network. It has the local clustering of the ring lattice and low average distances from the random graph. Obtaining small-world properties as a mix of a ring lattice and random graph is referred to as the Watts–Strogatz model. In Text Box 1, we show how lattices and small-world networks are easily generated in *R*. We also show how to weigh the edges, which is needed to model connections of varying strengths.

Box 1. How to construct common network structures in *R* [30].

Using *Igraph*, we can create lattices and small-world networks[a] [30].

1. *make_lattice()* allows us to create lattice structures of any size and dimension;
2. With *sample_small_world()*, we generate various networks by varying the rewiring probability *p*. When $p = 1$, we obtain a random network and with $p = 0$, we obtain a ring lattice. For in-between values of *p*, we obtain small-world networks that inherit the local clustering of ring lattices and the short average distance of random networks. For illustrations, see Figure 2;
3. **Running example:** We create a small-world network, which we use for simulations later in the paper. We create differences in edge weights with the *apply()* function[b].

```
library("igraph")
library("magrittr")
set.seed(4343)
n_nodes <- 40
sw_network <- igraph::sample_smallworld(dim = 1,
size = n_nodes,
nei = 2,
p = 0.3) %>%
as_adjacency_matrix() %>% as.matrix()  %>%
apply(c(1,2),function(x) x * #make weighted network
rnorm(1, mean = 1, sd = 0.3))

sw_network[lower.tri(sw_network)] <- #Forcing symmetrical around
t(sw_network)[lower.tri(sw_network)] #diagonal
```

---

[a] Annotated and fully reproducible code is found in the supplementary material at https://osf.io/4z9u2/. This includes all visualisation codes.

[b] We use the pipe operator "%>%" from the *magrittr* package. Alternatively, with *R* version 4.1, "|>" is a default operator. For explanations, see www.datacamp.com/community/tutorials/pipe-r-tutorial (accessed 04-08-2021)
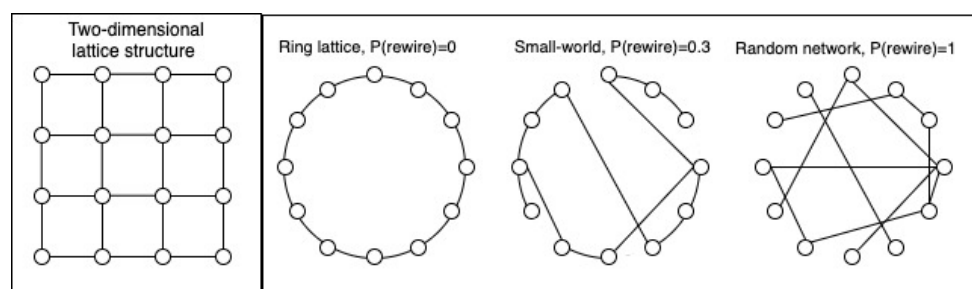
Figure 2. On the left, a two-dimensional lattice structure. On the right, a small-world and a random network are created from a ring lattice by varying the rewiring probability of edges.

3.2. Equilibrium Configurations

Once we have determined a network structure ω, we initiate a random configuration. Using Markov chain Monte Carlo algorithms, we can then update the network configurations sequentially such that the system's time spent in a configuration is equivalent to its Ising probability. This way we create a dynamic Ising system unfolding in time (For a smooth dynamics visualisation of an Ising system with varying β see http://bit-player.org/2019/glaubers-dynamics. Unfortunately, *R* is not suited for similar fast dynamics image updating).

We can use the dynamic system to study the stable states it settles into over time. We refer to these as *equilibrium configurations* [31,32]. Understanding dynamics means studying the equilibrium configurations as we vary its parameters (β and α). To aid this, we will summarise network configurations through their mean-field: $\bar{x}$. This is the average value of all nodes. For example, if $\bar{x} = 1$, all nodes are in state 1, and if $\bar{x} = -1$, all nodes are -1. These two values indicate perfectly aligned states. If $\bar{x} \approx 0$, on the other hand, nodes are randomly fluctuating. In Text Box 2, we show how to simulate the dynamics of $\bar{x}$ as we vary α and β.

Box 2. How to sample states from an Ising system [33].

We can easily study the average behaviour of an Ising system using four R-functions[a].

1. We generate a network structure, ω, by using one of our earlier introduced methods (see Text Box 1);
2. With `IsingSampler()`, we draw n states from an Ising distribution, assuming ω, α, and β;
3. We use `ParSim()` as a convenience function [33]. It automates the process of looping over sequences of α and β. For extensive simulations, it also allows for multi-core processing;
4. Lastly, `ggplot()` is used to visualise results. Figure 3 shows the result of looping over β, while Figure 4 shows a similar plot for α;
5. **Running example**: Using our previously generated network, we can simulate data as follows. See the supplementary material for the full simulation using *parSim()*.

```
library("IsingSampler")
samp <- IsingSampler(n = 1000, #sample size
graph = sw_network,
thresholds = rep(0,nrow(sw_network)),
beta = 2,
responses = c(-1L, 1L))#encoding
mean_spin <- apply(samp, 1, function(x) mean(x))
```

---

[a] Annotated and fully reproducible code, including all visualisations, can be found in the supplementary materials at https://osf.io/4z9u2/.

3.3. Dynamics of β: A Pitchfork Bifurcation

Figure 3 shows the distribution of $\bar{x}$ for different values of β (we have assumed $\alpha = 0$). The resulting pattern is known as a pitchfork bifurcation. A bifurcation refers to qualitative shifts in equilibrium points [31]. The shift happens around $\beta = 0.225$. Below this point, the equilibrium is composed of nodes randomly fluctuating, leading to $\bar{x} \approx 0$. For $\beta > 0.225$, two equilibria diverge towards -1 and 1. Conceptually, the Ising system shifts from a disordered ($\bar{x} = 0$) to an ordered state ($\bar{x} \pm 1$) as β is increased. The change from disorder to order occurs suddenly at a critical point (this is not very clear from our simulation). This critical point has some remarkable properties that have driven the physical interest in the Ising model [1,2,5]. To our knowledge, disorder to order critical points have **not** been quantitatively identified for social or psychological phenomena.
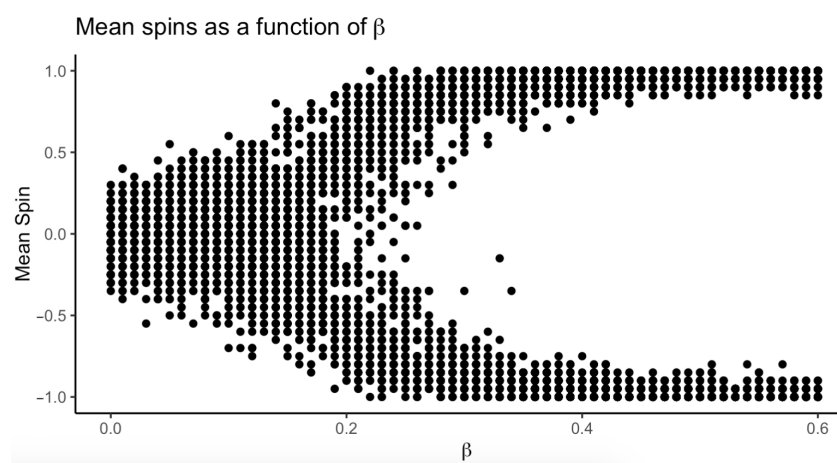

Figure 3. Result of sampling $\bar{x}$ from an Ising distribution, with $\alpha = 0$ and varying β between 0.00 and 0.06. The simulation assumes the small-world network structure of 40 variables from Text Box 1. Deviations from $\bar{x}$ become more likely for smaller networks and the pitchfork shape less pronounced.

Qualitatively, [8] uses the pitchfork bifurcation to model attitude polarisation. This follows from the idea that our attitudes (e.g., towards vegetarianism) are shaped by aligning network elements. Furthermore, if we interpret β as involvement, then an increase in involvement leads to more aligned and stronger attitudes. If this process takes place in multiple people with slight initial differences, we expect their attitudes to polarise— following the bifurcation—as involvement increases. Lastly, since β amplifies H, it also weighs the importance of α if that is a non-zero. If the external field is sufficiently strong relative to the connections, then an increase in β leads to nodes being aligned with their respective external fields rather than their neighbours (for further discussion, see [34]).

3.4. Dynamics of α: Hysteresis

Otherwise, socio-psychological research tends to focus on the behaviours of the Ising system in the aligned phase—we give multiple examples of this later. Here, the question is about how external influences, α, interact with the alignment-driven network. We can easily adapt the code from Text Box 2 to study the dynamics of α. Results are presented in Figure 4. More importantly, for this simulation, we assume a high β so we are in the alignment-ordered phase. Figure 4 shows that the system has two *tipping points*, at $\alpha = -1.5$ and 1.5 [35]. Below and above the tipping points, the system is aligned with the external field. Between the tipping points, the system can be aligned in both directions; as to which one will depend on the *history* of the system [36]. This dynamic is called *hysteresis*. We can understand it better by considering our perception of ambiguous stimuli [37].
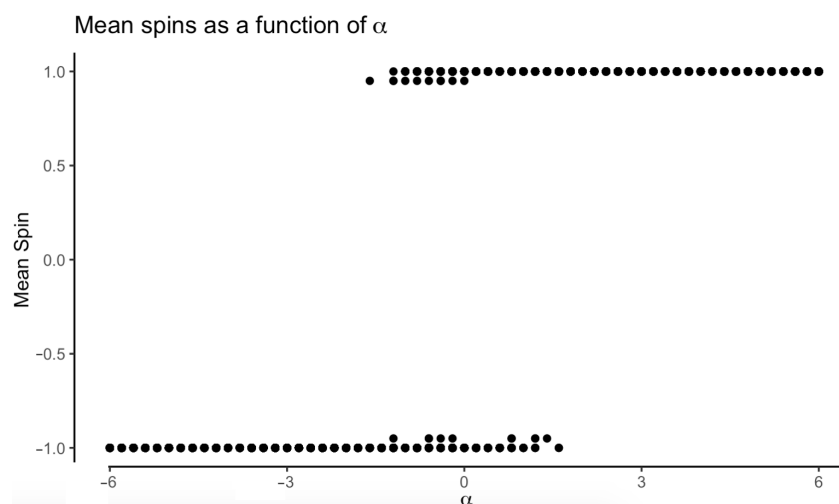
Figure 4. Result of sampling $\bar{x}$ from an Ising distribution with $\beta = 2$ (ordered phase) and varying α between -6 and 6. The simulation assumes the small-world network structure of 40 variables from Text Box 1. We see the outline of a hysteresis effect.

