UC Irvine

UC Irvine Electronic Theses and Dissertations

Title

Cultural Consensus Theory on Network Structures

Permalink

https://escholarship.org/uc/item/4zn4j458

Author

Agrawal, Kalin

Publication Date

2015

Copyright Information

This work is made available under the terms of a Creative Commons Attribution-NonCommercial-ShareAlike License, availalbe at https://creativecommons.org/licenses/by-nc-sa/4.0/

Peer reviewed|Thesis/dissertation

eScholarship.org

Powered by the California Digital Library
University of California

UNIVERSITY OF CALIFORNIA,
IRVINE

Cultural Consensus Theory on Network Structures

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Mathematical Behavioral Sciences

by

Kalin Alak Agrawal

Dissertation Committee:
Professor William H. Batchelder, Chair
Professor Louis Narens
Professor Emeritus John Boyd

2015

© 2015 Kalin Alak Agrawal

DEDICATION

To my parents.

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

ACKNOWLEDGMENTS

My graduate studies and this dissertation would not have been possible without support from a number of people and institutions at UC Irvine and elsewhere.

I would like to thank my advisor and committee chair, Bill Batchelder, for his enduring commitment to and support of my studies. He is clear and jovial, has a bewildering depth of knowledge, and is admirably forthright. He, along with my advancement and dissertation committee members John Boyd, Katherine Faust, Joachim Vandekerckhove, Jennifer Trueblood, and Jeff Krichmar provided excellent feedback at several stages in the development of this dissertation. Thanks in particular to Louis Narens, Kimberly Jameson, and Doug White for drawing me into research early on in my time at UC Irvine. I am grateful to Janet Phelps and Joanna Kerner for their wonderful and friendly support at the IMBS. My fellow MBS students gave me plenty of academic distraction, immeasurable technical help, and great friendship. In particular, thanks to Giorgio Gosti, Matt Feldmann, Ray Mendoza, Daniel Jessie, Tolga Oztan, and Santiago Guisasola. Thanks, too, to my peers in the Cognitive Psychometrics Lab, including Gregory Alexander, Royce Anders, Zita Oravecz, James Negen, Francis Lee, and Soobin Kristen Ahn for providing feedback on presentations and assisting with collecting data. Before I started at UC Irvine, encouragement to pursue graduate school came from Professors Fulvio Domini and Tom Dean, and family members Avani Agarwal and Patrick Sellers. I especially thank Joanne Christopherson for her generous and energetic support of my interest in education and for helping with my pursuits after graduate school.

This dissertation and accompanying research is produced with financial support from the UC Irvine School of Social Sciences and Institute for Mathematical Behavioral Sciences. I am thankful for their support. Support is also gratefully acknowledged from grants to William H. Batchelder from the Air Force Office of Scientific Research and Army Research Office. Additional support has come from the organization of the 2012 International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction.

Janny Li has given me a balance and perspective that I had not anticipated, but that I cannot imagine having gone without. She has supported me in producing this dissertation and in so much more, and I thank her for it.

I am indebted to my parents and brother in so many ways that I cannot possibly list them. I cherish their love and support, and cannot thank them enough.

CURRICULUM VITAE

Kalin Alak Agrawal

EDUCATION

Doctor of Philosophy
in Social Science - Mathematical Behavioral Sciences **2015**
University of California, Irvine *Irvine, California*

Bachelor of Science in Computer Science **2002**
Brown University *Providence, Rhode Island*

TEACHING EXPERIENCE

Teaching Assistant **2008–2015**
University of California, Irvine *Irvine, California*

REFEREED CONFERENCE PUBLICATIONS

Kalin Agrawal, William H. Batchelder **Apr 2012**
Cultural Consensus Theory: Aggregating Signed Graphs Under a Balance Constraint
2012 International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction

ABSTRACT OF THE DISSERTATION

Cultural Consensus Theory on Network Structures

By

Kalin Alak Agrawal

Doctor of Philosophy in Mathematical Behavioral Sciences

University of California, Irvine, 2015

Professor William H. Batchelder, Chair

Cultural Consensus Theory (CCT) consists of cognitive models for aggregating the responses of informants to test items about some domain of their shared cultural knowledge. This paper proposes variants of CCT for pooling undirected, signed graphs collected from error-prone and biased informants. Informants provide dichotomous 'plus' or 'minus' responses to judgments on all possible ties among a fixed set of named nodes. The primary goal is to achieve a single pooled signed graph that better reflects the "wisdom of the crowd" for small datasets than simple marginal averaging of responses.

These models break the typical CCT assumption of conditional independence of question items in two ways. First, the models attribute the quality of a response to properties of the pair of nodes in question. Both continuous and discrete nodal properties add dependencies between responses by the same informant. Second, a hard constraint on the aggregate graph imposes dependencies among the values of the aggregate graph ties.

We show that graph elicitations of different kinds warrant the use of new CCT models and that the models discussed here illuminate aspects of the underlying graph structure that are otherwise hidden using standard CCT methods.

A large component of the work involves novel estimation algorithms that operate under hard constraints on discrete parameters, something that has not been done before with CCT. Question ordering for undirected graphs and an incomplete design are discussed as well as possible extensions and related work.

Chapter 1

Introduction

1.1 Background

Cultural Consensus Theory (CCT) is a framework for aggregating (pooling, fusing) quantitative responses to multiple questions of a shared domain elicited from respondents. First developed in the 1980s, CCT methods have been used as a measurement tool in a wide array of social science research involving researchers who do not know the shared beliefs of the group of respondents being questioned, but where they do assume the respondents have been exposed to (and can report on) the culturally-correct answers to their questions. We call this shared, culturally-correct knowledge of the respondents' notion of the true answers to these questions the "consensus" or "cultural consensus".[1]

The task of the researchers using CCT, then, is to measure the consensus from these respondents' individual responses, where the researchers do not know the consensus ahead of time, nor do they know which respondents have more or less knowledge. Indeed the task of CCT is

[1] There are two main notions of the term "consensus", one of which is the *process* of Group Decision Making that guides a group of respondents towards a shared opinion on a topic as measured a number of items (e.g. a questionnaire), and the other is the *state* of agreement among such respondents (Herrera-Viedma et al., 2014). This paper on CCT concerns the latter form of consensus.

akin to simultaneously determining an answer key for a test given to test-takers and scoring the ability of those test-takers with respect to the answer key (Batchelder and Romney, 1988; Karabatsos and Batchelder, 2003; Oravecz et al., 2013). Additionally, as with related Item Response Theory (IRT) methods, CCT gauges the difficulty of the questions. In CCT, these tasks are all accomplished with cognitively-based response process models with consensus answers and cognitive characteristics of the respondents estimated endogenously.

For the remainder of this document, the term "experts" will be used to refer to respondents in the context of CCT and gathered data. This term is the author's shorthand and acknowledgement of one of the main field uses of CCT: to identify the experts among the surveyed respondents for follow-up study.

Thus, CCT is useful to researchers in situations where (a) the consensus knowledge of the experts is unknown to the researcher ahead of time, (c) the researcher has access to limited number of experts who may or may not have had equal access to this shared cultural knowledge, (c) the researcher can construct a relevant questionnaire but does not know which questions are more or less difficult, and (d) the researcher does not know much about the characteristics of the expert respondents.

While CCT models have been developed to aggregate responses on scales such as ordinal (Anders and Batchelder, 2015) and continuous (Batchelder et al., 2010) the canonical variant of CCT is the General Condorcet Model (GCM) (Romney et al., 1986; Batchelder and Romney, 1988; Karabatsos and Batchelder, 2003; Batchelder and Anders, 2012). The GCM applies to survey conditions involving a set of forced-choice, true/false questions pertaining to a single domain of expert response competence, and where the researcher does not know the culturally correct answers (consensus) ahead of time. Anthropological, and ethnographic applications abound in CCT literature (e.g. Weller, 2007) where the canonical true/false questionnaire format has been used in such cases as medical anthropology and linguistics (e.g. Oravecz et al., 2013). Questions might read something like “Human consciousness survives after death; T/F?” or “Alzheimer’s Disease is contagious; T/F?”.

CCT works well and has been used widely in such cases when question items are presumed not to intertwine, though share a common domain. This presumption may not be reasonable when experts respond to one question based on their response to another one or when question items depend on one another in some way. Recent work in CCT scales has begun to distinguish between observable measurement scales (such as the dichotomous forced choice task) and that of internal scale representation (such as a probability scale) (Batchelder and Anders, 2012).

This paper applies the CCT framework to dichotomous forced-choice questions that are designed to elicit the relationships between pairs of items (i.e. paired comparisons). Besides previous work in such *pairwise* CCT (e.g. Batchelder et al., 1997; Butts, 2003; Batchelder et al., 2010; Agrawal and Batchelder, 2012), there is interest in the informant accuracy of reports on the same social network coming from conflicting sources with heterogeneous credibility (An and Schramski, 2015).

Consider the motivations for furthering these models. We may have experts of varying abilities or varying access to underlying networks. Situations where field researchers could use this might include:

- *Social Psychology experiments in pairwise social relationships:* E.g. the kindergarten observers.
- *Criminology*: Detection of rivalries between gangs or covert terrorist networks.
- *Marketing and product analysis*: Classification of products into "high", "medium" and "low" class based on similarity, where there is no external scale for comparison.
- *Linguistics*: Similarity or use of terminology.
- *Eyewitness testimony*: Determine probable state of items or social network after-the-fact.

In particular, we consider the situation where responses to questions on all *undirected* pairs of a fixed set of nodes are collected from experts as *response graphs* and a single consensus graph is determined. Posed questions on pairs are generally inherently directed, which will be discussed as well. Questionnaires on all *directed* pairs from a fixed set of items (nodes) therefore elicit complete directed graphs (digraphs) from each expert. And applying certain CCT models to pool these expert response digraphs results in a consensus digraph that may be taken as the culturally-correct and culturally-true natures of the relationships among those nodes. Of particular interest to the author is the potential for CCT to extend to social networks, where the nodes represent human individuals, and the consensus graph or digraph can be regarded as a consensus *social network* on the relationships asked about in the questionnaire. For example, a survey of this type might ask each of several expert observers of a kindergarten classroom whether, for each directed pair of children, one child likes the other or not (e.g. "Does Alice like Bob?", "Does Bob like Carol?").

One benefit of applying CCT to graphs is that the elicitation provides multiple points of data on each relationship and can recover graphs with responses from only handfuls of experts. The graph modeling process often takes an "$n = 1$" sample and tries to find where that single graph sample came from. Given what we know about informant accuracy in general, and the variability and unreliability of recognition memory , we should be suspicious when utilizing such single-sample derived graphs as bases for analysis. The literature lacks available data for these multiple-expert social network response graphs in applied scenarios and is especially lacking in such datasets that can be used for model validation (where the ground truth is known).

Pairwise CCT is not new, and we leverage preceding work. CCT has been applied to pairwise judgments as judgments on existence of network ties as well as signed ties:

- Batchelder et al. (1997) is probably the closest model to what we have going on here.
- Butts (2003) presents a Bayesian model for digraph aggregation and estimation of heterogeneous expert response parameters. Uses hit rates and false alarm rates instead of the 2HT model.
- Batchelder (2009) developed a CCT model for aggregating digraphs. Arcs of a particular social network were presented as questionnaire items.
- Agrawal and Batchelder (2012) presented the a similar GCM-Tie model, but used the prior on the consensus graph as a way to constrain the aggregate to be balanced (i.e. such that the node could be partitioned into one or two groups).
- Brusco et al. (2012) proposed a graph aggregation method that puts graphs from multiple experts together to get a "best fit" consensus graph, given certain informational constraints on the consensus. This is an example of a more general discussion of multi-objective optimization. Pairwise CCT is important because it performs this optimization task using objectives grounded in theory about the experts rather than informational properties of the consensus. Another benefit of CCT on graphs is that it performs this optimization task using objectives grounded in theory about the experts rather than informational properties of the consensus.

It is important to mention that CCT models are now often analyzed in the Bayesian hierarchical framework. Hierarchical modeling is used when practical, and we discuss possibilities for future research to extend the models hierarchically when the models are not. Beyond a model likelihood, this requires further specification of model priors, hyperpriors and sometimes special methods for parameter inference. GCM models are set up as hierarchical Bayesian models (e.g. Anders and Batchelder, 2012; Batchelder and Anders, 2012; Oravecz et al., 2013) where the prior distributions on model parameters have, in turn, hyperprior distributions on those distribution hyperparameters. The benefits of hierarchical modeling are that they tend to better fit the observed data and can provide interpretable (sub)population-level parameters. We follow in the newer tradition of Bayesian modeling for CCT models. And, in particular, we utilize Markov Chain Monte Carlo (MCMC) methods to estimate posterior distributions of the model parameters and hyperparameters, given some collected data. Likewise, we utilize the same general-purpose MCMC Bayesian estimation tools used above, specifically Just Another Gibbs Sampler (JAGS) (Plummer, 2003) that is programmed via the R Statistical Programming environment and related software packages of rjags, R2jags, and coda (R Core Team, 2013; Plummer, 2014; Su and Yajima, 2015; Plummer et al., 2006).

Certain issues have been raised with respect to CCT in general, and with Pairwise CCT specifically. Butts (2003) and Batchelder (2009) (or Batchelder et al. (2010)) call for constraints to be placed on the consensus digraphs, for example. And research suggests that experts are not independent in their responses within the course of a survey. Another point is classification models are better suited to certain pairwise elicitations than models that focus only on ties.

One concern with the pairwise elicitation is that respondents may not respond to each question (tie) independently of the others. Some of the models discussed here retain that property and others in Chapter 2 extend this. For the former models, we discuss model checks that may detect problems with independence (e.g. excessive transitivity) and a survey design that may discourage this behavior when obtaining the data.

The primary purpose of this paper is to advance understanding, algorithm development, and application of CCT as applied to ties on undirected networks. This paper introduces new datasets for validating such models and introduces a survey design method to assist in maintaining an assumption of certain CCT models (conditional independence of responses).

New in this paper are:

- The application of Bayesian hierarchical CCT to the pooling of graphs.
- Additional exploration of CCT on undirected graphs.
- Providing a node-based reduction of parameters for Pairwise CCT that is different from Batchelder et al. (1997). Due to the lack of directed arcs, we must no longer make an assumption of in-degree or out-degree homogeneity for subjects in modeling their responses to nodes. Since we can't do in-degree and out-degree homogeneity (Batchelder et al., 1997), we will resort to a decomposition of the Rasch difficulty parameter (β_{jk}) into a tie difficulty that is a sum of two nodal difficulties.
- A survey design to handle a potential hazard of applying models with assumptions of conditional independence to data coming from elicitations that could very well produce dependence.
- Introduction of datasets for validating undirected graph aggregation methods.
- Accommodating the discrete constraints of a forced partition in CCT consensus graph aggregation, and implementing these using novel MCMC methods.