If we look across the illustrations in Figure 5, we will at some point experience a sudden transition between perceiving a face and a full human posture. Similar to the Ising system, the transition point depends on our starting point. What we perceive between the transition points ambiguously depends on history, similar to the states of our Ising system between -1.5 and 1.5. Furthermore, just after our perception has changed, if we return our gaze to the previous picture, our change in perception is not reversed. Rather, the change in perception persists. To undo the change, we need to go back several figures. This also applies to our Ising system: once we change α past one tipping point, we need to turn back to the other tipping point in order to reverse the change. That is the challenge of hysteresis: it is hard to reverse.
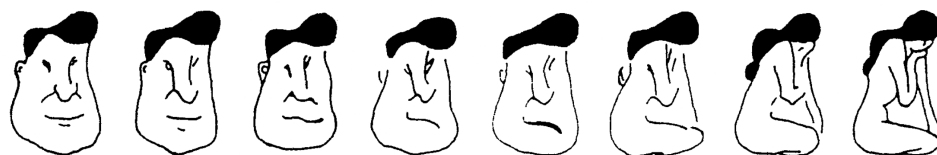

Figure 5. A series of ambiguous stimuli used by [37] to illustrate *hysteresis*. By looking across the illustrations in the figure, one will experience a transition in perception. This transition point depends on the direction we started from. Figures that are in between the transition points are ambiguous.

3.5. Summarising the Dynamics as a Cusp

So far, we have simulated two cases. We found a pitchfork bifurcation when we varied β while keeping $\alpha = 0$. We also found hysteresis as we varied α, with $\beta = 2$. Although this simulation approach is pedagogically useful, it has two shortcomings: (1) It does not give us a mathematical expression relating $\bar{x}$, α, and β. (2) It is cumbersome to repeat the simulation for many combinations of α and β. We can overcome both problems and achieve a relatively simple expression for the dynamics through a *mean-field approximation* (MFA) [38,39]. In Appendix A, we derive the MFA and perform a graphical bifurcation analysis to derive the dynamics. The central mathematical assumption of the MFA is that nodes interact with a mean field instead of their neighbours. This mean field approximates the effect of the individual interactions between nodes. This is computationally convenient as we avoid computing $H(X)$ and Z. The mean field is composed of $\bar{x}$ and the average

number of neighbours for the network, $\bar{d}$. Mathematically, we approximate H as follows (see Appendix A for further explanation):

$$H(\boldsymbol{x}) = - \sum_{<i,j>} \omega_{i,j} x_i x_j \approx - \sum_i x_i \bar{x} \bar{d} \qquad (3)$$

With this simplification and a few extra mathematical steps, we can determine the expected equilibrium points as we systematically vary α and β. The approximation works better for denser and larger networks [39]. We see the results in Figure 6. Figure 6A shows the hysteresis dynamics arising as we vary α while keeping $\beta = 2$; Figure 6B shows the pitchfork bifurcation appearing from varying β with $\alpha = 0$. Lastly, we can integrate the pitchfork and hysteresis into a single model, as seen in Figure 6C, to get a complete picture of the Ising dynamics. This is the so-called cusp–catastrophe model [31,40].

We can use it to summarise the dynamics of local alignment as follows. The transition from an disordered to an ordered state governed by alignment happens suddenly at a critical point as we increase β (pitchfork bifurcation). In the disordered phase, we have linear control over the system's state through the external field. We lose this control in the alignment's ordered phase. Here, perturbations of the external field have little impact—the system becomes resilient. That is, until we reach a tipping point after which the entire systems flips. Therefore, alignment induces both resilience but also the capacity for "dramatic" changes [41]. Furthermore, alignment induces two tipping points between which the system's state depends on its history. This also means that once we pass a tipping point, the change is hard to reverse. The dramatic and hardly reversible changes that are obtained in this region of the model space have inspired the naming of the model as a *catastrophe* model.
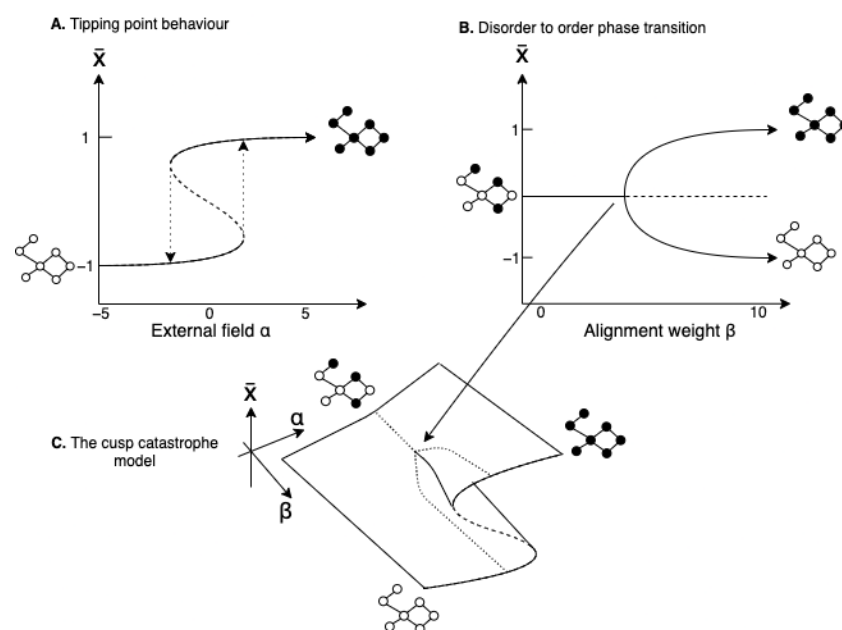

Figure 6. The cusp catastrophe model (**C**) presents a unified picture of the alignment dynamics arising from the dynamical Ising model. It encompasses pitchfork bifurcation (**B**) and hysteresis (**A**).

3.6. Variable Encoding

A key assumption of the Ising model is that nodes are dichotomous. However, different encodings are possible. Ref. [42] provides a thorough discussion of this matter, which we will briefly summarise here. There are two generally used options, $(1, -1)$ and $(1, 0)$. The different encodings make different theoretical assumptions about the dynamics. Nevertheless, we can always transform between different encodings once one is chosen. Ref. [43] covers the mathematical details which are implemented in *IsingFit::LinTransform()*.

We can understand the different consequences of $(1, -1)$ and $(1, 0)$ by considering how they impact the H, and consequently $P(X)$ (we leave α out, assuming it to be 0):

$$H(\boldsymbol{x}) = - \sum_{<i,j>} \omega_{i,j} x_i x_j \tag{4}$$

The alignment of 1 and -1 encoded states decreases H ($-x_i x_j = -(1 \cdot 1) = -(-1 \cdot -1) = -1$). This implies an increased probability of the configuration. Thus, configurations with many neighbours aligned at 1 or -1 become likely to occur. We therefore say that the system "wants" to align in both these directions. This is different for variable states coded as $(1, 0)$. In this case, the alignment of positive states 1 decreases H, but the alignment of 0 does not. In fact, H is unchanged by unaligned and 0 aligned neighbours ($-x_i x_j = -(0 \cdot 0) = -(1 \cdot 0) = 0$). Hence, the system does **not** "want" to align at 0, although it does want to align at 1. This is useful when we model a binary system that only "wants" to align in one direction. Present versus absent *symptoms* have been modelled this way [18]. In contrast, if the system "wants" to align in either of the directions, then the $(-1, 1)$ encoding provides a better description of the system. Positive versus negative *political attitudes* have been modelled with this encoding [8,17]. The crucial choice of whether to use the $(0, 1)$ versus $(-1, 1)$ configurations depends on whether one theoretically *expects negative states to increase alignment*; for political attitudes, the hypothesis that negative feelings of a political candidate increase the negative cognitions of that candidate is plausible; meanwhile, for depression, the hypothesis that the absence of one problems causes the absence of another may be less plausible.

3.7. Theoretical Applications

We can regard the theoretical Ising model as a *system archetype* [44]: a common structure (alignment process in a network) producing a characteristic behaviour (cusp dynamics). Knowledge of system archetypes can help theoretical progress in psychology, a feat many authors have called for [45–50]. This is because system archetypes allow researchers to skip the difficult step of crafting a novel model by recognizing that their phenomenon fits an existing one. We see this in a diverse range of fields such as economy [10,51,52], molecular biology [9,53,54], social sciences [11], and biological ecology [55–57]. Within psychology, the Ising model has a long-standing tradition of making a good first approximation within *opinion research*. Both within and across groups of people, opinions are argued to follow alignment processes [17,19,36,58]. More specifically, alignment has repeatedly been linked to our need for consistency of beliefs and aversion towards ambiguity [17,19]. Once this correspondence is established, many hitherto unrelated opinion phenomena can be integrated. For instance, ref. [17] argues that the Ising framework can explain cross-pressures, spillover effects, partisan cues, and ideological differences in attitude consensus. Similarly, ref. [59] argues that an Ising model of attitudes can explain the "mere thoughts" effect, cognitive dissonance, heuristic reasoning, and systematic reasoning. As mentioned, this work is extended by [8] who integrates facts about persuasion to explain political polarisation.

Similar ideas are found in climate research where the notion of *tipping points* has seen growing attention [60]. Early climate research focused on detecting and preventing environmental tipping points, which can have catastrophic consequences due to their sudden and hardly reversible nature [35,61–63]. One strategy of prevention is the enabling of large-scale societal changes. Once again, this has been theorised as possible due to our alignment-driven opinions, which give rise to *social tipping points* [64,65].

Taken together, these different lines of research converge on the point that opinion dynamics can be approximated by the Ising model. However, we also caution that opinion dynamics are likely to be more complex than any single simple model [66,67]. Therefore, we should regard the Ising model of opinions as a good first approximation while recognising its limitations. Further progress is likely to involve a combination of system archetypes providing a more thorough account of the complexities.

4. The Statistical Ising Model

In this section, we move from the theoretical to the statistical use. We first introduce the statistical application and its software implementations. Table 2 gives an overview of software packages related to statistical Ising estimation. We end this section with some general recommendations for choosing software. Central factors shaping this choice are *sample size*, *network architecture*, and further analysis options. We first motivate the recommendations by introducing network psychology, frequentist and Bayesian Ising estimations. As in the previous section, text boxes introduce the software packages in *R*.

4.1. A Decade of Statistical Ising Models

The Ising model has played an important role for network applications in psychology [15,58,68], where it formed the basis of the estimation of psychometric networks [16,20,69]. Over the last decade, there has been an exponential increase in articles based on network psychometrics [16]. The approach offers an alternative to the standard psychometric view in which psychometric variables are represented as effects of a latent variable [7,70]. Although network psychometrics represents an abstract statistical toolkit that can be used independently of one's antecedent theory, it aligns naturally with the view that psychological constructs, e.g., depression or conservatism, emerge from networks of causally interacting elements. These elements could be disorder symptoms [15] or our feelings, behaviours, and beliefs [19]. Because we cannot directly observe the interactions between variables, the psychometric challenge is to measure and estimate these. The general solution is to compute the *conditional dependence structure* from observed data [16,20]. This structure is chosen because it is uniquely identified, in contrast with directed acyclic graphs, and easier to interpret compared to unconditional dependency networks [20]. As we will see, this structure can be estimated from binary data using the Ising model.

4.2. eLASSO Estimation

We can now disregard the alignment interpretations of the Ising model we introduced in the previous section. Instead, we can view the Ising model as a probability distribution that is governed by main effects (α parameters) and pairwise interactions/edges (parameters in ω). The model then simply describes the joint probability distribution of a set of variables. We can use it to estimate the so-called pairwise Markov Random Field (pMRF) for binary variables [20,71]. The pMRF represents the probability distribution in terms of pairwise conditional dependencies between variables. Thus, the edges in ω represent the pairwise dependency after controlling for all other variables (e.g., partial correlations for continuous variables), while absent edges reflect conditional independence [72].

Ref. [69] introduced the estimation of pMRFs using the Ising model to the context of network psychometrics. Similar approaches have been used under various names (homogeneous association model, log-multiplicative association model) in other fields [73–75]. Within network psychometrics, ref. [69] developed the elASSO estimator for binary data, which combines pseudo-likelihood estimation with LASSO (Least Absolute Shrinkage Selection Operator) [76]. This combination is now the default estimation process and has since been expanded to ordinal, continuous, and mixtures of variables [23,77].

Table 2. Software packages in *R* relevant to estimating Ising models from binary data.

Package::Function()	Description	Pros	Cons	Encoding
IsingFit::IsingFit()	eLASSO estimation	Small–medium samples detecting present edges Applicable to >20 variables	Large samples Interpreting absent edges	(1, 0)
psychometrics::Ising()	Full maximum likelihood	Large samples Extensive further analysis options	Small samples Max of 20 variables	Any
mgm::mgm()	eLASSO	For mixtures of binary, continuous, and ordinal variables	Large samples Interpreting absent edges	(1, 0)
rIsing::Ising()	eLASSO	Small–medium samples Detecting present edges Applicable to >20 variables	Large samples Interpreting absent edges	(1, 0)
rbinnet::select_structure()	Bayesian estimation Slap and spike prior	Evidence of absent edges Model uncertainty Prior information use	Work in progress Prior information dependent	(1, 0)
BGGM::explore(type = "binary")	Bayesian estimation F-matrix prior	Evidence of absent edges Model uncertainty Prior information use	Prior information dependent	(1, 0)
BDgraph::bdgraph()	Bayesian model selection G-wishart Prior	Model uncertainty Prior information use	Prior information dependent	(1, 0)
IsingFit::LinTransform()	Transforms between (1, 0) and (1, −1) encodings	Works with unregularised models (psychometrics)		Any
NetworkComparisonTest::NCT()	Group comparison test	Works with eLASSO models		

Understanding eLASSO is quite straightforward if one is familiar with logistic regression (see [78] for an introduction). Ref. [69] uses the Ising model as a likelihood function to derive the conditional probability of an observed binary variable given all other measured variables. In Figure 7, this conditional probability is shown for each node of a simple three-node network. More importantly, if x_1, x_2, and x_3 are observed variables, this equation translates directly into a logistic regression. Hence, we can estimate β and α with one regression per node. A caveat is that we will have two estimates per edge, β_{ij} and β_{ji}, because each node serves both as the dependent and independent variables. However, they will converge as sample size goes to infinity. To solve this and complete the adjacency matrix ω, [69] uses the *and-rule*: any edge is the average β if *both* β_{ij} and β_{ji} are non-zero, otherwise $\omega_{ij} = 0$. A benefit of this node-wise approach is that we avoid computing Z. Consequently, a pseudo-likelihood estimation is needed for large networks.
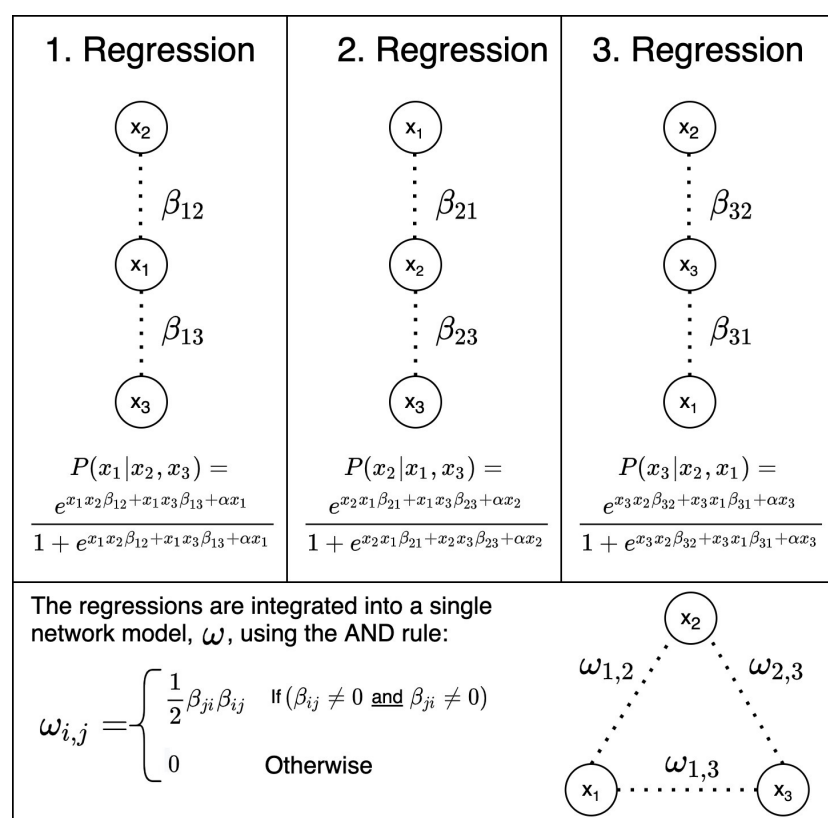

Figure 7. With a pseudo-likelihood estimation, we first estimate the neighbourhood of each node. This is performed through one logistic regression per node (here, we consider a simple three-node example). The bottom panel shows how we combine the neighbourhoods into a single network model, ω, through the AND rule: an edge is present if both β_{ij} and β_{ji} are non-zero. This step is necessary because each node is both the dependent and independent variables; hence, we have two β estimates.

This multiple logistic regression approach makes it apparent that we are performing multiple testing. This is a central difficulty in hypothesis testing because it inflates the rate of false positives. This issue was recognised by [69] who remedied this problem through LASSO regularisation. This procedure shrinks estimates towards zero, lowering the amount of detected edges and their strengths. The amount of shrinkage depends on a parameter ρ. Choosing the optimal ρ is performed through information criteria-based model selection. This approach to model selection depends on a hyper-parameter γ. More importantly, this parameter must be set by the analyst as it regulates the *error-type control* for eLASSO, which means it determines whether false positives or negatives are preferred. Lower γ favours more edges, while higher γ favours stronger shrinkage and

hence, sparser networks. Therefore, a low γ increases the false-positive rate, while a high γ inflates the false-negative rate. We will see that each estimator has a corresponding error-type controller.

In Text Box 3, we perform eLASSO estimation based on the sample from Text Box 2. We use the default of $\gamma = 0.25$. The result as well as the original network we generated in Text Box 1 are shown in Figure 8. We see that eLASSO recovers the network well, with two false negatives and no false positives. It should be noted that β is usually **not** identified when we fit statistical Ising models, therefore we do not get an estimate of it. The only exception to this is *psychometrics*, which allows for its estimation from multi-group data (we will discuss this option later).