The remainder of the document is structured as follows. First we establish some notation regarding graphs and structures used in the models described in this paper. Chapter 2 then presents a variant of the GCM, framed as a Paired Difficulty GCM, along with some initial comparisons against the GCM on validation data. This will serve as an introduction to the CCT model on which we will base other models. Chapter 3 adds both a prior hard constraint on the consensus and adds a coordination layer to address conditional independence assumptions in pairwise elicitation. Finally, we conclude with general discussion, further steps to undertake, and ideas for future research.

1.2 Notation

In order to continue, it will be useful to provide notation and terminology for graphs and to discuss some properties of graphs. Graphs will be used in multiple ways in this paper. Graphs are both the form of the response data collected from informants and the consensus object that is the target of the aggregation.

1.2.1 Graph notation

Formally, graph structures are represented as finite sets of elements. Typically, a *graph* is a pair consisting of a set of nodes and a set of ties. Let $G = (V, E)$, where V is a finite set of nodes, with $|V| = N$ and E is a finite set of pairs of distinct nodes from V. For *digraphs* (directed graphs), E is a set of ordered pairs of nodes, (j, k), called *arcs.* Whereas for graphs (i.e. *undirected* graphs), E is a set of unordered pairs of nodes, $\{j, k\}$, called ties.[2] In this paper we only consider *simple* graphs and digraphs, where nodes in pairs must be distinct. Digraphs are called *symmetric* if, for every distinct pair of nodes, $(j, k) \epsilon E$ iff $(k, j) \epsilon E$. We may consider a *subgraph* generated by a subset of V as $G' = (E', V')$, where V' is the chosen subset of nodes and $E' = \{j, k\} \epsilon E | j, k \epsilon V'$.

Graphs and digraphs may have a limited number of ties and arcs, respectively. A graph (digraph) is *complete* iff E contains a tie (arc) for each possible undirected (directed) pair of distinct nodes in V.

Since the focus of this paper is on graphs, most of the following terminology and properties are defined in terms of graphs. Excepting some mathematical complication, extending these

[2] Graphs are alternatively called *networks*, depending on the field of study and author. Here, we reserve the term *network* to generally reference graphical structures including graphs and digraphs. Arcs are sometimes known as *arrows* and ties are alternately known as *edges* or *lines.* Nodes are alternatively called *vertices* or *points* or *actors.*

to digraphs should be intuitive and the notation translatable. When we want to reference the notion of either undirected or directed graphs, we will refer to them as *networks*.

Any two nodes that have a tie in E are called *adjacent* nodes. Likewise, any two ties that share a common node are called adjacent ties. That is, for $i, j, k \epsilon V$, if $\{i, j\}, \{j, k\} \epsilon E$, then ties $\{i, j\}$ and $\{j, k\}$ are adjacent via node j. A *path* is a sequence of distinct alternating nodes and ties, beginning and ending with a node. A (simple) *cycle* is a path with distinct nodes except the first and last nodes, which must be identical. The number of ties in a path or cycle is its length.

Two nodes are considered *connected* in a graph if there exists a path between them. A *component* of a graph is a set of nodes (and the associated ties) for which all nodes within that set are connected and for which every node not in that set is not connected with any node in that set. A graph may then have from 1 to N components.

It is sometimes useful to associate each tie with a value (besides the presence or absence in E) when modeling graphs. We call such graphs *valued graphs*. Valued graphs still have a tie set E, which may or may not be complete, but also have a tie function, σ, that maps ties into a value set. Let $\Sigma = (V, E, \sigma)$ be a *signed graph*, which is a valued graph where V and E are as before and where $\sigma : E \rightarrow \{0, 1\}$. The '0' and '1' advisably stand for real-world ties that are intrinsically "negative" and "positive", respectively.[3] Indeed, a complete binary valued graph represents a an undirected graph if we let each tie's binary value act as an indicator for whether $\{j, k\} \epsilon E$ in the corresponding unvalued graph.

A subgraph generated by three nodes is called a *triad*. Thus, in complete graphs, every combination of three nodes is a triad of three adjacent nodes and three adjacent ties. In signed complete graphs, triads have tie values in one of four configurations (discounting rotations): $\{1, 1, 1\}$, $\{1, 1, 0\}$, $\{1, 0, 0\}$, and $\{0, 0, 0\}$.

[3] A more intuitive coding would let $\sigma : E \rightarrow \{-1, 1\}$. However, in this dissertation some of the math is easier to read using the $\{0, 1\}$ coding. The choice is mathematically arbitrary.

With a graph of N nodes, there are up to $\binom{N}{2} = N(N-1)/2$ possible ties (and exactly that many in complete graphs). Therefore, without any constraints on features or properties, there are $\binom{N}{2}$ possible graphs and an equivalent number of complete signed graphs.

1.2.2 Observed graphs and consensus graph

As with other CCT papers, this paper discusses aggregation methods for questionnaires consisting consisting of a series of question items that are issued to each informant, resulting in a single consensus.

In the Pairwise CCT models to be discussed, such question items are on the dichotomous nature of ties among nodes, with the major goal of aggregating disparate graphs from error-prone experts. We concentrate on questions framed as judgments of the tie as a whole rather than a judgment on the comparison of the two nodes.

We will aggregate judgments on all ties in a signed complete graph. To apply these Pairwise CCT models, we require judgments from $M > 1$ experts on the (dichotomous) values of signed ties between all pairs from a fixed set V consisting of N nodes, though we discuss relaxing the requirement of completeness in a later section.

Let the random variable $X_{i,jk} \epsilon \{0,1\}$ denote the signed response of expert i to tie $\{j,k\}$, $1 \leq j < k \leq N$, where

$$X_{i,jk} = \begin{cases} 1 & \text{if expert } i \text{ reports tie } \{j,k\} \text{ is positive ('+')}, \\ 0 & \text{if expert } i \text{ reports tie } \{j,k\} \text{ is negative ('}-\text{')}, \\ \text{NA} & \text{if there is missing data for expert } i \text{ on tie } \{j,k\} \ . \end{cases} \tag{1.1}$$

Then $\mathbf{X}_i = \{X_{i,jk} | 1 \leq j < k \leq N\}$ is a signed complete graph that represents the responses of expert i, called their *response graph.* $\mathbf{X} = \{\mathbf{X}_i | 1 \leq i \leq M\}$ is the collection of response graphs from all the experts. This data structure is the undirected analogue to the "three-way" response profile or list of undirected "cognitive social structures" analyzed by Batchelder et al. (1997) and Krackhardt (1987).

After aggregation of any kind, decision-makers, analysts, anthropologists, etc. typically desire a single consensus graph. We shall call this the *aggregate* signed complete graph $\mathbf{Z} = \{Z_{jk} | 1 \leq j < k \leq N\}$, where

$$Z_{jk} = \begin{cases} 1 & \text{if the consensus graph tie } \{j,k\} \text{ is positive ('+')}, \\ 0 & \text{if the consensus graph tie } \{j,k\} \text{ is negative ('}-\text{')} \ . \end{cases} \tag{1.2}$$

Chapter 2

Paired difficulty model for aggregating response graphs

2.1 Introduction

This chapter reviews a hierarchical Bayesian model implementation of the General Condorcet Model (GCM) variant of CCT and then proposes an extension called the Paired Difficulty model (PDGCM) to address its application to graph data. CCT is advanced by this model because it applies the framework to undirected graphs, which have not been addressed directly before.

2.2 GCM/Tie Model

In this model, we collect all M experts' complete signed response graphs on N nodes ($\binom{N}{2}$ questions per expert) in $\mathbf{X}$, as in (1.1) above. As with other CCT models, we presume that there is a latent ground truth consensus that experts have been exposed to prior to question-

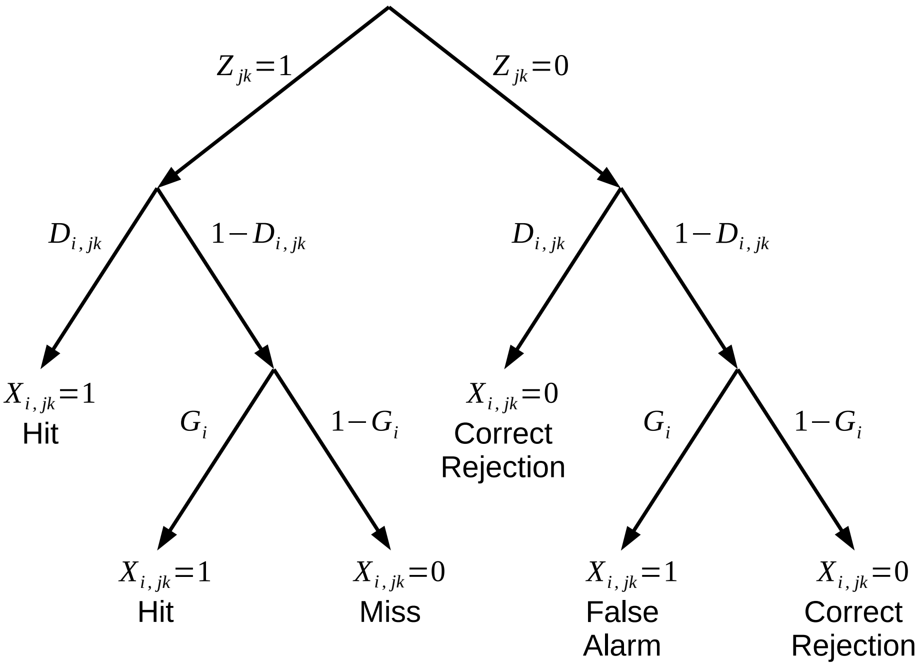


Figure 2.1: Multinomial processing tree for the response model portion of the Tie model, depicting 2HT response in Axiom 2.3.

ing. The response model assumes that each tie judgment by each expert is either detected directly from the consensus graph or it is a biased guess. This assumption is typical of other CCT models, and it is based on the *two high threshold* model (2HT) of signal detection theory (Macmillan and Creelman, 2004). The basic idea of the 2HT assumption is that detection thresholds for positive and negative ties are both sufficiently high so that detections only lead to the consensus tie; however responses that depart from the consensus graph may occur when detection fails and a guess is made. This model follows the GCM (e.g. Karabatsos and Batchelder, 2003) response model closely, where the model specifies parameters for expert detection probabilities and guessing probabilities as follows. Let $D_{i,jk} \epsilon [0, 1]$ be the probability that expert i detects the sign of tie $\{j, k\}$, and $G_i \epsilon [0, 1]$ the probability expert i guesses '+' to any tie that they do not detect. We collect these expert parameters into arrays $\mathbf{D} = \langle D_{i,jk} \rangle$ and $\mathbf{G} = \langle G_i \rangle$. That is, experts have latent expert-item response ability and latent guessing bias that apply to their responses to each tie. The complete response model is stated below in four axioms.

2.2.1 Specification of the GCM/Tie Model

Axiom 2.1 (Common truth)**.** *All experts respond according to a shared, but unobserved, complete signed consensus graph as in* (1.2)*:* $\mathbf{Z} = \{Z_{jk} | 1 \leq k < j \leq N\}$ *on* N *nodes.* $Z_{jk} = 1$ *when edge* $\{j, k\}$ *is '+' and* $Z_{jk} = 0$ *when it is '−'.*

Axiom 2.2 (Conditional independence)**.** *The responses from all the experts are conditionally independent given the consensus graph and other model parameters; namely for all realizations* $\mathbf{x}$ *of* $\mathbf{X}$*,*

$$\Pr(\mathbf{X} = \mathbf{x} | \mathbf{Z}, \mathbf{D}, \mathbf{G}) = \prod_{i=1}^{M} \prod_{j=2}^{N} \prod_{k=1}^{j-1} \Pr(X_{i,jk} = x_{i,jk} | Z_{jk}, D_{i,jk}, G_i) \ . \tag{2.1}$$

The component response probabilities above are conditional on two response parameters and the consensus tie value. Under a signal detection theory (SDT) framework, we characterize an expert's correct response to a $Z_{jk} = 1$ tie (consensus value is "positive") as that of detecting a "signal" and a correct response to a $Z_{jk} = 0$ tie (consensus value is "negative") as that of a correct rejection. In an SDT parameterization, we would characterize the expert-item response probability with a *hit rate* and *false alarm rate.* We follow the GCM cognitive model used in Karabatsos and Batchelder (2003) and in Batchelder and Romney (1988) by using a reparameterization of hit rates and false alarm rates. In 2HT, hit rates and false alarm rates are reparameterized as an expert-item competence $D_{i,jk}$ and expert guessing bias G_i used above. Informally, we can read this response model as: The expert correctly responds with the consensus value (Z_{jk}) with probability $D_{i,jk}$ or, failing that, guesses "positive" with probability G_i. The reparameterization is given by

$$H_{i,jk} = D_{i,jk} + (1 - D_{i,jk})G_i, \tag{2.2}$$

$$F_{i,jk} = (1 - D_{i,jk})G_i. \tag{2.3}$$

The current parameterization requires that the hit rate be greater than the false alarm rate, $H_{i,jk} \geq F_{i,jk}$. One consequence of this is that the model does not account for experts who may purposefully (or perversely) respond opposite to the encoding used in the survey. That is, we do not account for experts who would respond "negative" to a tie they recognized as a "positive" tie in the consensus (and vice-versa).

Axiom 2.3 (Marginal tie responses)**.** *The component response probabilities above* (2.1) *are given by*

$$\begin{aligned} &\Pr(X_{i,jk} = 0 | Z_{jk}, D_{i,jk}, G_i) = \\ &\qquad [(1 - D_{i,jk})(1 - G_i)]^{Z_{jk}} \, [D_{i,jk} + (1 - D_{i,jk})(1 - G_i)]^{(1-Z_{jk})} \quad , \end{aligned} \tag{2.4}$$

$$\begin{aligned} &\Pr(X_{i,jk} = 1 | Z_{jk}, D_{i,jk}, G_i) = \\ &\qquad [D_{i,jk} + (1 - D_{i,jk})G_i]^{Z_{jk}} \, [(1 - D_{i,jk})G_i]^{(1-Z_{jk})} \quad . \end{aligned} \tag{2.5}$$

Eq. (2.4) gives the probability the expert responds "negative" under the two possible latent consensus values, with the multiplicands chosen by the value of their respective exponents (determined by Z_{jk}). We can interpret the the first multiplicand as the probability the expert does not know the consensus value and guesses negative when the consensus value is really negative. The second multiplicand is the probability that the expert knows the consensus value *or* does not know the value and guesses negative when the consensus value is really positive. Eq. (2.5) gives the probability the expert responds "positive". The the first multiplicand is the probability the expert knows the consensus value *or* does not know the consensus and guesses positive when the consensus value is really positive. The second multiplicand is the probability the expert does not know the consensus value and guesses positive when the consensus value is really negative. Together, the axioms through 2.3 comprise the Multinomial Processing Tree (MPT, e.g. Smith and Batchelder, 2010) visualized in Figure 2.1.