The main argument for LASSO is its limiting effect on over-fitting for smaller samples, leading to better out-of-sample generalisability [79]. The sparsity induced by LASSO has also been argued to help to interpret networks as only the most important edges remain [69]. Hence, LASSO is advisable when the goal is to identify important present edges in small to medium samples.

Box 3. How to perform eLASSO estimation with *IsingFit*.

---

eLASSO with `Isingfit()` in R^a.

eLASSO is implemented in the *IsingFit* package [69]. It has the advantage of being easy to use since it only requires a single hyper-parameter γ to be set. It uses a $(0, 1)$ encoding and performs list-wise deletion in the case of missing data. Group comparisons of eLASSO models can be executed with the *NetworkComparisonTest* package.

Running Example: Using `Isingfit()`, we can recover the assumed network structure generated in Text Box 1. More importantly, for ease of interpretation and estimation, we use a simpler six-node network generated by altering the *n_nodes* argument. We can then re-sample 1000 states and fit an eLASSO model using *IsingFit()*. Figure 8 shows the original network (right) and a network estimated with the default value of $\gamma = 0.25$. We see that our estimated network recovers most present edges and generates no false positives.

```
library("IsingFit");
samp_ising0 = IsingFit(samp,
gamma =  0.25, #default
AND = TRUE)
```

---

[a] Annotated and fully reproducible code is found in the supplementary material at https://osf.io /4z9u2/. This includes all visualisation code.
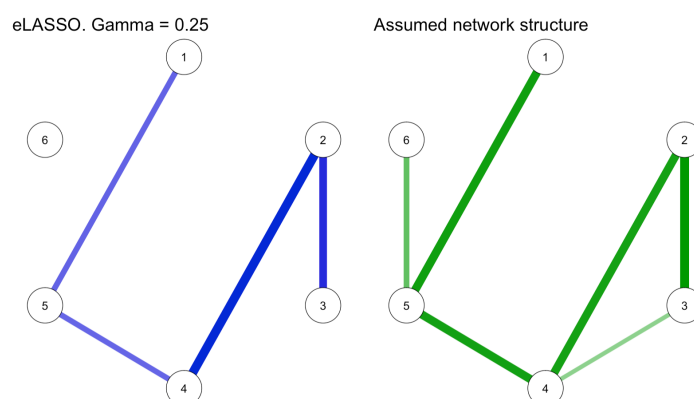

Figure 8. To the **right**, the "true" network structure (Text Box 1), which we simulated at 1000 observations per node from Text Box 2. To the **left**, we estimated the network using $\gamma = 0.25$.

For larger samples, controlling over-fitting through LASSO becomes less important. Instead, researchers should consider which additional analyses are required (group comparisons, measurement invariance, predictability, missing data handling). In particular, *psychometrics* and *BGGM*, which we will cover later, offer extensive additional functionalities.

Besides sample size, the choice of estimator also depends on the theoretical importance of *absent* edges (or negligibly small effects cf. [80]). Since eLASSO shrinks estimates towards zero, it increases the number of absent edges. The issue for estimated networks is that absent edges arise for two theoretically distinct reasons: (1) the lack of power to detect an existing effect, or (2) the correct detection of zero relation between variables. Since both cases are typically visualised as an absent edge, they are hard to distinguish in practise. This is problematic because they require different conclusions. In the former, we should remain uncertain, while we can build our confidence in no-effects in the latter. Problems arise when they are conflated as we get fooled into building unwarranted confidence in no-effects. Ref. [81] documents several instances of this issue in the literature.

4.3. Bayesian Estimation

We can distinguish the absence of evidence from the evidence of absence with the Bayesian estimation. This is implemented in *BDgraph*, *Rbinnet*, and *BGGM*. Their central difference to eLASSO is their computation of multiple hypotheses per edge ω_{ij}. *Rbinnet* evaluates $H_0 : \omega_{ij} = 0$ and $H_1 : \omega_{ij} \neq 0$. *BGGM* can perform an expanded three-hypotheses-per-node test: $H_1 : \omega_{ij} = 0$, $H_2 : \omega_{ij} > 1$ and $H_3 : \omega_{ij} < 1$. In all cases, we can use H_1 to clearly separate the evidence of absence from the absence of evidence.

A central component of Bayesian statistics is *prior probabilities* allowing researchers to influence the estimation process with background knowledge, as coded in a prior distribution over the parameters of the model. Unless the sample is very large, in which case the prior's impact becomes negligible, the priors pull estimates towards the center of the prior distribution. Network models involve many parameters, which makes it tricky to set individual prior probabilities. Additionally, little work has been performed on how to set priors for network psychometrics. Here, we suggest one computational workaround. In Text Box 5, we show how to easily compute a *robustness analysis*. That is, we repeat the analysis with different analysis choices (e.g., priors) to see if the results are robust in these changes.

We first perform Bayesian estimation with *rbinnet*, which is followed by a robustness analysis of the result [82]. We use *rbinnet* to compute two hypotheses per edge, $H_0 : \omega_{ij} = 0$ and $H_1 : \omega_{ij} \neq 0$. As explained in Text Box 4, we can use the hypotheses to generate a novel *edge-uncertainty* graph (introduced in [83]). The estimation depends on a prior-inclusion and precision parameter, which we must specify. In Text Box 5, we examine the robustness of the results on changes in the prior-inclusion and precision parameters.

Box 4. How to perform Bayesian Ising estimation using *rbinnet*. We show this information in three *edge-uncertainty graphs* in Figure 9.

Bayesian Estimation with *rbinnet*[a].

rbinnet is a new package for the Bayesian estimation of Ising models [82]. First, it uses pseudo-likelihood estimation to find good candidate values for each edge through the `fit_pseduolikelihood()` function. Once the candidate values are found, we compute the edge hypotheses, $H_0 : \omega_{ij} = 0$ and $H_1 : \omega_{ij} \neq 0$, with `select_structure()`. This requires 3 parameters to be set: a prior edge inclusion probability with a default of 0.5, a prior variance intercept, and a precision parameter.

Using the `select_strucuture()` output, we construct three edge uncertainty plots based on *Bayes factors* (BF). The inclusion BF of an edge, ω_{ij}, is computed as $BF_{\omega_{ij}} = \frac{H_1}{H_0}$, i.e., it computes how many times more an edge is likely to be included compared to not being included. Vice versa, smaller values indicate that we have evidence for an absent edge. Following [82], we say that $BF > 10$ is evidence of inclusion. $BF < 0.1$ is evidence of an absent edge. Between 10 and 0.1, we remain uncertain about the nature of edges. We show this information in three *edge-uncertainty graphs* in Figure 9. Blue edges indicate evidence of inclusion, grey means we are uncertain, while red is evidence of exclusion. The simulation is based on the 1000 samples we took earlier.

Running example: We can perform a Bayesian estimation of our simulated data with two functions from *rbinnet*[b]:

```
library("rbinnet") #rbinnet in not on CRAN yet.
#must be installed from GitHub (done in analysis script).
df_final <- as.matrix(samp)
starting <- fit_pseudoposterior(x = df_final,
prior_var = 1)$sigma
obj <- select_structure(x = df_final,
sigma = starting,
omega = 0.5,
prior_var_intercepts = 1,
precision = 0.975,
output_samples = TRUE,
number_iterations = 5e3,
number_burnin_iterations=100,
hierarchical = FALSE)
```

---

[a] Annotated and fully reproducible code is found in the supplementary material at https://osf.io/4z9u2/. This includes all visualisation code.

[b] Note that we changed the *n_nodes* argument in Text Box 1 from 40 to 6 for this estimation.

Box 5. Robustness analysis of *rbinnet* results. We see the results in Figure 10.

> **Robustness analysis with *parSim*[a].** Statistical estimation inevitably involves
> difficult analysis choices. In these cases, checking whether results are robust
> on different choices is a good strategy. The main downside is the extra effort
> and the results required to explain. We can ease the process with the *parsim*
> package [33]. It automatically loops over conditions and stores outputs. Here,
> we demonstrate this functionality with *rbinnet*. We first specify a range of prior
> inclusion probabilities and precision values to loop over. For each run, we store
> the posterior inclusion probability per edge. We see the results in Figure 10,
> which reveals four robustly included edges, two non-robust (smeared out),
> and nine robustly excluded variables.
>
> _____
> [a] Annotated and fully reproducible code is found in the supplementary material at https://osf.io
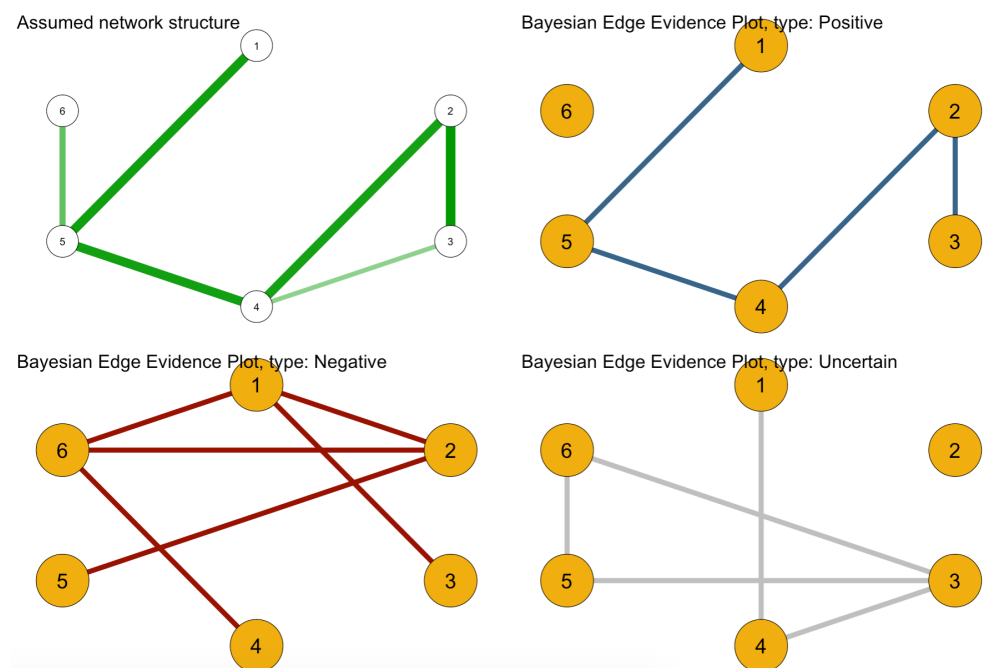>   /4z9u2/.

Figure 9. Result of *rbinnet* used to construct edge uncertainty plots. The edges of coloured graphs represent the inclusion Bayes factor $BF = \frac{P(\omega_{ij} \neq 0)}{P(\omega_{ij} = 0)}$. The red graph indicates the evidence of absence of an edge ($BF < 0.1$). The blue graph provides the evidence of inclusion of an edge ($BF > 10$). The grey graph shows the absence of evidence ($0.1 < BF < 10$).
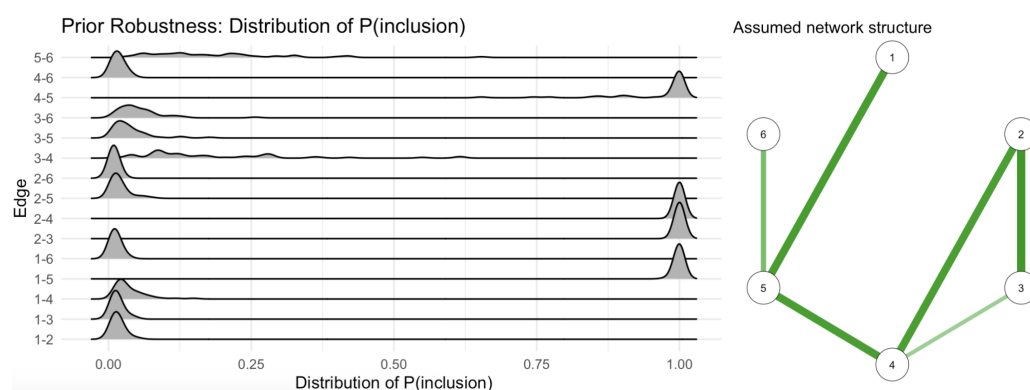
Figure 10. Robustness analysis based on *rbinnet*. The robustness of posterior inclusion probabilities is studied for various combinations of prior inclusion probabilities and precisions. We see that four edges have consistently high posterior inclusion probabilities (robust)—which are also the strongest in the assumed structure. Two edges have their probabilities "smeared out", indicating non-robustness. Finally, nine edges have robustly low posterior inclusion probabilities.

The second Bayesian option is the *BGGM* (we refer to the general R-package, not the specific estimation procedure of [84]). This package allows not only for exploratory estimation, as we have covered so far, but also for the confirmatory analysis of a pre-specified structure. It is a flexible function that accepts continuous, ordinal, binary, and a mixture of variable types. BGGM also offers a range of extra analysis options; for instance, missing data handling, group comparisons, and node predictability. In Text Box 6, we use *BGGM* to perform exhaustive hypothesis testing. That is, we test three hypotheses per edge: $H_1 : \omega_{ij} = 0$, $H_2 : \omega_{ij} > 1$, and $H_3 : \omega_{ij} < 0$. As explained in the text box, we used the three hypotheses to create the stacked bar plot of Figure 11.

Box 6. Bayesian Ising estimation using *BGGM*.

Bayesian Estimation with *BGGM*[a].

BGGM is a general Bayesian network estimator [84]. Using `BGGM::explore()`, we can perform a model search, which returns a distribution over likely network structures. It uses a matrix-F prior distribution with a single parameter for the analyst to set. If we want a single model to plot, we use `BGGM::select()`. In this process, we must set a Bayes factor cut-off for the inclusion of edges—the error-type control parameter of *BGGM()*. Here, we use *BGGM* to execute an exhaustive search of three hypotheses per edge. We summarise the information in the stacked bar plot in Figure 11. In this case, the absence of evidence is displayed as equal evidence across hypotheses.

Running example: We use two functions to perform model search and selection with *BGGM*[b]:

```
library("BGGM")
explore_network <- explore(samp,
type = "binary",
iter = 5000)
selected_network <- select(explore_network,
alternative="exhaustive",
bf_cut = 3)
```

---

[a] Annotated and fully reproducible code is found in the supplementary material at https://osf.io/4z9u2/. This includes all visualisation code.

[b] Note, we changed the *n_nodes* argument in Text Box 1 from 40 to 6 for this estimation.
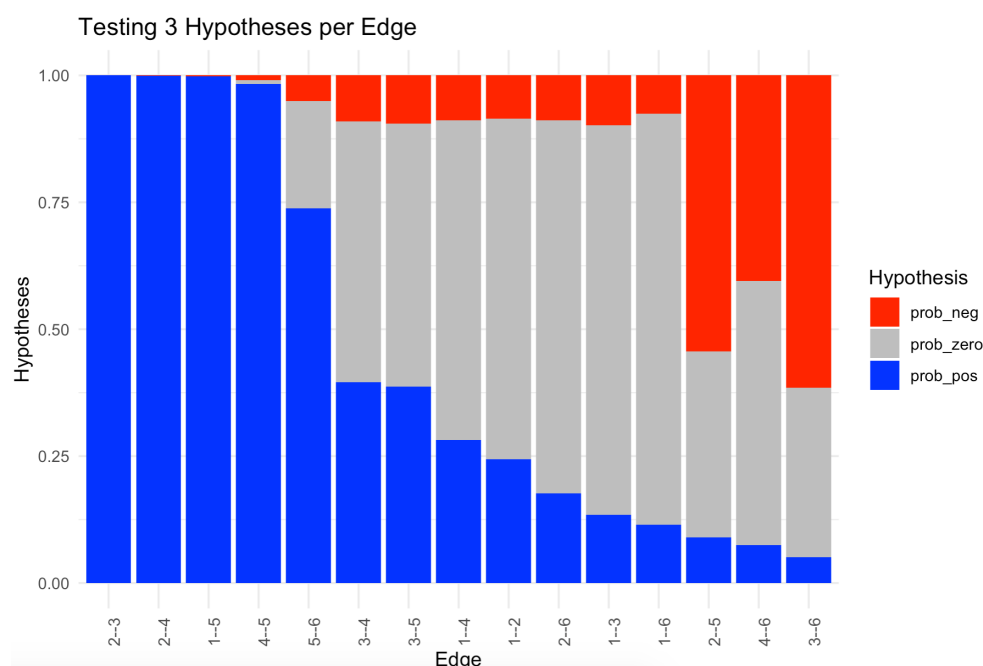
Figure 11. With *BGGM*, we compute three hypotheses per edge: $H_1 : \omega_{ij} = 0$ (grey), $H_2 : \omega_{ij} > 1$ (blue), and $H_3 : \omega_{ij} < 0$ (red). The absence of evidence then amounts to equal probabilities across the edge hypotheses.

4.4. Maximum Likelihood Estimation

Lastly, it is also possible to estimate Ising models using the *maximum likelihood* estimation (MLE) with *psychometrics*. This is a frequentist estimator without LASSO. In Text Box 7, we demonstrate this and explain the pruned MLE estimation. This offers a versatile package that bridges statistical network modelling with structural equation modelling (SEM) [85]. Following SEM, *psychometrics* allows for model-constraining, which is useful for group comparisons and measurement invariance testing, among others. A noteworthy feature of Ising models is that we can identify β through constraints. This requires multiple group data where we then constrain the network structure and external fields in order to be equal. We illustrate how this is executed in the analysis script. In Text Box 7, we estimate a fully saturated and pruned model and compare their fit using *psychometrics::compare()*.

4.5. Summary of Recommendation

In this section, we have covered the use of the Ising model as a likelihood function for parameter estimation. As we have seen, several *R* packages have implemented Ising estimation methods. The choice of Ising estimator depends on the research question and data. The central question is if further analysis is needed (missing data handling or group comparisons, for instance). In particular, *psychometrics* and *BGGM* offer extensive analysis options. Secondly, we highlighted that *sample size* and interpretability of absent edges are important to estimator choice. eLASSO is well-established and easy to use in small to medium sample sizes where present edges are in focus. For larger samples, full maximum likelihood is preferred over eLASSO. If absent edges are of theoretical importance, Bayesian estimators are good options as they compute the evidence of absence directly.

Box 7. Pruned maximum likelihood estimation using *psychometrics*. The comparison results are shown in Table 3.

> **Maximum likelihood estimation with *psychometrics*[a].**
>
> *psychometrics* offers several options for both estimation and further analyses. Here, we focus on *pruned* MLE estimation. With pruning, we first estimate a saturated model, then non-significant edges are removed before the entire model is re-estimated without them. Performing this in *psychometrics* involves three steps: **model specification** → **run model** → **prune model**. For our analysis, we first specify a model using the `ising()` function. At this stage, we choose encoding, parameters, optimiser, missing data handling. Once the model is set up, we execute it with `runmodel()`.
>
> **Running example**: We estimate a fully saturated model (degrees of freedom = 0) and a pruned model using our previously generated data. For the pruning, we assume a significance level of 0.05, which we control using *false discovery rates* (fdr). We then find the preferred model by comparing their fit indices. The comparison results are shown in Table 3[b].
>
> ```
> model_setup <- Ising(samp)
> model1 <- model_setup %>% runmodel()
> model2 <- model_setup %>%
> prune(alpha = 0.05,
> adjust = "fdr") %>% runmodel()
> psychometrics::compare(model1,model2)
> ```
>
> ---
> [a] Annotated and fully reproducible code is found in the supplementary material at https://osf.io /4z9u2/. This includes all visualisation code.
> [b] Note, we changed the *n_nodes* argument in Text Box 1 from 40 to 6 for this estimation.