Axiom 2.4 (Rasch knowledge)**.** *Each expert has an ability, $\alpha_i \in \Re$, and each tie has its own difficulty, $\beta_{jk} \in \Re$, such that*

$$D_{i,jk} = \frac{1}{1 + exp(-(\alpha_i - \beta_{jk}))} \; . \tag{2.6}$$

This reduces of the number of expert-item parameters from $M \times \binom{N}{2}$ to $M + \binom{N}{2}$ and is a typical modeling step in CCT and Item Response Theory (IRT) (Fischer and Molenaar, 1995). The result is that the logit expert-item competence $D_{i,jk}$ of expert i knowing the consensus value of tie $\{j, k\}$ is a linear function of the expert's ability and the tie's difficulty. Therefore, we may construe the expert-item competence as a sort of competition between the expert and the question item where only their *relative* difference matters.

2.2.2 Likelihood of observed data

Individual responses are assumed conditionally independent within and between expert response graphs, a common (albeit not necessarily very realistic) assumption made in CCT literature (e.g. Batchelder et al., 1997; Butts, 2003). With this assumption, the joint likelihood is given as a simple product of the individual tie response likelihoods, as given in in (2.1).

We will use the Likelihood of the model for performing Bayesian inference on the model parameters. Combining (2.1), (2.4), and (2.5), we obtain the following joint likelihood for the GCM/Tie Model:

$$
\begin{aligned}
&\Pr(\mathbf{X}=\mathbf{x}|\mathbf{Z},\mathbf{D},\mathbf{G}) = \\
&\quad \prod_{i=1}^{M}\prod_{j=2}^{N}\prod_{k=1}^{j-1} \\
&\quad \Big([(1-D_{i,jk})(1-G_i)]^{Z_{jk}}\,[D_{i,jk}+(1-D_{i,jk})(1-G_i)]^{(1-Z_{jk})}\Big)^{1-x_{i,jk}} \\
&\quad \Big([D_{i,jk}+(1-D_{i,jk})G_i]^{Z_{jk}}\,[(1-D_{i,jk})G_i]^{(1-Z_{jk})}\Big)^{x_{i,jk}} \quad .
\end{aligned}
\tag{2.7}
$$

Above, the exponents $1 - x_{i,jk}$ and $x_{i,jk}$ act as mutually exclusive (and exhaustive) indicators, indexed on the $x_{i,jk}$. Additionally, the GCM/Tie Model can be used with missing data by letting the probability of the missing datum, given the parameters, to be unit, which does not contribute any meaningful change to the overall likelihood.

2.3 Bayesian specification of the GCM/Tie Model

Characterizing the GCM/Tie Model as a Bayesian graphical model requires specification of prior distributions over the model parameters and hyperparameters, and they are generally mutable in these models as desired.

We use a *mixed hierarchical* model, where certain parameters' distributions are drawn from a hyperdistribution of fixed effects. A similar form of this model was used in Agrawal and Batchelder (2012). In contrast, this model specifies hierarchical Bayesian prior distributions and distributions on the consensus that assumes a priori dependence between consensus tie values. Gelman et al. (2004) and Jackman (2009) provide good backgrounds on hierarchical Bayesian modeling.

One of the benefits of the hierarchical approach is better modeling of differences in groups of parameters and a broader application of models. Here we take advantage of this by parameterizing an overall mean difference between the expert abilities and those of the tie difficulties. The inclusion of the hierarchical modeling parameters —as opposed to a fixed effects model— has a minimal cost in terms of additional parameters since each block of exchangeable parameters could add as few as one or two extra parameters, depending on modeling decisions (Lee, 2011).

To proceed, we first note the response model described thus far using standard Bayesian Graphical Model distribution notation. This is followed by specification of additional priors needed to complete the model. In general, these distributions are chosen to reflect a researcher's prior understanding of parameters involved. When there is a lack of prior knowledge, one arguably leans towards using a so-called "non-informative", or "flat", prior which reflects the researcher's uncertainty in the distribution of the parameter's values (e.g. all values are equally probable).

2.3.1 Response model

The response model of Axioms 2.2 and 2.3 may be rewritten in distribution notation without changing or adding to the model. Recall (2.5) as the probability that a given response is positive (i.e. $x_{i,jk} = 1$). These may be written as

$$\text{Response:} \qquad X_{i,jk} \sim \text{Bernoulli}(Z_{jk}D_{i,jk} + (1 - D_{i,jk})G_i) \ . \tag{2.8}$$

The Rasch equation (2.6) of Axiom 2.4 remains as written since it is not a random variable.

2.3.2 Expert bias, tie difficulty, and expert ability prior distributions and hyperdistributions

Expert guessing bias is set to match Karabatsos and Batchelder (2003), where

$$\text{Expert guessing bias:} \qquad G_i \sim \text{Uniform}(0,1)\ . \tag{2.9}$$

Following Batchelder and Anders (2012), we note that prior research suggests that guessing biases tend to be estimated near 1/2 and we infrequently find real *informative* experts providing extreme tendencies in overall response (i.e. responding all ‘positive’ or all ‘negative’) or extreme estimated biases. Indeed, simulations studies confirm that the generating bias and the observed tendency overall are correlated (as expected).

The group of tie difficulties are drawn from a Normal distribution centered on 0, and with a standard deviation that has its own distribution,

$$\text{Individual tie's difficulty:} \qquad \beta_{jk} \sim \text{Normal}(0,\sigma_\beta)\ . \tag{2.10}$$

Each tie's difficulty is therefore drawn from a population-level mean of 0, but with a latent (variable) scale. The zero-centered difficulty identifies the model since the Rasch function (2.6) does not vary when the same value is added to both the ability and difficulty parameters. As different data may suggest different variation among the ties' difficulties, we will allow the scale parameter to float as a random variable with its own hyperdistribution:

$$\text{Scale of tie difficulties:} \qquad \sigma_\beta \sim \text{Uniform}(0, \kappa_{\sigma_\beta}) \ , \tag{2.11}$$

$$\text{Tie difficulty scale upper limit:} \qquad \kappa_{\sigma_\beta} = 3 \ . \tag{2.12}$$

This choice of a uniform distribution over the scale of the Normally-distributed difficulty is borrowed from Jackman (2009); the uniform is useful because it is straightforward to interpret and easy to set hyperparamaters for. The specific value of $\kappa_{\sigma_\beta} = 3$ was chosen as a reasonable upper limit to observable deviation around the mean for the difficulties, given the limited size of data discussed in this paper. Informal testing has convinced the author that more extreme upper bounds on this non-informative prior results in longer sampling runs needed for convergence of the estimations with no appreciable qualitative difference in the results. In simulation studies and in real application of the model, the researcher will want to take note of high posterior density on near the upper limit of the scale parameters' priors; such a phenomenon indicates that it should be raised.

The choice of expert ability prior and hyperprior distributions is analogous to the tie difficulties in that they have complementary influence on the resulting probability of expert-item competence in the Rasch model (2.6). However, unlike the tie difficulties, which are centered at 0, expert ability is drawn from a Normal distribution centered at some arbitrary mean and with a standard deviation independent of that used in the tie difficulties,

$$\text{Location of expert abilities:} \qquad \mu_\alpha \sim \text{Normal}(0, 2) \ , \tag{2.13}$$

$$\text{Individual expert's ability:} \qquad \alpha_i \sim \text{Normal}(\mu_\alpha, \sigma_\alpha) \ . \tag{2.14}$$

The location of the mean of expert abilities can float around. With the sizes of the data used in the testing of this model, and those used in previous CCT literature, we do not need to concern ourselves with extreme differences between abilities and difficulties, and so resort to a limited fixed scale on the population distribution of abilities.

We model the expert ability scale parameter hyperdistribution in the same manner as that used for difficulties. Thus,

$$\begin{aligned} &\text{Scale of expert abilities:} && \sigma_\alpha \sim \text{Uniform}(0, \kappa_{\sigma_\alpha}) \ , && (2.15) \\ &\text{Expert ability scale upper limit:} && \kappa_{\sigma_\alpha} = 3 \ . && (2.16) \end{aligned}$$

While different from these other models, the priors and hyperdistributions chosen here are easy to interpret and work with and are applicable to a wide range of populations.

2.3.3 Consensus tie priors

This model assumes a prior on consensus tie values as conditionally independent, and therefore the whole consensus may be factored as a product of their respective conditional (tie) prior distributions, where each tie of the graph independently has the same probability of being 1 (or 0). The expectation of the density of positive ties in the graph is a hyperparameter in $[0, 1]$. With conditionally independent ties, we have

$$\Pr(\mathbf{Z} = \mathbf{z}) = \prod_{j=2}^{N} \prod_{k=1}^{j-1} \Pr(Z_{jk} = z_{jk}) \ . \tag{2.17}$$

This is a standard CCT assumption, where each consensus item Z_{jk} is modeled as a Bernoulli trial with some expected base rate of being one answer or another. We model this with a hierarchical uniform prior distribution on the single probability,

$$\text{Consensus graph positive density:} \qquad p_Z \sim \text{Uniform}(0,1)\ , \tag{2.18}$$

$$\text{Consensus tie value:} \qquad Z_{jk} \sim \text{Bernoulli}(p_Z)\ . \tag{2.19}$$

The choice of a uniform distribution on p_Z is typical (e.g. Batchelder and Anders, 2012), though other modelers may decide to put a hierarchical distribution on p_Z in the presence of more information. In the present application, we assume lack of knowledge about the overall expected density of positive ties in the consensus graph. This implies that each configuration of consensus tie values is equally likely a priori.

2.4 Properties of the GCM/Tie Model

Axiom 2.1 maintains the single-culture assumption of the GCM and many other CCT models. This states that there is a single complete signed graph that represents the consensus knowledge of all the experts. Because of this assumption, one would expect that there would be a lot of dependencies between the responses from the experts even though they respond without collaboration. Axiom 2.2 assures that these dependencies disappear when the responses are conditioned on the consensus partition and the other parameters. A conditional independence assumption like (2.1) is typical of response models in psychometric test theory (Fischer and Molenaar, 1995) as well as many other parametric models. Axiom 2.3 expresses the marginal response probabilities defined in (2.1) as a function of the relevant parameters consisting of four exponentiated terms as (2.4) and (2.5). The terms correspond, respectively, to the usual signal detection terms *misses* and *correct rejections* in (2.4) and *hits* and *false alarms* in (2.5). For example if $Z_{jk} = 1$ and $x_{i,jk} = 1$, then the tie is positive and the expert correctly responds positive. This is a hit and it occurs in the 2HT model if the expert detects the positive tie, with probability $D_{i,jk}$, or fails to detect it and guesses 'positive', with probability $(1 - D_{i,jk})G_i$, as reflected in the first term in (2.5).

Axiom 2.3 provides knowledge and detection parameters for every combination of expert and tie, so some further restriction in parameters is necessary to make the model workable. Axiom 2.4 addresses this problem by applying a standard psychometric modeling idea in (2.6) to the $D_{i,jk}$ called the Rasch model (Fischer and Molenaar, 1995). The Rasch model is also applied to the GCM when items and experts are modeled with heterogeneous difficulty. In the GCM, as in test theory, the Rasch model is applied to the probability that expert i is correct to item j; however, in this case we apply it to a latent parameter $D_{i,jk}$ that is indexed by experts i and ties $\{j, k\}$. In essence the Rasch model implies that the effects of expert ability and tie difficulty on the probability of detecting the signal are additive as can be seen by computing

$$logit(D_{i,jk}) = log\left[D_{i,jk}/(1 - D_{i,jk})\right] = \alpha_i - \beta_{jk} \ . \tag{2.20}$$

It is well known that the Rasch model is non-identified as one can add a positive constant to both the abilities and difficulties in (2.20) without changing the value of $D_{i,jk}$. A common method of identifying the model with the Rasch component is to anchor one of the difficulty or ability parameters to some fixed point against which all others are interpreted as relative, an approach used in Karabatsos and Batchelder (2003). Another method is used by Johnson and Albert (1999) which identifies a model by way of a prudent choice of prior. Improper priors that are "flat" across the real line would leave the model non-identified, but the choice of Normal priors with limited scales on the ability and difficulty parameters keeps their posterior location stable on the real line, while allowing the data to shift them relative to one other.

The rather simple arrangement of consensus prior for **Z** (3.18 & 3.19) implies a *consensus model* that simply does not specify any dependence between any two items consensus values. This is in contrast the the PDGCM discussed later that imposes a constraint.

The model comprised of Axioms 2.1 - 2.4 differ from the General Condorcet Model (GCM) in Batchelder and Romney (1988), Karabatsos and Batchelder (2003), and Batchelder and Anders (2012) in two important ways. First, the GCM items are N separate dichotomous true/false questions rather than dichotomous questions about the ties between nodes in a graph. Second, the hierarchical modeling choices are new and different compared to some other recent hierarchical Bayesian GCM implementations.

2.5 Implementation and estimation

The models discussed in this paper are all posed as Bayesian Graphical Models (e.g. Lee, 2014). Posterior distributions for model parameters (i.e. parameters, given observed data) can theoretically be estimated using numeric sampling algorithms. In practice, not all Bayesian Graphical Models can be estimated well given practical finite limits on time, computing power, as well as peculiarities of certain models.

Fortunately, this paper's models resemble the hierarchical GCM (Oravecz et al., 2013) for which numeric estimation algorithms have worked well. The general-purpose numeric Bayesian modeling software family that includes BUGS, WinBUGS, OpenBUGS, and JAGS has successfully been used to perform analyses on various data on cultural beliefs (e.g. Oravecz et al., 2013, 2014). The models discussed in this paper are all implemented using the Just Another Gibbs Sampler (JAGS) software (e.g. Plummer, 2003) accessed from the R programing environment through the R package `R2jags` and `rjags` (R Core Team, 2013; Su and Yajima, 2015; Plummer, 2014). The appendix includes the exact JAGS code used in this paper.

Some nice benefits of the numeric sampling methods, including the JAGS implementation used here, are that (a) closed form solutions to posterior distributions are not necessary, (b) models are relatively easy to modify and adapt to different situations, (c) posterior-predictive data (and accompanying statistics) can be easily drawn given the samples from the joint posterior distribution of the model parameters, and (d) missing data are handled automatically.

Notably different from the hierarchical GCM described in Batchelder and Anders (2012) are that expert biases are not hierarchical and we allow for item and expert heterogeneity. The G_i parameters in the Tie Model are modeled with a (flat) Beta prior such as has been used in the GCM described in Karabatsos and Batchelder (2003) and in Butts (2003). Using the Rasch model for combining item difficulty and expert ability parameters to compute a per-response competence handles data where items and/or experts cannot be assumed to be relatively homogeneous. These parameters are in the real line and are hierarchically modeled following Jackman (2009) (e.g.) with respect to population-level location and scale parameters.

Since we utilize a slightly different variant of the hierarchical GCM to form the Tie Model, we perform simulation studies to ensure the code and models are working as intended.