Table 3. Output of model comparison using *psychometrics*. We estimated a saturated model (DF = 0) and a pruned model (DF = 39) from data generated in Text Box 2. A comparison of the two models' fit indices (AIC and BIC) shows that the pruned model is preferred as it has both lower AIC and BIC.

Model	DF	BIC	AIC
Model 1: saturated	0	4965	4862
Model 2: Pruned	11	4909	4858

5. The Practical Gap between Statistical and Theoretical Ising Use

In the paper, we have discussed both the statistical and theoretical uses of the Ising model. Although mathematically identical, they will often be used separately. This is because the statistical Ising model provides little evidence for the theoretical one. The theoretical model assumes that the system is alignment-driven based on which we can derive the cusp dynamics. To verify the theoretical model, we should ideally study a system over time in order to observe alignment and its consequent bifurcation and hysteresis behaviour. Subsequently, the statistical model can play an important role in parameter estimation. But the statistical model alone does not provide strong evidence for the theoretical model. This is because of the statistical equivalence between the network, latent variable, and item response-models [7]. The central tenet of the statistical model is that it makes weak assumptions, and consequently yields weak evidence for *any* specific data-generating structure such as the theoretical Ising model. This is also its strength. This means that it is a suitable tool whenever we want to explore our data and we have little intuition about the data-generating mechanism. In these cases, we can avoid additional theoretical assumptions—unlike latent variable models, directed acyclic graphs, etc.—by estimating the pRMF, which is always uniquely identified. This makes the statistical Ising model

a powerful exploratory tool: it identifies a correct data pattern regardless of what the underlying data-generating mechanism is. This result can then inspire and constrain further hypotheses that typically need more substantive information through experimental manipulations or time-series data.

Author Contributions: Conceptualization, A.F., D.B., S.E. and H.L.J.v.d.M.; methodology, A.F.; software, A.F.; validation, D.B., S.E. and H.L.J.v.d.M.; formal analysis, A.F.; writing—original draft preparation, A.F.; writing—review and editing, A.F., D.B., S.E. and H.L.J.v.d.M.; supervision, D.B., S.E. and H.L.J.v.d.M. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data is simulated using the code found on our OSF.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MFA   Mean-field approximation
MLE   Maximum likelihood estimation

Appendix A

Here, we briefly derive the *mean-field approximation* (MFA) of the Ising model (The derivation is inspired by Simon Dedeo's course on renormalization: www.complexityexplorer.org/courses/67-introduction-to-renormalization/). It consists of two parts. We first derive the MFA expression, which we subsequently perform a bifurcation analysis of in order to determine its expected equilibrium points.

To derive the MFA, we re-write H as the sum of "alignment" for each node i (we assume α to be 0, and we omit the double negative sign of the exponent):

$$H(\boldsymbol{x}) = \sum_{<i,j>} \omega_{i,j} x_i x_j = \sum_i \sum_j \omega_{i,j} x_i x_j = \sum_i x_i \sum_j \omega_{i,j} x_j \tag{A1}$$

The central assumption of the MFA is that nodes interact with an approximation of their neighbours. This approximation is the *mean-field*, $\bar{x}$, multiplied by the average number of neighbours in the system, $\bar{\omega}$. We approximate the neighbour interaction term as follows:

$$\sum_i x_i \sum_j \omega_{i,j} x_j \approx \sum_i x_i \bar{\omega} \bar{x} \tag{A2}$$

Conceptually we replace our fine grained description of the system by aggregating the nodes into a single mean field. In this process, we throw out information about individual variation. For a discussion of the MFA's validity under various network structures, see [39].

We can now inset our re-written H into the Ising equation:

$$P(\boldsymbol{X} = \boldsymbol{x}) = \frac{exp(\beta \sum_i x_i \bar{\omega} \bar{x})}{Z} \tag{A3}$$

The final step is to eliminate Z. We can accomplish this by obtaining the expected value of a node. We start by finding the probability that a single node is positive:

$$P(x) = \frac{exp(\beta x_i \bar{\omega} \bar{x})}{Z} = \tag{A4}$$

$$\frac{1}{Z} \prod_i exp(\beta x_i \bar{\omega} \bar{x}) = \tag{A5}$$

$$\frac{1}{Z} \cdot (exp(\beta x_1 \bar{\omega} \bar{x}) \cdot exp(\beta x_2 \bar{\omega} \bar{x}) \cdot exp(\beta x_3 \bar{\omega} \bar{x}) \dots) \tag{A6}$$

Each factor gives the probability of its corresponding node being positive. We see that this probability is equal for all nodes:

$$P(x_i = 1) = \frac{1}{Z} exp(1\beta \bar{\omega} \bar{x}) \tag{A7}$$

Based on this, we can obtain the expected value of the node $\mathbb{E}(x_i)$. This is the sum over each possible state value multiplied by its respective probability. Since each node has two states, 1 and -1, we obtain:

$$\mathbb{E}(x_i) = 1 \cdot P(x_i = 1) + (-1 \cdot P(x_i = -1)) = \frac{1}{Z} exp(\beta \bar{\omega} \bar{x}) - \frac{1}{Z} exp(-1\beta \bar{\omega} \bar{x}) \tag{A8}$$

We eliminate $\frac{1}{Z}$ by using $1 = P(x_i = 1) + P(x_i = -1)$. This leads us to an expression where $\frac{1}{Z}$ cancels out in the end:

$$\mathbb{E}(x_i) = \frac{\frac{1}{Z} exp(\beta \bar{\omega} \bar{x}) - \frac{1}{Z} exp(-\beta \bar{\omega} \bar{x})}{1} =$$

$$\frac{\frac{1}{Z} exp(\beta \bar{\omega} \bar{x}) - \frac{1}{Z} exp(-\beta \bar{\omega} \bar{x})}{P(x_i = 1) + P(x_i = -1)} =$$

$$\frac{\frac{1}{Z} exp(\beta \bar{\omega} \bar{x}) - \frac{1}{Z} exp(-\beta \bar{\omega} \bar{x})}{\frac{1}{Z} exp(\beta \bar{\omega} \bar{x}) + \frac{1}{Z} exp(-\beta \bar{\omega} \bar{x})} =$$

$$\frac{exp(\beta \bar{\omega} \bar{x}) - exp(-\beta \bar{\omega} \bar{x})}{exp(\beta \bar{\omega} \bar{x}) + exp(-\beta \bar{\omega} \bar{x})}$$

Lastly, we use a trigonometric rule to rewrite the expression in terms of $tanh()$. This gives us the final mean-field approximation where we also re-inset α:

$$\mathbb{E}(x_i) = tanh(\beta \bar{\omega} \bar{x} + \alpha) \tag{A9}$$

We can think of the MFA as a difference equation that predicts $\bar{x}$ at the next time point given the current mean spin: $\bar{x}_{t+1} = tanh(\beta \bar{\omega} \bar{x}_t + \alpha)$. The goal is then to find the equilibrium points. They are the values of $\bar{x}_t$ such that it is equal to $\bar{x}_{t+1}$, i.e., the output is not changed from the input. A bifurcation analysis amounts to determining these equilibria as we vary β and α [31]. We take a graphical approach where we plot $tanh()$ together with the diagonal $y = x$. The diagonal is useful because it generally shows where $\bar{x}_{t+1} = \bar{x}$. Therefore, we can find the equilibria of $tanh()$ by seeing where it intersects with the diagonal. Figure A1 shows $tanh()$ in blue for different values of β, assuming $\alpha = 0$. For low values of β, we see one intersection so that one equilibrium at $\bar{x} = 0$. As β increases, three equilibriums arise around 1, 0, and -1. It is important to notice that 0 is an unstable equilibrium, so we do not expect the system to be in this state (see [31] for an explanation). To construct a bifurcation plot, we systematically plot the equilibria as a function of β. By doing so, we obtain Figure A2 showing the pitchfork bifurcation we simulated earlier (see Figure 3). Following the same method but varying α, we can obtain the hysteresis plot of Figure 6A. Lastly, we trust [31] in that we can integrate our bifurcation analyses into the cusp catastrophe model.
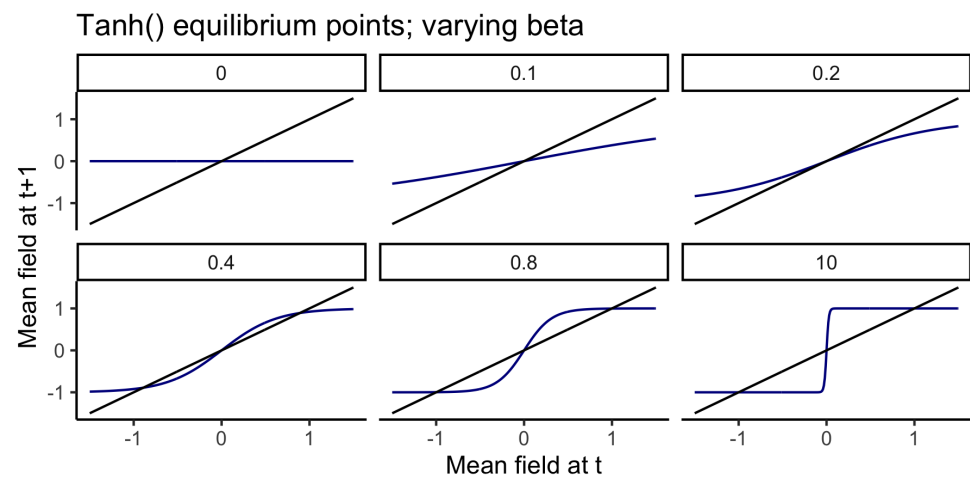
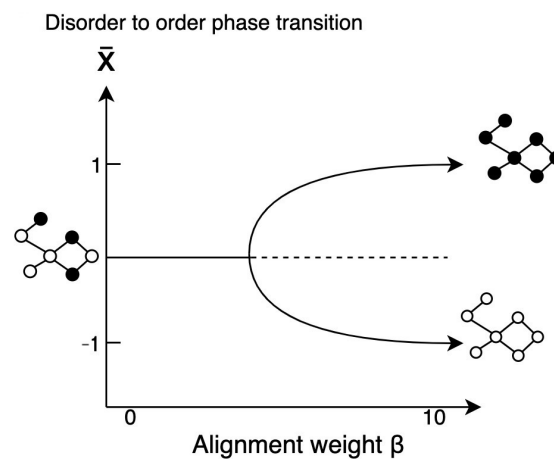**Figure A1.** Equilibrium plots for *tanh()* with varying β.