2.6 Data acquisition and survey design

Our data collection requirements are different than those typically used in social network elicitation for the purpose of network reconstruction. These models use a complete network elicited from each of several experts, which distinguishes these datasets from the common snapshot ("$n = 1$") networks used frequently in social science research (e.g. Add Health, Bearman et al., 2004). The datasets we require are similar to the cognitive social structures (CSS) used by Krackhardt (1987) and Batchelder et al. (1997). In those papers, the data collected comprised of response digraphs, with one digraph from each expert. Eliciting digraph (or graph) data from experts is tedious for them in that there are many responses required (recall the $O(N^2)$ number of arcs). To ease the survey designer and the expert respondents, a check-off form is sometimes used for digraph elicitation where a single sheet is used for each node. For a given sheet, the expert checks off their judgment about that node's directed relationships with each other node listed, completing such a task for each sheet of an N page booklet. When eliciting undirected graphs from experts, we might be tempted to use symmetrized digraph data collected in the manner just mentioned. Or, one might also choose a different method, such as described here.

While this Tie Model is structurally equivalent to the GCM used elsewhere, the question items (ties) for which this model is presently applied threaten to break core assumptions of conditional independence of responses. In the GCM, the question items are from the same topical domain, but such that the consensus answer to one item has no impact on the consensus answer to another item. Likewise, the GCM/Tie Model assumes that the expert responds without connecting their responses to different items.[1]

[1] In GCM data collections, the question orders are often randomized to account for order effects.

The question of breaking consensus conditional independence is broached in Butts (2003), Batchelder (2009), and Agrawal and Batchelder (2012), where constraints are imposed on the consensus network structure via the Bayesian model prior.

Regardless, experts may or may not respond to each question independently. Let us consider an example to motivate why a researcher may want to collect data in a different way than in GCM or digraph questionnaires. Randomized in order or not, the questionnaire may contain a sequence of three ties in close sequence that form a triad. Let there be nodes named *Alice*, *Bob*, and *Carol* among a set of nodes. A questionnaire may yield these in close sequential proximity (See Table 2.1). In this situation, with Q1 and Q2 on their mind and their responses for them already elicited, the model assumes that their response to Q3 will be made independent of their responses to Q1 and Q2. However, it is reasonable to suppose that either internal logic, demand characteristics, or some other motivation might drive the expert to report the Q3 response based on some heuristic that utilizes their previous responses, thereby introducing a dependency that is not handled by the current model.

Order	Question	Response
Q1	Is Alice friends with Bob?	No
Q2	Is Bob friends with Carol?	Yes
Q3	Is Carol friends with Alice?	?

Table 2.1: Example of question items forming a triad. In this example, the expert has responded “No” and “Yes” to the first to ties of a triad (Q1, Q2). The expert may base their response to the last tie in the triad (Q3) on their responses to Q1 ad Q2, each of which share a node with Q3.

The models do not build in such heuristics, and they assume that ties are either recognized independently or guessed when unrecognized (and never logically determined). Therefore, we attempt to mitigate this issue by using a survey design technique that makes such logical inferences more difficult for the respondent. We identify these likely components of such heuristics in this elicitation task:

1. Applying a heuristic to a tie response that uses responses from previous responses only comes into play when that tie is completing some cycle of ties that have been presented thus far.
2. A heuristic based on such completion of such a cycle will be more difficult for longer cycles when compared with shorter cycles.
3. Utilizing a heuristic on previous ties will be more difficult when distractors are interspersed.

We focus on these issues when preparing an algorithm for ordering ties on the questionnaire, and so the specific heuristic need not be identified. Let us consider an example heuristic that certainly applies to triad cycle completions. In completing a triad on a questionnaire, there are four possible configurations of response to the first two ties, and for each we have an "Friend/Enemy" heuristic as shown in Table 2.2. This heuristic is simple for triadic cycles, and similar such heuristics may play a role in longer cycles (consider "The enemy of a friend of a friend of a node is that node's enemy.").[2] An analogous heuristic could conceivable play a role when experts respond to questions on geographic proximity.[3]

[2] The notion of structural balance in a signed graph is discussed in Cartwright and Harary (1956) and Harary (1959), where an expert may tend towards a perception of a network so that nodes in a set can be partitioned as two subset that have only negative ties between them and positive ties within them. In such balanced networks, any cycle of nodes must have an even number of negative ties. This leads to a possible logical heuristic where an expert may attempt to maintain balance in their response network by attending to the number of negative ties in a cycle they are completing for a given question item.

[3] Suppose experts are asked whether pairs of states share a border or not. For two states, A and B, they respond "border", and for states B and C they respond "not border". Their response to {A, C} may be biased towards "not border" since if A and B are close and B and C are far, then A and C must also be far.

Resp. to Q1	Resp. to Q2	Rule	Resp. to Q3
Friends(+)	Friends(+)	Friends of friends are friends.	Friends(+)
Friends(+)	Enemies(-)	Enemies of friends are enemies.	Enemies(-)
Enemies(-)	Friends(+)	Friends of enemies are enemies.	Enemies(-)
Enemies(-)	Enemies(-)	Enemies of enemies are friends.	Friends(+)

Table 2.2: Friend/Enemy heuristic example for responses to triad completion in social network elicitation.

Our approach to the survey design question does not really need to consider which heuristic is being used, so long as the heuristic relies on completing cycles of ties that have been previously elicited. We have a design that uses three assumptions on such heuristics that are based on cycle completion.

The design is for a complete graph, so all ties are asked. $N - 1$ ties are first presented, for which all nodes are addressed, but no logical cycles are completed. Ties that begin completing the longest possible cycles are presented next, starting with the cycle of length N. There is only one newly-completed cycle of length N, given that the $N - 1$ ties have been asked that are a path on all the nodes. This one remaining tie that completes the sole length N cycle connects one node to the other. See "Tie 6" in Figure 2.2 for an illustration of this N'th tie.

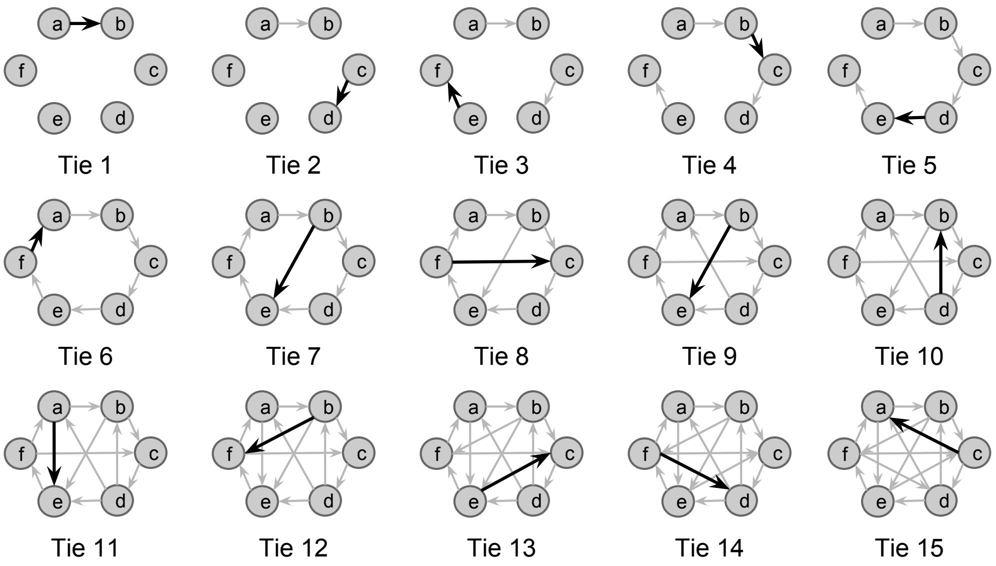


Figure 2.2: An example of survey design presenting 15 ties (questions) for the $N = 6$ nodes.

The algorithm operates as follows.[4] Each run of this algorithm produces a survey design for one expert. Multiple runs of the algorithm are required for generating randomized designs for all experts.

1. N is given, fixed.
2. Create an empty ordered list for retaining arcs to be presented.[5]
3. Create a list of all $N(N-1)$ arcs that also stores attributes about each arc with respect to the arcs presented thus far. Attributes include: the minimum cycle length of all cycles that would be created if the arc were added to the presentation list; the number of cycles that would be completed if the arc were added to the presentation list; degrees of the nodes involved.

[4] Interested readers may request R code for this algorithm from the author.

[5] Note that the questions are generally inherently directed when actually presented.

4. For each arc in the full arc list, update the attributes according to the current state of the presentation arcs list.
5. Create a subset of the arcs list that has the following properties: arcs that have not been presented already, arcs that have the maximum minimum-size cycle that would be created if added, arcs that minimally unbalance the in/out degree distributions of the nodes.
6. Choose an arc from the subset to add to the presentation list.
7. **If** the presentation list is not complete (i.e. if it does not have $\binom{N}{2}$ arcs with an arc for each tie the graph of N nodes) *and* the subset list of arcs is empty,
    - **then** we reject this run of the algorithm, and restart the algorithm under different random number generator starting state.
    - **Else**, of the arcs in the subset, choose a random arc with uniform probability and add it to the presentation arcs list.
8. **If** the presentation arcs list is not complete,
    - **then** go to step 4.
    - **Else**, present the list of presentation arcs.

The benefits of using such an algorithm is useful because (a) we mitigate the threat on conditional independence on the expert's responses with respect, and (b) we may further enhance this mitigation by selectively omitting data towards the end of the questionnaires to create an incomplete design.

2.7 Simulation studies for the GCM/Tie Model

Before even applying estimation algorithms towards real data, it is important (a) to validate that the inference engine (i.e. estimation algorithm) is running as expected for controlled inputs and (b) that data predicted by the posterior are similar to input data. In this section, we first review simulation studies and insights provided by them. In the next section, we apply the model to real data to validate the procedure.

MCMC estimation methods for the GCM have been implemented under various incarnations (e.g. Karabatsos and Batchelder, 2003; Oravecz et al., 2013). Some of these (e.g. Oravecz et al., 2013) use the same general-purpose MCMC Bayesian estimation software employed here. However, here we utilize the logistic parameterization of the Rasch parameters of expert ability and tie difficulty and so we undertook some basic simulation studies to verify that the machinery is operating as expected before applying it to real data.

1. Simulating data from the model and hyperparameters should provide data that passes model checks. This ensures the priors and response model are reasonable.
2. Apply the GCM/Tie Model estimation algorithm to a handful of particular datasets with known properties to verify that the posterior parameter values behave as we expect.
3. Check that the estimation software is generally good at recovering the parameter values used to generate the data by simulating multiple datasets with known generative parameters and estimate those parameter values using the algorithm.
4. For a few chosen datasets generated using some random parameters (ability, bias, difficulty, but not answer key), obtain parameter samples from the joint posterior distribution of the parameters. For each of these sampled parameters, we generate post-predictive data and compare that post-predictive data (and statistics of the data) with the original simulated data (and statistics). The post-predictive data should "look similar" to the input datasets. Specifically, statistics of the original data should fall within the highest-probable value ranges of the distributions of corresponding posterior-predictive statistics.

Overall signal in the experts' collective responses can be measured by the mean expert-item competence

$$\bar{\mathbf{D}} = \frac{2}{MN(N-1)} \sum_{i=1}^{M} \sum_{j=1}^{N-1} \sum_{k>j}^{N} D_{i,jk} \ . \tag{2.21}$$

As with other parameters, the overall signal can be discussed as a *generating* value (as when used to simulate data) and as an *estimated* value (as the posterior mean of estimated expert-item competences).

2.7.1 Looking for a one-factor structure of responses

Before running simulation studies, we will verify that the chosen priors produce data with a one-factor structure to responses by checking that the simulated data retains characteristics of data appropriate to the model. As pointed out in Batchelder and Anders (2012), we expect a one-factor nature of the expert-by-expert correlations matrix (excluding the main diagonal where informants trivially correlate with themselves perfectly). We do this by using the fully specified hierarchical Bayesian model to randomly generate 20 sets of response graphs for each of two sizes of $\{M, N\}$. The two sizes used were { M=10, N=6 } (called `smalltest`) and { M=25, N=8 } (called `largetest`). For each of these generated data we calculate expert-by-expert correlation matrix $\mathbf{M}$ using the `fa` function in the R CRAN package `psych` (Revelle, 2014). This function implements the MINRES algorithm (Comrey, 1962, 1973) for estimation of a single-factor solution towards estimating the off-diagonal elements of $\mathbf{M}$ along with a series of eigenvalues for this solution. One interpretation for the eigenvalues is that the largest values depict the variance explained by latent factors and that the lowest values correspond to variance explained by random noise. Plotting the ordered eigenvalues is known as a Scree Plot or Scree Test, although there is no generally-accepted formal test for which eigenvalues to accept as latent factors and which to dismiss as noise (Raîche et al., 2013).

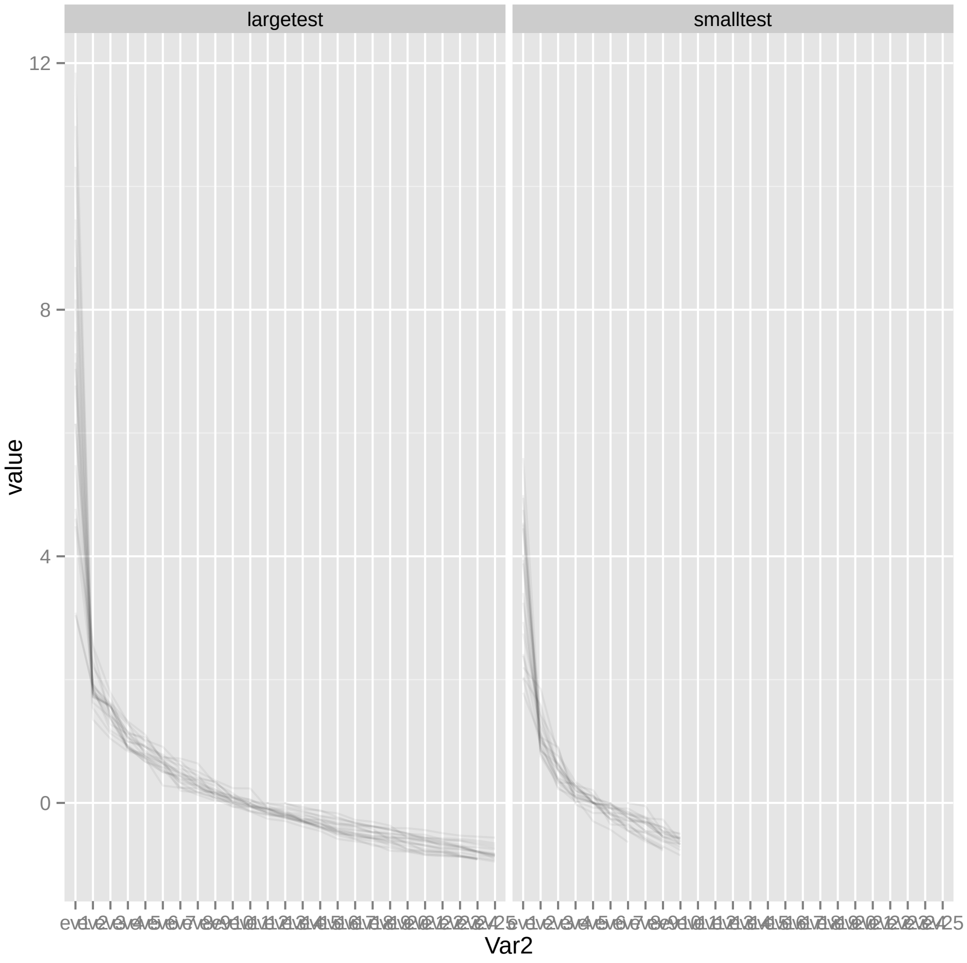


Figure 2.3: Scree clouds from simulated data from the full GCM. The consistent pattern and sharp drop-off suggests a one-factor solutions for much of the simulated data.