Figure A2. The bifurcation plot resulting from plotting equilibrium points of the MFA while varying the alignment weight β.

References

1. Brush, S.G. History of the Lenz-Ising model. *Rev. Mod. Phys.* **1967**, *39*, 883. [CrossRef]
2. Ernest, I. Ising: Beitrag zur theorie des ferromagnetismus. *Z. Für Phys.* **1926**, *31*, 253–258.
3. Onsager, L. Crystal Statistics. I. A Two-Dimensional Model with an Order-Disorder Transition. *Phys. Rev.* **1944**, *65*, 117–149. [CrossRef]
4. Bhattacharjee, S.M.; Khare, A. Fifty years of the exact solution of the two-dimensional Ising model by Onsager. *Curr. Sci.* **1995**, *69*, 816–821.
5. Sole, R.V. *Phase Transitions*; Primers in Complex Systems; OCLC: Ocn757257299; Princeton University Press: Princeton, NJ, USA, 2011.
6. Galam, S. Sociophysics: A review of Galam models. *Int. J. Mod. Phys. C* **2008**, *19*, 409–440. [CrossRef]
7. Kruis, J.; Maris, G. Three representations of the Ising model. *Sci. Rep.* **2016**, *6*, 1–11. [CrossRef] [PubMed]
8. van der Maas, H.L.J.; Dalege, J.; Waldorp, L. The polarization within and across individuals: The hierarchical Ising opinion model. *J. Complex Netw.* **2020**, *8*, cnaa010. [CrossRef]
9. Duke, T.A.J.; Bray, D. Heightened sensitivity of a lattice of membrane receptors. *Proc. Natl. Acad. Sci. USA* **1999**, *96*, 10104–10108. [CrossRef]
10. Bornholdt, S.; Wagner, F. Stability of money: Phase transitions in an Ising economy. *Phys. A Stat. Mech. Appl.* **2002**, *316*, 453–468. [CrossRef]
11. Stauffer, D. Social applications of two-dimensional Ising models. *Am. J. Phys.* **2008**, *76*, 470–473. [CrossRef]
12. Roudi, Y. Statistical physics of pairwise probability models. *Front. Comput. Neurosci.* **2009**, *3*, 22. [CrossRef]
13. Martyushev, L.M.; Seleznev, V.D. Maximum entropy production principle in physics, chemistry and biology. *Phys. Rep.* **2006**, *426*, 1–45. [CrossRef]
14. Jaynes, E.T. On the rationale of maximum-entropy methods. *Proc. IEEE* **1982**, *70*, 939–952. [CrossRef]

15. Borsboom, D.; Cramer, A.O. Network analysis: An integrative approach to the structure of psychopathology. *Annu. Rev. Clin. Psychol.* **2013**, *9*, 91–121. [CrossRef]
16. Robinaugh, D.J.; Hoekstra, R.H.A.; Toner, E.R.; Borsboom, D. The network approach to psychopathology: A review of the literature 2008–2018 and an agenda for future research. *Psychol. Med.* **2020**, *50*, 353–366. [CrossRef] [PubMed]
17. Brandt, M.J.; Sleegers, W.W. Evaluating belief system networks as a theory of political belief system dynamics. *Personal. Soc. Psychol. Rev.* **2021**, *25*, 159–185. [CrossRef]
18. Cramer, A.O.J.; van Borkulo, C.D.; Giltay, E.J.; van der Maas, H.L.J.; Kendler, K.S.; Scheffer, M.; Borsboom, D. Major Depression as a Complex Dynamic System. *PLoS ONE* **2016**, *11*, e0167490. [CrossRef]
19. Dalege, J.; Borsboom, D.; Harreveld, F.V.; Lunansky, G.; Maas, H.L.J.v.d. The Attitudinal Entropy (AE) Framework: Clarifications, Extensions, and Future Directions. *Psychol. Inq.* **2018**, *29*, 218–228. [CrossRef]
20. Epskamp, S.; Maris, G.K.J.; Waldorp, L.J.; Borsboom, D. *The Wiley Handbook of Psychometric Testing: A Multidisciplinary Reference on Survey, Scale and Test Development*; Wiley: Hoboken, NJ, USA, 2016; pp. 953–986.
21. Fried, E.I.; Bockting, C.; Arjadi, R.; Borsboom, D.; Amshoff, M.; Cramer, A.O.; Epskamp, S.; Tuerlinckx, F.; Carr, D.; Stroebe, M. From loss to loneliness: The relationship between bereavement and depressive symptoms. *J. Abnorm. Psychol.* **2015**, *124*, 256. [CrossRef] [PubMed]
22. Boschloo, L.; Borkulo, C.D.v.; Borsboom, D.; Schoevers, R.A. A Prospective Study on How Symptoms in a Network Predict the Onset of Depression. *Psychother. Psychosom.* **2016**, *85*, 183–184. [CrossRef] [PubMed]
23. Epskamp, S.; Borsboom, D.; Fried, E.I. Estimating psychological networks and their accuracy: A tutorial paper. *Behav. Res. Methods* **2018**, *50*, 195–212. [CrossRef]
24. R Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2020.
25. Newman, M.; Watts, D. Renormalization group analysis of the small-world network model. *Phys. Lett. A* **1999**, *263*, 341–346. [CrossRef]
26. Broido, A.D.; Clauset, A. Scale-free networks are rare. *Nat. Commun.* **2019**, *10*, 1–10. [CrossRef] [PubMed]
27. Watts, D.J.; Strogatz, S.H. Collective dynamics of 'small-world' networks. *Nature* **1998**, *393*, 440–442. [CrossRef] [PubMed]
28. Dalege, J.; Borsboom, D.; Van Harreveld, F.; Van den Berg, H.; Conner, M.; Van der Maas, H.L. Toward a formalized account of attitudes: The Causal Attitude Network (CAN) model. *Psychol. Rev.* **2016**, *123*, 2. [CrossRef]
29. Borsboom, D.; Cramer, A.O.; Schmittmann, V.D.; Epskamp, S.; Waldorp, L.J. The small world of psychopathology. *PLoS ONE* **2011**, *6*, e27407. [CrossRef]
30. Csardi, G.; Nepusz, T. The igraph software package for complex network research. *InterJournal* **2006**, *1695*, 1–9.
31. Strogatz, S.H. *Nonlinear Dynamics and Chaos*; Taylor & Francis Inc.: Abingdon, UK, 2014.
32. Guastello, S.J.; Koopmans, M.; Pincus, D. *Chaos and Complexity in Psychology*; Cambridge University Press: Cambridge, UK, 2010.
33. Epskamp, S. parSim: Parallel Simulation Studies. Available online: cran.r-project.org/web/packages/parSim/parSim.pdf (accessed on 29 September 2021).
34. Dalege, J.; van der Maas, H.L.J. Accurate by Being Noisy: A Formal Network Model of Implicit Measures of Attitudes. *Soc. Cogn.* **2020**, *38*, s26–s41. [CrossRef]
35. Scheffer, M.; Carpenter, S.; Foley, J.A.; Folke, C.; Walker, B. Catastrophic shifts in ecosystems. *Nature* **2001**, *413*, 591–596. [CrossRef]
36. Galam, S.; Gefen (Feigenblat), Y.; Shapir, Y. Sociophysics: A new approach of sociological collective behaviour. I. mean-behaviour description of a strike. *J. Math. Sociol.* **1982**, *9*, 1–13. [CrossRef]
37. Ditzinger, T.; Haken, H. Oscillations in the perception of ambiguous patterns a model based on synergetics. *Biol. Cybern.* **1989**, *61*, 279–287. [CrossRef]
38. Bianconi, G. Mean field solution of the Ising model on a Barabási–Albert network. *Phys. Lett. A* **2002**, *303*, 166–168. [CrossRef]
39. Waldorp, L.; Kossakowski, J. Mean field dynamics of stochastic cellular automata for random and small-world graphs. *J. Math. Psychol.* **2020**, *97*, 102380. [CrossRef]
40. van der Maas, H.L.J.; Kolstein, R.; van der Pligt, J. Sudden Transitions in Attitudes. *Sociol. Methods Res.* **2003**, *32*, 125–152. [CrossRef]
41. Siegenfeld, A.F.; Bar-Yam, Y. An Introduction to Complex Systems Science and its Applications. *arXiv* **2019**, arXiv:1912.05088.
42. Haslbeck, J.M.; Epskamp, S.; Marsman, M.; Waldorp, L.J. Interpreting the Ising model: The input matters. *Multivar. Behav. Res.* **2020**, *56*, 303–313. [CrossRef] [PubMed]
43. Kruis, J. Transformations of mixed spin-class Ising systems. *arXiv* **2020**, arXiv:2006.13581.
44. Meadows, D.H.; Meadows, D.L.; Randers, J.; Behrens, W.W. *The Limits to Growth*; Yale University Press: New Haven, CT, USA, 1972.
45. Oberauer, K.; Lewandowsky, S. Addressing the theory crisis in psychology. *Psychon. Bull. Rev.* **2019**, *26*, 1596–1618. [CrossRef]
46. Fried, E.I. Lack of theory building and testing impedes progress in the factor and network literature. *PsyArXiv* **2020**. [CrossRef]
47. Mischel, W. *The Toothbrush Problem*; Association for Psychological Science: Washington, DC, USA, 2008; Volume 21.
48. Borsboom, D.; van der Maas, H.; Dalege, J.; Kievit, R.; Haig, B. Theory Construction Methodology: A practical framework for theory formation in psychology. *PsyArXiv* **2020**. [CrossRef]