While one outdated rule suggests that the ratio of the first to second eigenvalue be greater than a certain threshold (e.g. Weller, 2007), there is no formal statistical test for the number of factors for the correlations matrix. However, we can explore the qualitative distribution of factor values by plotting their values. Figure 2.3 shows a resulting “cloud” of scree plots that result from the GCM/Tie Model using the specified priors and hyperpriors. We can see that the one-factor solution that the Tie Model is expected to generate is confirmed by noting that there is a tight band of expected eigenvalues under prior model assumptions.

Given that there is no statistical test for the number of factors in these data, we will later use the distribution of the ratio of first-to-second eigenvalues in postpredictive model checks.

We should note that there is a clear empirical correlation between the mean generating ability ($\bar{\mathbf{D}}$) and the resulting data's ratio of first-to-second eigenvalues ($r = 0.7$ for the `largetest` data). This corresponds with the assumption that a higher overall signal in the data should correlate with a single-factor solution to the responses. The implication is that higher signal in the data yields steeper scree plots. That is, a higher signal corresponds with a single cultural answer key for data generated according to the model, given the model as a source of data.

We conclude for our purposes that these simulated data are reasonable and proceed to test the ability of the GCM/Tie Model to recover the generating parameter values in further simulations.

Given the relatively small size of these data, we should pay attention to "pathological" cases in the simulations, especially where the discrete nature of the output data makes certain conditions occur with non-zero probability. One oddity with the current priors is that occasional datasets have a few of the experts responding with all 1s or all 0s; this did not phenomenon was only occasional in the real data as well.

2.7.2 Simulation case studies

We next look a handful of special case datasets to determine if the posterior distributions on the parameters are reasonable and expected. We started by creating data with $M = 10$

and $N = 6$ (i.e. 15 ties) that matched the following conditions that could very well occur in the field, but are not well-covered by the prior distributions:

1. Very low overall detectability.
2. Low overall detectability, i.e. overall low $\alpha_i - \beta_{jk} < 0$, with $\bar{\mathbf{D}}$ around 0.2 by design.
3. Overall $\bar{\mathbf{D}} = 0$, i.e. $\bar{\mathbf{D}}$ around zero by design.
4. Small number of experts and majority are bad., i.e. $\bar{\mathbf{D}}$ around 0.2 by design for some experts, and $\bar{\mathbf{D}}$ around 0.8 for the rest.

$\hat{R}$, a measure of convergence for numeric Bayesian estimations, values were lower after increasing the burn-in to 7000 and with 10000 iterations. Following this change, $\hat{R}$ values were less than 1.223 for all parameters. In all of these tests, $\hat{R}$ was below 1.1 for continuous parameters. Four chains were run, with 3000 retained samples per chain (following burn-in), and 12000 total samples. The generating answer key graph $\mathbf{Z}$ had the following values for all special cases: 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1 and 0.

- Case 1 ("awful signal"): Here, the data is a random-looking mess without any structure that might correlate the subjects' responses. As expected, the estimated μ_α (amu) is below zero and that all experts have low estimated α_i parameters. Indeed, only 0.02858 of ability samples are 0 or above. σ_α is within a reasonable range as well and captured by the hyperprior of Uniform(0,3). $\bar{\mathbf{D}}$ is estimated at 0.3012, which is higher than the generating $\bar{\mathbf{D}}$ for some reason.

  The estimated $\mathbf{Z}$ answer key shows lots of ambiguity (with means near 0.5 instead of the extremes) and with the point estimates for each tie not matching the generating $\mathbf{Z}$ very well. Z_{jk} are 0.037, 0.68, 0.82, 0.68, 0.73, 0.70, 0.75, 0.75, 0.66, 0.77, 0.57, 0.79, 0.75, 0.72 and 0.59. Median Z_{jk} are 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1 and 1, matching the generating answer key at rate of 0.5333 . Note that chance matching at 50% is the expectation, given the priors, when the experts have low abilities.

- Case 2 ("low signal"): The results were similarto the situation #1, but the mean ability was higher (but still negative) and the **Z** answer key showed improved resolution towards the extremes (i.e. more definitive), and fewer "errors" with respect to recovering the generating answer key. Specifically, 0.051 of ability samples are 0 or above (so, strangley there is even stronger evidence of poor overall ability than situation #1).

  Z_{jk} are 0.40, 0.89, 0.46, 0.86, 0.20, 0.85, 0.23, 0.93, 0.38, 0.90, 0.78, 0.88, 0.54, 0.92 and 0.70. Median Z_{jk} are 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1 and 1, matching the generating answer key better than situation #1 at rate of 0.8 .

  Reassuringly, the σ_α and σ_β were not pushing at the boundaries of the limited uniform distribution in the priors.

- Case 3 ("middling signal"): 0.4851 of μ_α samples are 0 or above (though μ_α was generated at 0).

  Z_{jk} are 0.040, 1.00, 0.18, 0.98, 0.037, 0.61, 0.016, 0.94, 0.014, 0.99, 0.090, 0.32, 0.082, 0.52 and 0.18. Median Z_{jk} are 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1 and 0, perfectly matching the generating answer key at a rate of 0.93. This is a good sign, given that this is the "middling" signal situation.

- Case 4 ("high and low signal"): 0.049 of μ_α samples are 0 or above.

  Z_{jk} are 0.050, 0.33, 0.18, 0.44, 0.15, 0.26, 0.14, 0.79, 0.28, 0.34, 0.39, 0.72, 0.24, 0.76 and 0.25. Median Z_{jk} are 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1 and 0, with matching rate at 0.73. This setup with widely varying abilities yielded good matching, though not perfect.

The results of the Tie Model on on these special datasets indicate that the Tie Model estimation algorithm is able to recover the latent consensus graph when there is sufficient signal in the expert's responses. That is, if subjects are guessing and the inter-expert correlations are all low, there is no way to weight the experts as knowledgeable. In such low-signal situations, we can see in equations (2.2), (2.3), (2.4) and (2.5) that the guessing bias predominates.

Given that there are limitations on the recovery ability of the latent truth when the overall signal of the experts is very low, we run a simulation study using the previous `smalltest` and `largetest` datasets. Applying the model to each of these provides latent generating values and sampled posterior values for each parameter. In particular, the mean expert-item competence in both estimation and generation indicates a strength of signal in the data (or generation). The estimate and ability to recover the generating answer key ($\mathbf{Z}$) should correlate with signal in the data. Figure 2.4 compares several of these measures for the two sizes of simulated data. The term `post_est` refers to the posterior mean of expert-item competence; `gen` refers to the generating mean competence; `n_uniqz` refers to the number of different graphs represented in the posterior distribution of $\mathbf{Z}$; `H_bits` is the number of bits of entropy in the posterior distribution of $\mathbf{Z}$, with higher entropy indicating a more evenly distributed discrete distribution of graphs; `zrecrate` indicates the rate of ties correctly categorized by the model in the posterior; `zpostconf` is the degree in (0,1) to which Z_{jk} tends towards extreme values (i.e. closer to $Z_{jk} = 0$ or $Z_{jk} = 1$). A higher `zrecrate` indicates greater posterior confidence in a tie's value, given the data.

Examination of Figure 2.4 indicates that the algorithm recovers the consensus graph perfectly in many simulations, and especially true for high-signal data (see `zrecrate` vs. `gen` or `post_est` where minimum `zrecrate` is 0.7).

Importantly, the algorithm recovers competence (indeed, individual expert ability too) quite well, noting that `gen` and `post_est` are highly correlated. Informally, a researcher expect an answer key recovery of $\mathbf{Z}$ a rate of 0.9 (for example) when the overall signal is $\bar{\mathbf{D}} > 0.5$ under `largetest` conditions.

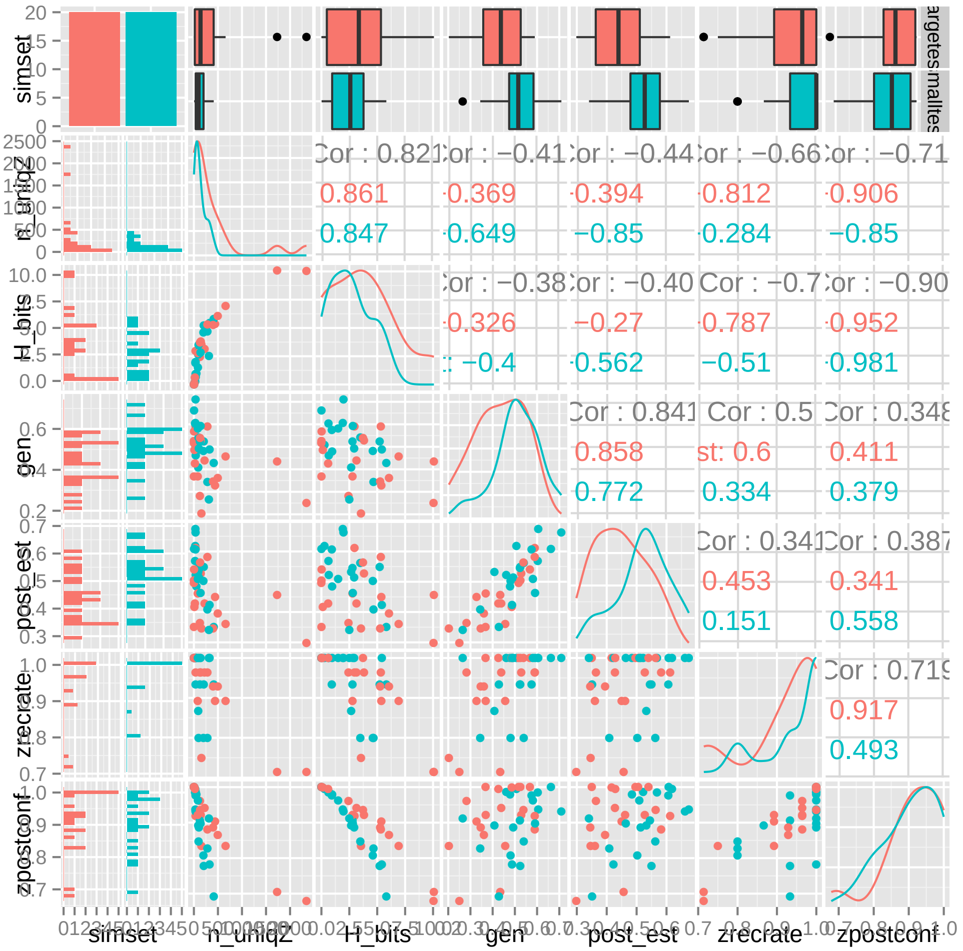


Figure 2.4: $\bar{\mathbf{D}}$ generating values (`gen`) and point estimates (`post_est`) versus the consensus recovery rate (`zrecrate`) and consensus posterior confidence (`zpostconf`) for the `smalltest` and `largetest` datasets.

Uncertainty in the posterior consensus graph should show in low-signal data as a "wide" and non-informative posterior on the distribution. For particular ties, this will mean that there is not a clear (crisp) posterior probability that a given tie is either positive or negative. This is evident in the positive correlation between competence (`gen` and `post_est`) and `zpostconf` for the `largetest`. Low signal situations should also provide a wide distribution of joint posterior consensus graphs. This joint latent answer key analysis is a point of differentiation between Tie Model analysis and GCM analysis. Note how `H_bits` (uncertainty in $\mathbf{Z}$) appropriately decreases as signal goes up.

This analysis of the signal strength does not yet give us a test of whether a given set of data has sufficient signal, but it does give some hints as to whether a given dataset should expect to provide a strong signal or not.

Several expected patterns emerged from the GCM/Tie Model simulations which gives confidence that we may apply them to real data.

2.8 Results of applying the GCM/Tie Model

Three questions are primarily important when applying CCT models to validation data.

- *How well does the posterior consensus graph match the known objective answer key graph?*

  This can only be accomplished with validation data such as the geographic knowledge surveys used here.

- *How "crisp" are the posterior consensus graph ties as well as the consensus graph as a whole?*

  Posterior tie estimates at the extremes (0 or 1) indicate a more "crisp" or confident consensus tie value. Likewise, fewer and highly concentrated consensus graphs indicate higher confidence and confirmation of the single-culture assumption.

- *Do expert and item parameters act as valid measurements of performance and difficulty?*

  We can test whether response performance against the known ground truth correlates with parameter estimates.

2.8.1 Geographic knowledge data

Four sets of geographic knowledge surveys concerning three distinct clusters of states (or nations) were issued to students at the University of California – Irvine. These provided expert judgments on whether or not states within each cluster shared a border. While state adjacencies are not social networks, they may substitute for the purposes of model validation since border statuses are widely known and easily verified.

The first three surveys were carried out in an undergraduate History of Psychology class. The three surveys each covered eight (8) states in either the Middle East (*MidEast*, $n = 64$), Central Europe (*Euro*, $n = 58$), or the Western US (*WestUS*, $n = 65$). A given survey, issued as a paper booklet, asked the expert (student) about each state's border status with every other state in the cluster. This is a standard method for obtaining directed social network relationships (e.g. Cognitive Social Structures (CSS) Krackhardt, 1987; Batchelder et al., 1997). Consequently, each expert provided a response digraph on state-to-state border status, omitting whether states border themselves. With the eight states in each region, this required each expert to respond to 56 questions. Each response digraphs was converted to a response graph by symmetrizing their responses such that a tie was coded as "yes" when the expert responded "yes" to at least one of the two response arcs corresponding to that tie. If one of the response arcs was omitted (missing), then the other arc response was used as the symmetrized tie response. If both arcs were omitted, then the tie was left missing as well.

The fourth set of data (*UndirWestUS*, $n = 10$) utilized the same eight Western US states as before, but was elicited using the survey design described in Section 2.6. The experts also consisted of students in a graduate level course. The obtained expert response graphs did not need any symmetrization and consisted of 28 questions per expert.

A plot of their actual performance against the known ground truth in Figures 2.5 and 2.6 gives a sense of the expert's overall abilities. This figure shows the empirical distribution of performance for all experts.

For the most part, the experts' actual true positive rate (TPR) is greater than their false positive rate (FPR), a property that the 2HT (Eqs. 2.2 & 2.3) assumes.

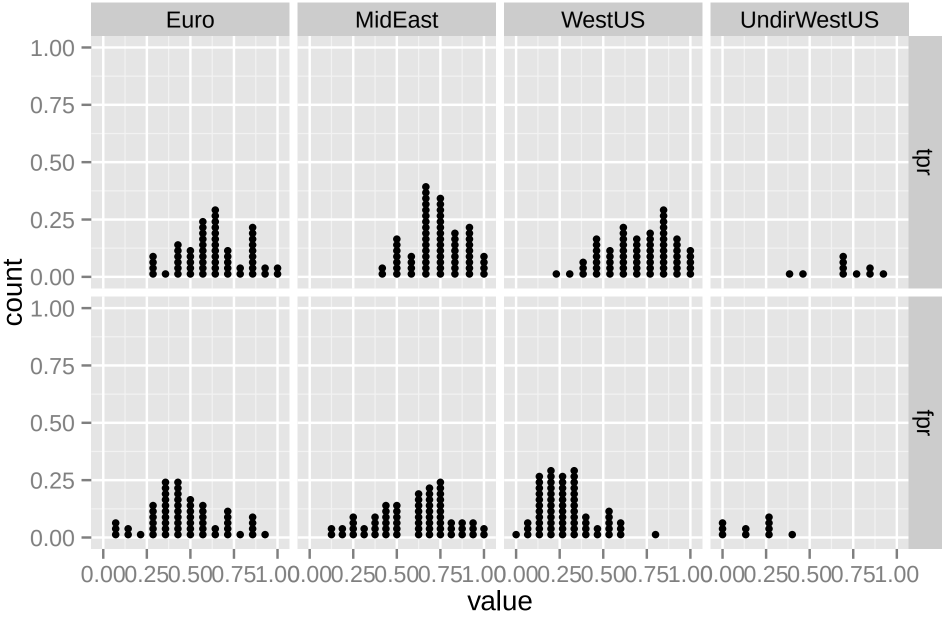


Figure 2.5: Dotplots of experts' true positive rate (TPR, top row) and false positive rates (FPR, bottom row) for each dataset.

2.8.2 Determination of single-factor structure and sufficient signal

Analyses begins by screening the available datasets for evidence of a single-factor solution to the inter-expert response correlations and for overall sufficient signal. Since we can safely assume only one ground truth for geographic knowledge questions, the inter-expert correlations should exhibit a single-factor solution.

We applied the the GCM/Tie Model estimation procedures to each of the four graph datasets and then generated post-predictive data using the samples from the posterior distribution of the model parameters.

Before looking at the results of the posterior consensus (or other parameters), we checked that certain statistics of the original data fell well-within their respective post-predictive distributions.

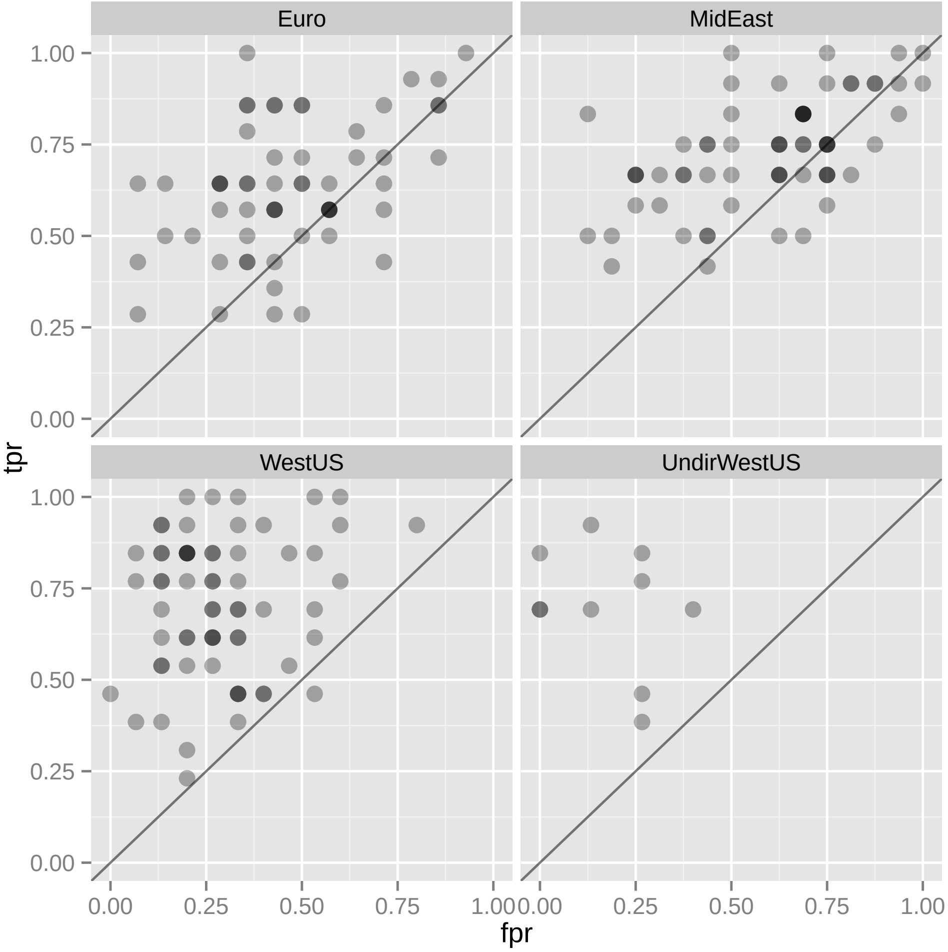


Figure 2.6: Scatterplots of experts' true positive rate (TPR) and false positive rate (FPR) for each dataset, per expert. Darker circles represent multiple experts with the same TPR and FPR.

Such post-predictive checks help assess whether the model is appropriate for given data. Gelman et al. (2004), Gelman and Hill (2006), and Jackman (2009) discuss these procedures in more detail. examine Below, statistics that the model should recapture in post-predictive simulations are examined.

Scree plot

As mentioned before, the Scree Plots for eigenvalues of the inter-expert response correlations have been used to examine if there is a violation of the single-culture assumption. Recently in CCT literature, the ratio of the first to second eigenvalues has been used as a post-predictive statistic since it is indicative of a single-solution (i.e. single culture) in the data.

Figure 2.7 suggests that only the UndirWestUS dataset is modeled correctly according to this statistic of the data since the scree plot of the data falls within the post-predictive "cloud" of scree plots based on the GCM/Tie Model.

First-to-second eigenvalues ratio

We may also look at the postpredictive distribution of the ratios of the first to second eigenvalues of the expert-to-expert correlation matrices. For each of the four datasets, Figure 2.8 shows postpredictive distributions of these ratios using the GCM/Tie Model along with the ratio obtained from the original data. The UndirWestUS dataset and, just barely, the Euro dataset suggest model fit for this test (the black bars denote the limits of each posterior highest-probability-density interval).

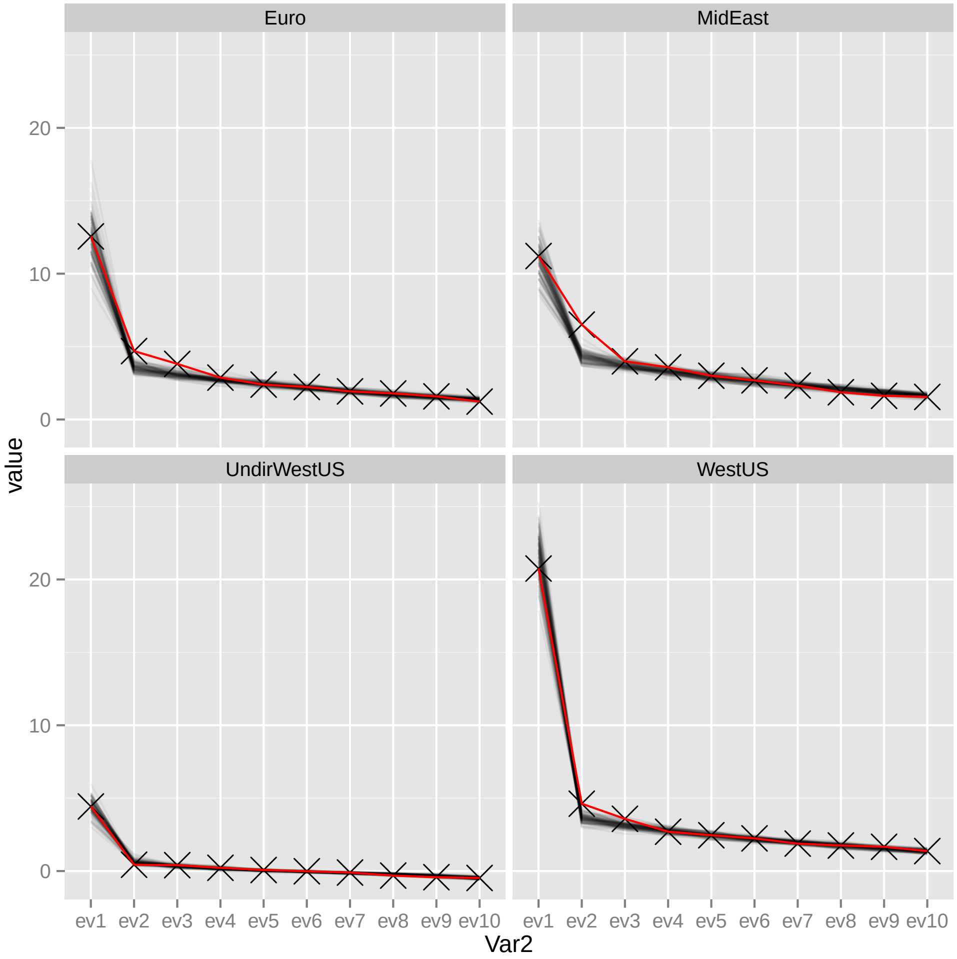


Figure 2.7: Scree clouds from postpredictive data from the GCM.

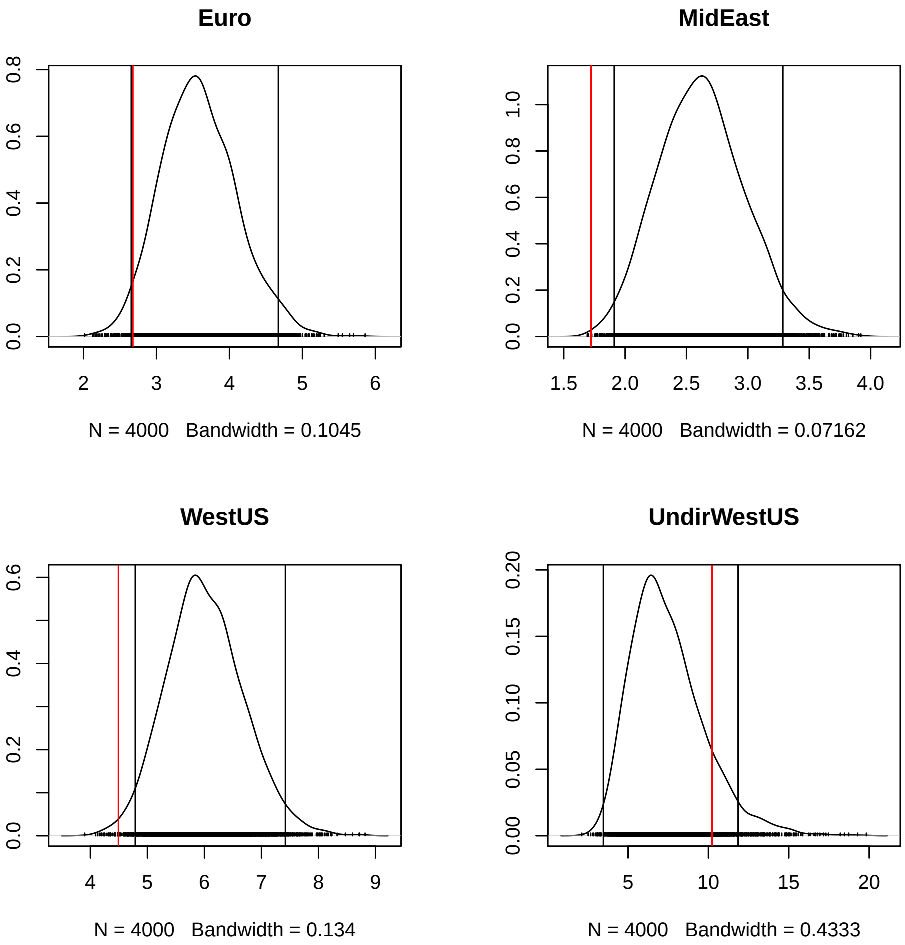


Figure 2.8: GCM Post-Predictive EV1 to EV2 ratio and data point. Black bars represent 95% highest probability density intervals, and the red bars are the ratios obtained from the data.

Mean posterior competence

Of the four datasets, only the UndirWestUS provided a posterior mean competence (posterior $\bar{\mathbf{D}}$) greater than 0.5. The posterior $\bar{\mathbf{D}}$ for Euro, MidEast, WestUS, and UndirWestUS are 0.37, 0.33, 0.45, and 0.56, respectively.

2.8.3 Determination of conditional independence

We are justified in suspecting that the data we employ may suffer from dependence among the responses and thereby thwart the GCM/Tie Model. Experts may tend towards responding to the third (last) tie in a triad of question ties in the manner discussed for social relationships or might tend towards "clumping" nodes that are, for example, spatially connected. For either, we count triads that satisfy the social *friends-of-friends* rule described in Table 2.2, so-called *balanced* triads.

The triad counts can be used as a postpredictive statistic in the same manner as the first-to-second eigenvalue ratios discussed above. Figure 2.9 shows these distributions along with the highest probability density interval and data statistics for each of the datasets and two triad types. For all but the MidEast dataset, the balanced triad type is towards the upper postpredictive 95% HPD interval.

2.8.4 Screening the data

After reviewing the model checks available above, we conclude that the UndirWestUS dataset could be used for further analysis in in an application setting and that the model may not fit the other data to our liking. We proceed to look at all the datasets here, however.

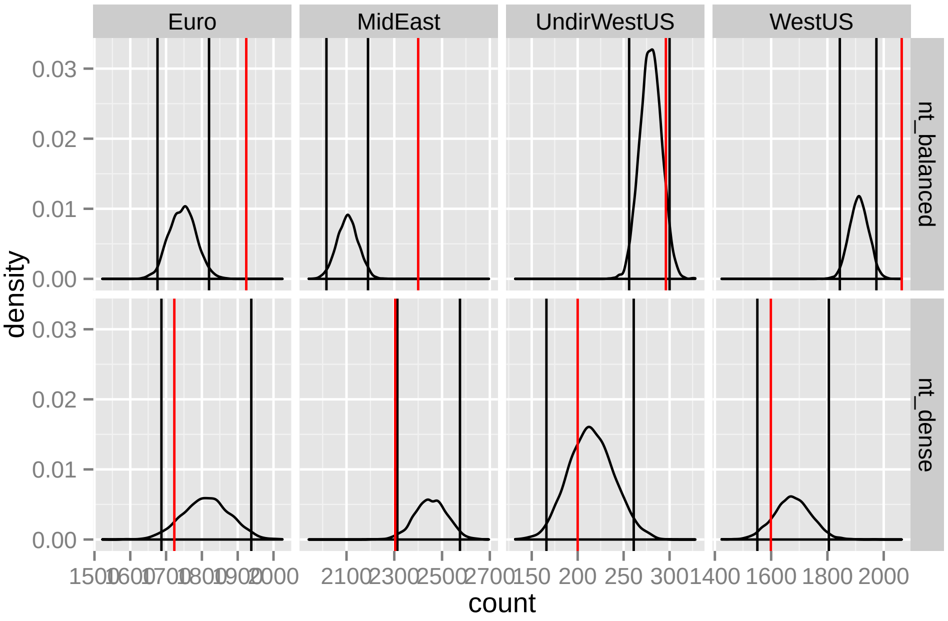


Figure 2.9: GCM postpredictive triad counts, HPDs, and statistic values. The balanced triad statistic for WestUS is at the right edge of the 95% HPD interval. Black bars denote the 95% highest probability density intervals, and the red bars denote the data statistic.