49. Robinaugh, D.; Haslbeck, J.M.B.; Waldorp, L.; Kossakowski, J.J.; Fried, E.I.; Millner, A.; McNally, R.J.; van Nes, E.H.; Scheffer, M.; Kendler, K.S.; Borsboom, D. Advancing the Network Theory of Mental Disorders: A Computational Model of Panic Disorder. *PsyArXiv* **2019**. [CrossRef]

50. Muthukrishna, M.; Henrich, J. A problem in theory. *Nat. Hum. Behav.* **2019**, *3*, 221–229. [CrossRef] [PubMed]

51. Zhou, W.X.; Sornette, D. Self-organizing Ising model of financial markets. *Eur. Phys. J. B* **2007**, *55*, 175–181. [CrossRef]

52. Hosseiny, A.; Bahrami, M.; Palestrini, A.; Gallegati, M. Metastable Features of Economic Networks and Responses to Exogenous Shocks. *PLoS ONE* **2016**, *11*, e0160363. [CrossRef]

53. Weber, M.; Buceta, J. The cellular Ising model: A framework for phase transitions in multicellular environments. *J. R. Soc. Interface* **2016**, *13*, 20151092. [CrossRef]

54. Matsuda, H. The Ising model for population biology. *Prog. Theor. Phys.* **1981**, *66*, 1078–1080. [CrossRef]

55. Nareddy, V.R.; Machta, J.; Abbott, K.C.; Esmaeili, S.; Hastings, A. Dynamical Ising model of spatially coupled ecological oscillators. *J. R. Soc. Interface* **2020**, *17*, 20200571. [CrossRef] [PubMed]

56. Wang, Y.; Badea, T.; Nathans, J. Order from disorder: Self-organization in mammalian hair patterning. *Proc. Natl. Acad. Sci. USA* **2006**, *103*, 19800–19805. [CrossRef] [PubMed]

57. Bialek, W.; Cavagna, A.; Giardina, I.; Mora, T.; Silvestri, E.; Viale, M.; Walczak, A.M. Statistical mechanics for natural flocks of birds. *Proc. Natl. Acad. Sci. USA* **2012**, *109*, 4786–4791. [CrossRef]

58. Maas, H.L.J.V.D.; Dolan, C.V.; Grasman, R.P.P.P.; Wicherts, J.M.; Huizenga, H.M.; Raijmakers, M.E.J. A dynamical model of general intelligence: The positive manifold of intelligence by mutualism. *Psychol. Rev.* **2006**, *113*, 842–861. [CrossRef]

59. Dalege, J.; Borsboom, D.; van Harreveld, F.; van der Maas, H.L. The Attitudinal Entropy (AE) Framework as a general theory of individual attitudes. *Psychol. Inq.* **2018**, *29*, 175–193. [CrossRef]

60. Milkoreit, M.; Hodbod, J.; Baggio, J.; Benessaiah, K.; Calderón-Contreras, R.; Donges, J.F.; Mathias, J.D.; Rocha, J.C.; Schoon, M.; Werners, S.E. Defining tipping points for social-ecological systems scholarship—an interdisciplinary literature review. *Environ. Res. Lett.* **2018**, *13*, 033005. [CrossRef]

61. Scheffer, M.; Bascompte, J.; Brock, W.A.; Brovkin, V.; Carpenter, S.R.; Dakos, V.; Held, H.; van Nes, E.H.; Rietkerk, M.; Sugihara, G. Early-warning signals for critical transitions. *Nature* **2009**, *461*, 53–59. [CrossRef] [PubMed]

62. Lenton, T.M.; Rockström, J.; Gaffney, O.; Rahmstorf, S.; Richardson, K.; Steffen, W.; Schellnhuber, H.J. Climate tipping points—Too risky to bet against. *Nature* **2019**, *575*, 592–595 [CrossRef]

63. Lenton, T.M. Early warning of climate tipping points. *Nat. Clim. Chang.* **2011**, *1*, 201–209. [CrossRef]

64. Otto, I.M.; Donges, J.F.; Cremades, R.; Bhowmik, A.; Hewitt, R.J.; Lucht, W.; Rockström, J.; Allerberger, F.; McCaffrey, M.; Doe, S.S. Social tipping dynamics for stabilizing Earth's climate by 2050. *Proc. Natl. Acad. Sci. USA* **2020**, *117*, 2354–2365. [CrossRef] [PubMed]

65. Bentley, R.A.; Maddison, E.J.; Ranner, P.H.; Bissell, J.; Caiado, C.; Bhatanacharoen, P.; Clark, T.; Botha, M.; Akinbami, F.; Hollow, M. Social tipping points and Earth systems dynamics. *Front. Environ. Sci.* **2014**, *2*, 35. [CrossRef]

66. Galesic, M.; Olsson, H.; Dalege, J.; van der Does, T.; Stein, D.L. Integrating social and cognitive aspects of belief dynamics: Towards a unifying framework. *J. R. Soc. Interface* **2021**, *18*, 20200857. [CrossRef]

67. Page, S.E. *The Model Thinker: What You Need to Know to Make Data Work for You*, 1st ed.; OCLC: on1028523969; Basic Books: New York, NY, USA, 2018.

68. Borsboom, D. A network theory of mental disorders. *World Psychiatry* **2017**, *16*, 5–13. [CrossRef]

69. Van Borkulo, C.D.; Borsboom, D.; Epskamp, S.; Blanken, T.F.; Boschloo, L.; Schoevers, R.A.; Waldorp, L.J. A new method for constructing networks from binary data. *Sci. Rep.* **2014**, *4*, 1–10. [CrossRef]

70. Fried, E.I. Theories and Models: What They Are, What They Are for, and What They Are About. *Psychol. Inq.* **2020**, *31*, 336–344. [CrossRef]

71. Kindermann, R.; Snell, J.L. *Markov Random Fields and Their Applications*; OCLC: 1030357447; American Mathematical Society: Providence, RI, USA, 2012.

72. Cox, D.R.; Wermuth, N. Linear dependencies represented by chain graphs. *Stat. Sci.* **1993**, *8*, 204–218. [CrossRef]

73. Anderson, C.J.; Vermunt, J.K. Log-Multiplicative Association Models as Latent Variable Models for Nominal and/or Ordinal Data. *Sociol. Methodol.* **2000**, *30*, 81–121. [CrossRef]

74. Wickens, T.D. *Multiway Contingency Tables Analysis for the Social Sciences*; Psychology Press: East Sussex, UK, 2014.

75. Marsman, M.; Borsboom, D.; Kruis, J.; Epskamp, S.; van Bork, R.; Waldorp, L.J.; Maas, H.L.J.v.d.; Maris, G. An Introduction to Network Psychometrics: Relating Ising Network Models to Item Response Theory Models. *Multivar. Behav. Res.* **2018**, *53*, 15–35. [CrossRef]

76. Ravikumar, P.; Wainwright, M.J.; Lafferty, J.D. High-dimensional Ising model selection using l1-regularized logistic regression. *Ann. Stat.* **2010**, *38*, 1287–1319. [CrossRef]

77. Haslbeck, J.M.B.; Waldorp, L.J. mgm: Estimating Time-Varying Mixed Graphical Models in High-Dimensional Data. *arXiv* **2020**, arXiv:1510.06871.

78. Hosmer, D.W., Jr.; Lemeshow, S.; Sturdivant, R.X. *Applied Logistic Regression*; John Wiley & Sons: Hoboken, NJ, USA, 2013; Volume 398.

79. Epskamp, S.; Kruis, J.; Marsman, M. Estimating psychopathological networks: Be careful what you wish for. *PLoS ONE* **2017**, *12*, e0179891. [CrossRef]

80. Meehl, P.E. Why Summaries of Research on Psychological Theories are Often Uninterpretable. *Psychol. Rep.* **1990**, *66*, 195–244. [CrossRef]
81. Williams, D.R.; Briganti, G.; Linkowski, P.; Mulder, J. On Accepting the Null Hypothesis of Conditional Independence in Partial Correlation Networks: A Bayesian Analysis. *PsyArXiv* **2021**. Available online: psyarxiv.com/7uhx8 (accessed on 27 July 2021).
82. Marsman, M.; Huth, K.; Waldorp, L.; Ntzoufras, I. Objective Bayesian Edge Screening and Structure Selection for Networks of Binary Variables. *PsyArXiv* **2020**, *26*. Available online: psyarxiv.com/dg8yx/ (accessed on 15 July 2021).
83. Huth, K.; Luigjes, J.; Goudriaan, A.; van Holst, R. Modeling Alcohol Use Disorder as a Set of Interconnected Symptoms-Assessing Differences between Clinical and Population Samples and Across External Factors. *PsyArXiv* **2021**. Available online: psyarxiv.com/93t2f/ (accessed on 25 June 2021).
84. Williams, D.R.; Mulder, J. Bayesian hypothesis testing for Gaussian graphical models: Conditional independence and order constraints. *J. Math. Psychol.* **2020**, *99*, 102441. [CrossRef]
85. Epskamp, S.; Isvoranu, A.M.; Cheung, M. Meta-analytic Gaussian Network Aggregation. *PsyArXiv* **2020**. [CrossRef]