In particular, the postpredictive eigenvalues ratio and postpredictive scree cloud checks only passed for the UndirWestUS data. Postpredictive triad statistics only passed for the UndirWestUS dataset. Additionally, the postpredictive mean competence suggests a reasonably strong recovery of consensus based on the simulation studies previously discussed.

We also note that $\hat{R}$ values for parameters are all in an acceptable range (i.e. low), except for some of the posterior tie parameters. These are somewhat exempt given that they are binary variables for which the $\hat{R}$ is not appropriate. The low overall signal (i.e. $\hat{\mathbf{D}}$) for the Euro, MidEast, and WestUS datasets are supported by the posterior estimates for overall expert abilities. The UndirWestUS was the only dataset to produce posterior mean $\alpha_\mu > 0$, as seen in the left column of Figure 2.10.

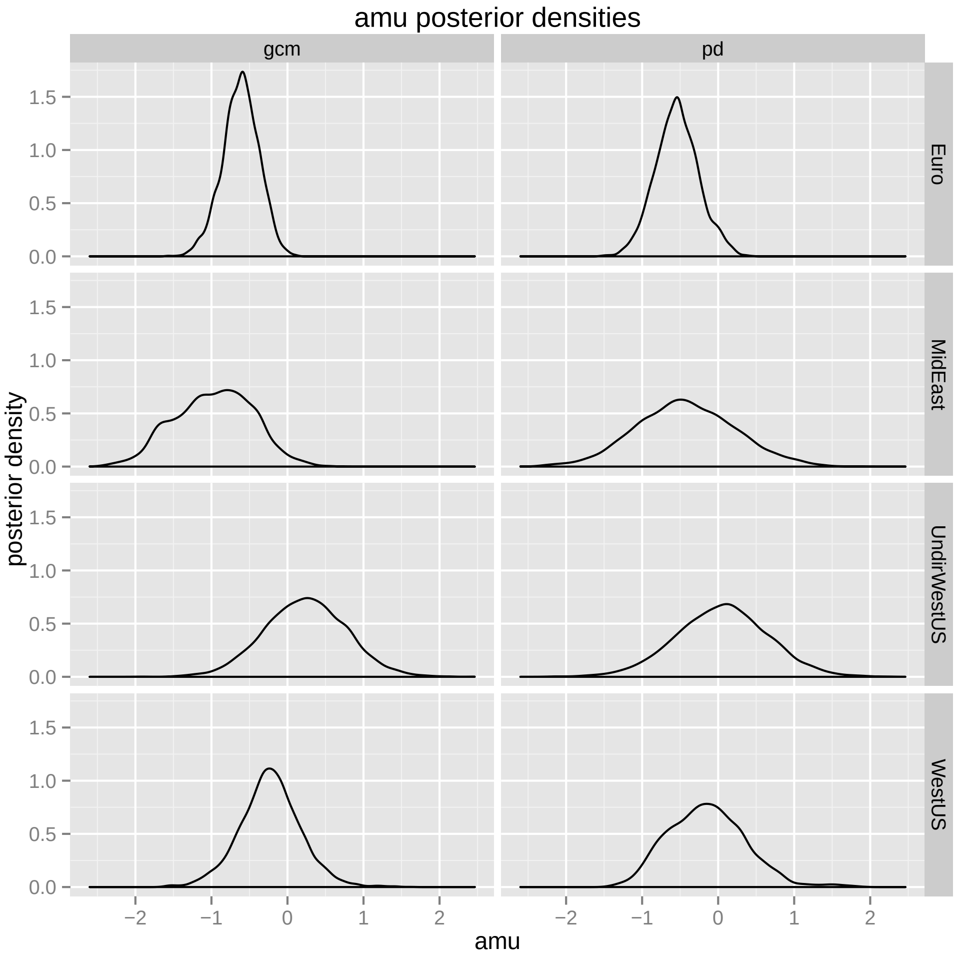


Figure 2.10: Ability location μ_α posterior distributions for each dataset and each model.

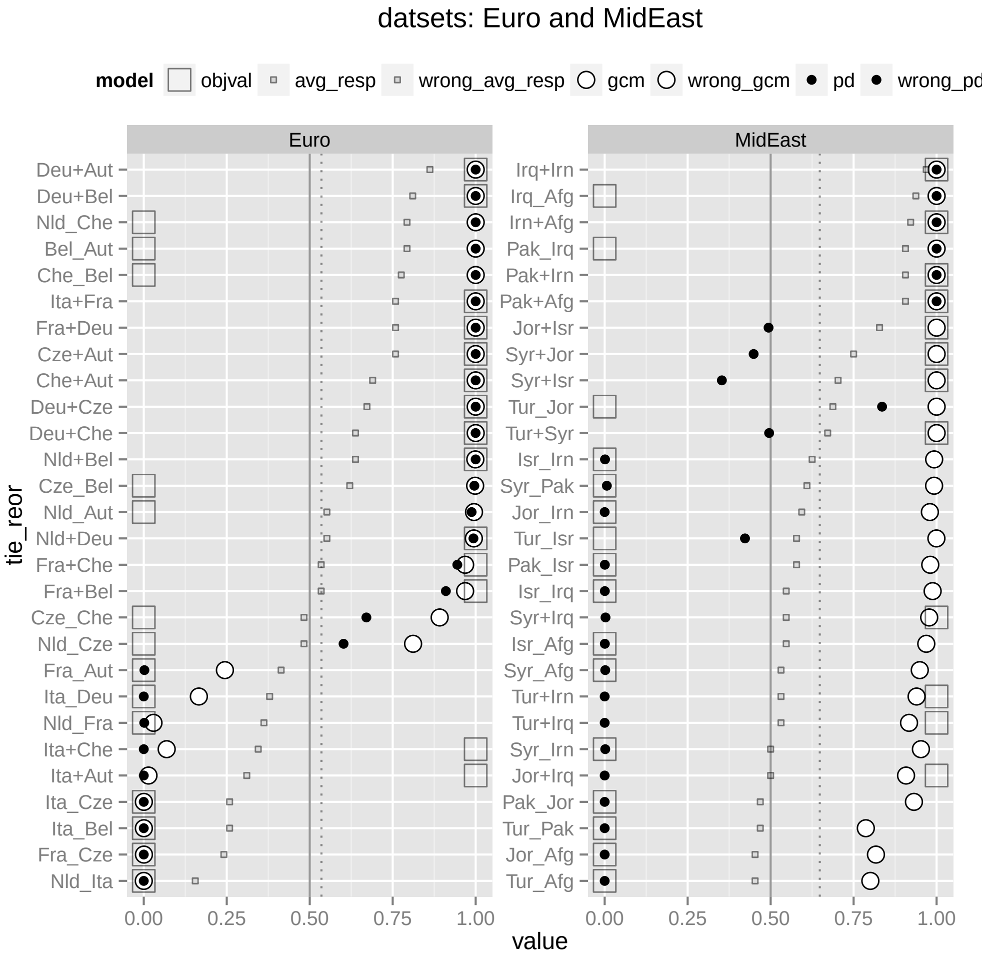


Figure 2.11: Posterior consensus mean Z_{jk} for both GCM/Tie Model and PDGCM, along with mean 'yes' response for each tie. This figure includes the Euro and MidEast datasets. Average response per tie is shown in the gradient of small squares. White circles are the GCM/Tie Model marginal posterior mean consensus tie values. Black circles are the PDGCM marginal posterior mean consensus tie values. The ground truth tie values are indicated by the large squares. Ties correctly identified by the majority rule have an average response on the side of the dotted line (overall response rate) that has a large rectangle (objval). Ties correctly identified by the GCM/Tie Model and PDGCM are on the same side of the solid line (x=0.5) as their corresponding large rectangles.

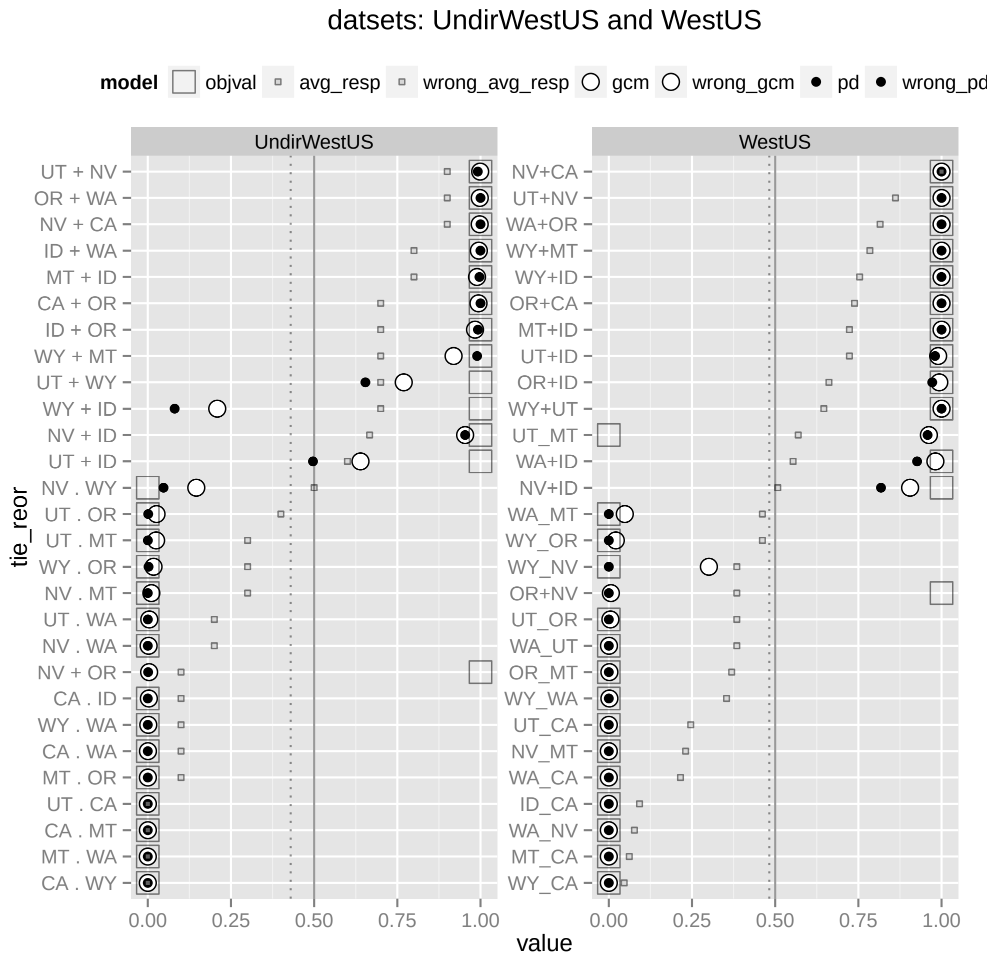


Figure 2.12: Posterior consensus mean Z_{jk} for both GCM/Tie Model and PDGCM, along with mean ‘yes’ response for each tie. This figure includes the WestUS and UndirWestUS datasets. Average response per tie is shown in the gradient of small squares. White circles are the GCM/Tie Model marginal posterior mean consensus tie values. Black circles are the PDGCM marginal posterior mean consensus tie values. The ground truth tie values are indicated by the large squares. Ties correctly identified by the majority rule have an average response on the side of the dotted line (overall response rate) that has a large rectangle (objval). Ties correctly identified by the GCM/Tie Model and PDGCM are on the same side of the solid line (x=0.5) as their corresponding large rectangles.

2.8.5 Posterior consensus

We now turn to one of the main outputs from CCT modeling: the estimated consensus.

The consensus tie value means for each data set are plotted in Figures 2.11 and 2.12. Posterior means near the extremes of 0 or 1 indicate high posterior confidence in that tie's value (as 0 or 1, respectively). Posterior means between these extremes indicate there is some uncertainty in estimated consensus tie value.

We see in these figures that the posterior Z_{jk} confidence is generally high for all the datasets with most posterior tie means towards the extremes. This is strikingly different from the gradation of confidences that the average tie response provide. Some ties, such as Fra_Aut in Euro, Tur_Pak in MidEast, UT + ID in UndirWestUS, and WY_NV in WestUS are indicate uncertainty in the posterior.

There are multiple reasons we could expect such uncertainty. For one, there is a possibility that the experts did not have much collective knowledge of such ties. Except for the notably extreme response bias found in the MidEast dataset, the posterior consensus distributions for the data do not qualitatively hint at the real signal disparity between them. Indeed, we see that overall correct responses, as scored against the known objective truth, are near chance for the Euro and MidEast datasets, but at 0.77 and 0.71 for the UndirWestUS and WestUS datasets, respectively (see Table 2.3). These real scores correspond to the posterior $\bar{\mathbf{D}}$ determined above. None of the datasets are from an overall highly competent set of experts.

Dataset	overall corr.	maj.rule corr.	GCM corr.	PDGCM corr.	GCM $\bar{\mathbf{D}}$	PDGCM $\bar{\mathbf{D}}$
Euro	0.58	0.68	0.68	0.68	0.37	0.37
MidEast	0.55	0.75	0.43	0.61	0.33	0.26
UndirWestUS	0.77	0.93	0.93	0.89	0.56	0.56
WestUS	0.71	0.93	0.93	0.93	0.45	0.44

Table 2.3: For each dataset, the rate of correctness with respect to the ground truth for all responses, majority-rule decision-making, GCM posterior mean decision-making, and PDGCM posterior mean decision-making. The last two columns show the posterior mean $\bar{\mathbf{D}}$ for each of the models on each dataset.

Z pattern	frequency	datset
1111001111101111011001101100	1752	Euro
1111101111101111011001101100	432	Euro
1111001111101111011000101100	320	Euro
1111001111101111011001111100	241	Euro
1111111111111111111111111111	1998	MidEast
1111111111111111111111111101	198	MidEast
1111101111111111111111111111	177	MidEast
1110111111111111111111111111	174	MidEast
0011000111111001010100010010	2159	WestUS
0011000111111001010101010010	991	WestUS
0011000101111001010100010010	282	WestUS
0011000111111000010100010010	86	WestUS
1100000100100100111000001101	1348	UndirWestUS
1100000100100100101000001101	477	UndirWestUS
1100000100100100101000000101	434	UndirWestUS
1100000100100110111000001101	241	UndirWestUS

Table 2.4: The four most frequent posterior consensus patterns and counts for each dataset under the GCM.

Another possibility is that the responses are actually from a multi-factor solution (i.e. there are multiple truths or cultures represented). The eigenvalue analysis already suggests a one factor solution for the UndirWestUS dataset. While the model does not anticipate multiple consensus patterns, multiple modes in the posterior distribution of joint consensus patterns would indicate such an issue. Note the "Top 4" Table 2.4 that provides the top four joint patterns (by frequency) in the posterior consensus for each dataset. Eyeballing these, we see a steep drop in frequencies following the modal pattern and that the less frequent patterns are similar to the modal pattern. There is no indication of multiple modes in the joint posterior **Z** discrete distribution.

We must ask what the GCM/Tie Model give the researcher over majority rule decision making in terms of the consensus. At first, there does not seem to be anything gained at all. Looking again at Table 2.3, we see that majority rule and GCM/Tie Model are equally adept at recovering the (known) ground truth for the low-signal Euro and high signal UndirWestUS and WestUS datasets. The lowest-signal MidEast faired poorer using GCM/Tie Model than for majority rule.

Let us pay close attention to

- ties that have low confidence in expert response (i.e. close to an even split),
- ties for which the GCM/Tie Model provides a highly different estimate than average response, and
- ties with high difficulty.

The reader may find it useful to look at the heatmap response charts at the end of this section (Figure 2.13) while exploring the "oddities".

Looking first at the UndirWestUS dataset which passed our screening, we find that the GCM recovered the ground truth for most of the ties. The most difficult ties (with the top four posterior β_{jk}) are WY + ID, NV . WY, UT + ID, and UT + WY.

The ground truth values of the borders match up with the posterior median tie values in all but two ties. These are the ties WY + ID and NV + OR. NV + OR appears to be a simple case of a generally mistaken understanding of that tie among the experts. More interesting is that the GCM/Tie Model estimates a that WY + ID are not bordering, even though the majority of experts correctly believed they do. For this tie, it is the case that the three experts with the top estimated ability (and actually three of the top four experts with respect to the ground truth) were coincidentally incorrect and opposed to the majority.

Several of the ties that have ambiguous raw response averages have their corresponding posterior mean Z_{jk} "pushed" towards the extremes. For NV . WY, the response average was on the incorrect side of the majority rule threshold (i.e. majority rule would not have matched the ground truth), but the GCM/Tie Model "corrected" for this and resolved the ambiguity.

NV . WY and WY + ID are interesting because they illustrate a strength and weakness in the GCM/Tie Model algorithm (and CCT in general). The weakness is that a minority of high-ability experts can override a low-ability majority, even when the majority is correct and the minority is incorrect. This is the case in the UndirWestUS data, where the three top α_i experts all responded "no" (0) to these two questions, thereby overriding the majority. Such an occurrence was likely chance in this dataset, but collusion to undermine the data is a potential threat to CCT. The strength of such phenomenon is that a minority of experts with access to correct knowledge can also override an ignorant majority.

We see UT + ID as the most contentious (ambiguous) posterior estimate. It is helpful to look at the heatmap responses at the end of this section. We see that the highest ability experts (top rows) are in disagreement, highlighting the importance of agreement high-ability expert responses in determining confidence in consensus.

Although the states covered by the WestUS dataset are the same as above, the expert population and survey method were different. Notably, however, border of Oregon and Nevada seems to be missed by both of these pools of experts. Majority rule and GCM/Tie Model incorrectly identified the same two ties (MT_UT and OR+NV). As with the UndirWestUS, the highest ability experts dominate the consensus estimates. And with the WY_NV tie, where the highest-ability experts *disagreed*, there was the least posterior confidence. Contrast this against the ties MT_UT, WA+ID, and ID+NV,

The power of correlation between experts is a detriment in the MidEast data. Looking at the heatmap and posterior **Z** plots, we see that several experts responded to nearly all questions with “yes” (1, light blue). Recovery of the ground truth is poor for the GCM/Tie Model. The overall response bias observed has overwhelmed the posterior such that the modal consensus graph suggests that all the states mutually border one another! One clear take-away is that surveys answer options should be balanced (either within a survey or between) to mitigate the issue where an expert simply checks off a “line” of responses down the page. Interestingly, when majority rule decisions are made using the overall mean response bias as a threshold, the experts perform better than the GCM/Tie Model. Further discussion about the bias problems in the MidEast dataset will be discussed after introducing the PDGCM.

The Euro dataset does not indicate any interesting phenomenon for a CCT application except for a general lack of knowledge of the region among the experts surveyed.

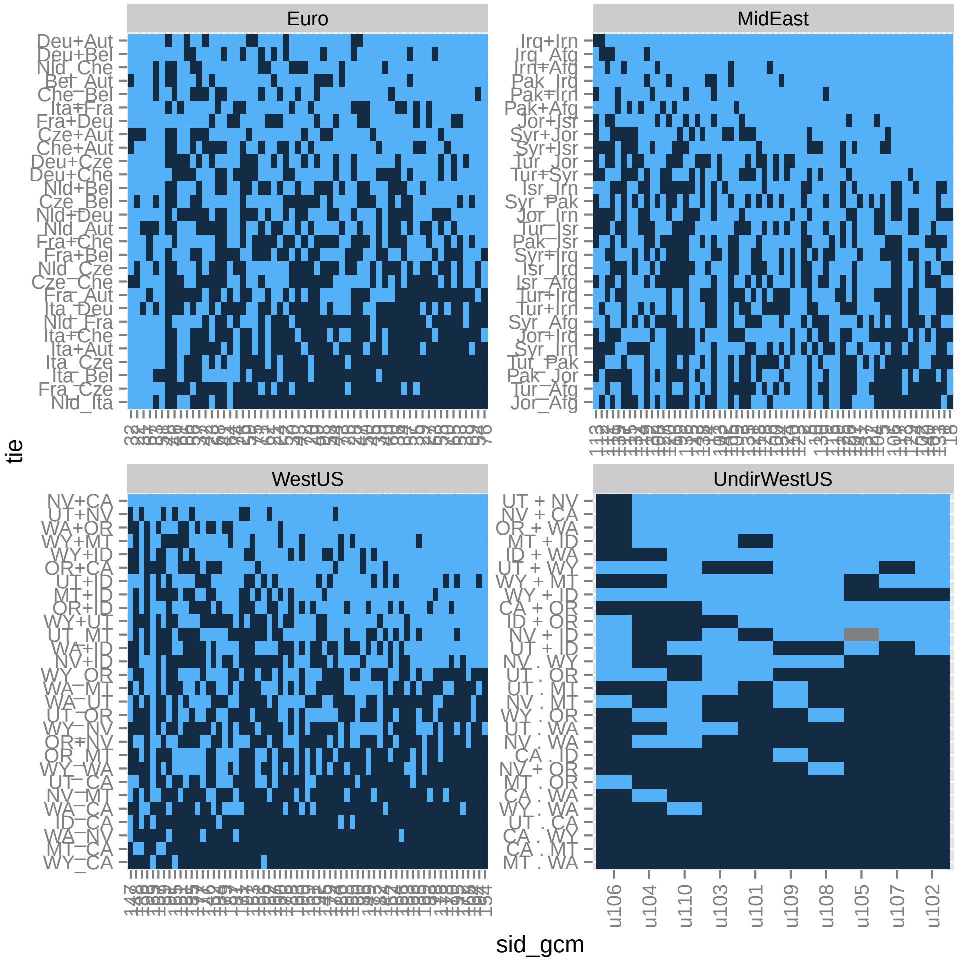


Figure 2.13: Data 'heatmap' plots. Rows ordered by average response. Columns ordered by GCM posterior α_i such that "best" experts are on the right. Light blue indicate positive response, and dark blue represents a negative response.

2.8.6 Discussion of results

The pushing of consensus values towards the extremes of their average response seen here is typical of CCT models. That is, the models tend to produce higher posterior confidence in the existing expressed majority beliefs.

One thing that is established is that CCT (via GCM) can solve some of the lack of aggregate clarity (definitive aggregate responses). Significantly, we should find some "reversals" and also some "tie-breakers", where the GCM responses are opposite of naive-mean opinion, or at least are able to break ties of half-half even responses.

In the case of the UndirWestUS, we see that there were a small group of three experts with high ability who held knowledge that the majority did not. This can only happen for some of the ties or else they would not have such relative high posterior ability (due to matching less with the rest of the experts). It is for these questions that CCT may be especially useful: When do high-ability experts seem to have different high-quality knowledge that differs from the majority?

The high correlation of experts in the MidEast dataset may also lead to its undoing. In the model, due to the high degree of correlation among the biased experts, the model accumulates a lot of information about their abilities and not so much about guessing bias. That is, the high correlation is attributed to ability rather than due to bias (chance).

2.9 Paired Difficulty Model (PDGCM)

We now consider a variant of the GCM that reparameterizes the tie difficulties as the sum of constituent node difficulties.

Modeling heterogeneous difficulty appears to be interesting and telling. However, what if heterogeneity in difficulties is better explained at the *nodal* level? For the experts from California, all ties to California should be easier than, say, all ties with Idaho.

Composing a tie's difficulty from the pair of difficulties of its nodes reduces the number of parameters. That is, each node has some difficulty associated with it, and all ties with that node incorporate its difficulty. This approach is used for modeling digraphs by Batchelder et al. (1997). Here is presented a modification of the previous GCM/Tie Model with a new function of tie difficulty that is composed of the mean of the respective nodes' difficulties. The Rasch model is retained for the expert competence on a given tie, as in the GCM/Tie Model.

The PDGCM is the same as the GCM/Tie Model in all ways except for a reparameterization of tie difficulty. Each GCM/Tie Model tie difficulty is a latent parameter β_{jk} indexed by the tie's nodes, {j} and {k}. The PDGCM model breaks this into a mean of two nodal difficulties, namely

$$\beta_{jk} = \frac{\tau_j + \tau_k}{2} \ . \tag{2.22}$$

In the Tie Model (GCM) with the Rasch parameterization, the difficulty β_{jk} is conveniently on the same scale as the abilities. Here, by averaging τ_j and τ_k, the scale of measurement is assured to be the same for ability α_i and each node difficulty τ_j.

As in the GCM/Tie Model, experts have latent expert-item response competence and latent guessing bias that apply to their responses to each tie.

2.9.1 Specification of the PDGCM

The complete response model shares Axioms 2.1 - 2.3 with the GCM/Tie Model and replaces Axiom 2.4 with the following axiom.

Axiom 2.5 (Rasch knowledge)**.** *Each expert has an ability, $\alpha_i \, \epsilon \, \Re$, and each node has its own difficulty, $\tau_j \, \epsilon \, \Re$, such that*

$$D_{i,jk} = \frac{1}{1 + exp\left(-\left(\alpha_i - \frac{1}{2}(\tau_j + \tau_k)\right)\right)} \ . \tag{2.23}$$

2.9.2 Likelihood of observed data

The likelihood for the PDGCM is the same as (2.7) from the GCM/Tie Model, excepting the substitution of (2.23) for $D_{i,jk}$. We obtain the same joint likelihood for the PDGCM as in the GCM/Tie Model, and it shares the ability to handle missing data.

2.10 Bayesian specification of the PDGCM

2.10.1 Response model

The response model portion of the Bayesian Graphical Model is no different from the GCM/Tie Model, described in 2.8. That is, their response probability is the same given given a probability that the expert will respond to a tie with the consensus value without guessing is $D_{i,jk}$.

2.10.2 Expert bias, node and tie difficulty, and expert ability prior distributions and hyperdistributions

Expert response bias is modeled in the same way as the GCM/Tie Model (2.9).

The $D_{i,jk}$ parameter is now parameterized according to (2.22). The individual difficulties are analogous to the tie difficulties of the GCM/Tie Model,

$$\text{Individual node's difficulty:} \qquad \tau_j \sim \text{Normal}(0, \sigma_\beta) \ . \tag{2.24}$$

The node difficulty scale parameter floats as a random variable, as in (2.11) and (2.12) of the GCM/Tie Model.

Individual ability with respect being able to compare a given node is also the the same as the GCM/Tie Model (2.13) and (2.14).

2.10.3 Consensus tie priors

The consensus graph is modeled in the same fashion as the GCM/Tie Model. It is an unconstrained graph with positive density prior given by (3.18) and tie value priors given by (3.19).

2.11 Properties of the PDGCM

Since the PDGCM shares so much with the GCM/Tie Model, we will consolidate description of the model properties, implementation, data acquisition, and simulation studies. The implementation and estimation of the PDGCM follows very closely to the GCM/Tie Model and is also implemented using the JAGS program. The exact code listings are in the appendix.

Axiom 2.5 reduces the number of parameters from $M + N(N-1)/2$ in the GCM/Tie Model to $M + N$ for modeling ability and difficulty. This is a key benefit, especially for larger models as the number of these expert and item parameters are linear with respect to the experts and items being analyzed, even as the number of comparisons grows polynomially.

The change in parameterization of the tie difficulty potentially adds something interesting about the identification of the model.

Note that the Bayesian specification still maintains that the mean of the difficulties is 0. This allows for comparison of relative difficulties of the nodes, which is one thing we are interested in drawing out from this model. The mean of the population abilities is still a random hyperparameter and will therefore place the population level ability with respect to the (zero) average node difficulties. These are all on the same scale, as before, where the Rasch "competition" now poses the single expert againts the combined difficulty of the two nodes. The zeroing of the difficulties fixes the identification issues in the Likelihood itself.

Simulation studies run in the same fashion as for the GCM/Tie Model did not reveal any substantive differences between the simulation behaviors of the two models.

For the purpose of comparison, we use the same data for analysis on the PDGCM that was used in the GCM/Tie Model.

2.12 Results of applying the PDGCM

Questions asked of the GCM/Tie Model are generally appropriate for the PDGCM: How well does the consensus match the known ground truth? How "crisp" is the consensus? How well do ability and difficulty parameters measure experts and items? Given the ground truth, we can ask about the validity of the ability and difficulty parameters as measurements.

For data screening, the postpredictive scree clouds were qualitatively similar to those found using the GCM, i.e. similar to Figure 2.7. Only the UndirWestUS dataset passed this screening.

The postpredictive ratios of first-to-second eigenvalues indicated that only the MidEast dataset was a poor fit, with the real ratios falling within the 95% HPDs for Euro, WestUS, and UndirWestUS.

The posterior $\bar{\mathbf{D}}$ for Euro, MidEast, WestUS, and UndirWestUS are 0.37, 0.26, 0.44, and 0.56, respectively. These corroborate the competence estimates in the GCM/Tie Model and notably indicate an even lower level of competence for the MidEast data (yet very similar for the others; see Table 2.3).

2.13 Discussion of both models

This section will compare the results of the PDGCM with those found from the GCM/Tie Model on the same four datasets. Follow-up analyses and extensions will be proposed.

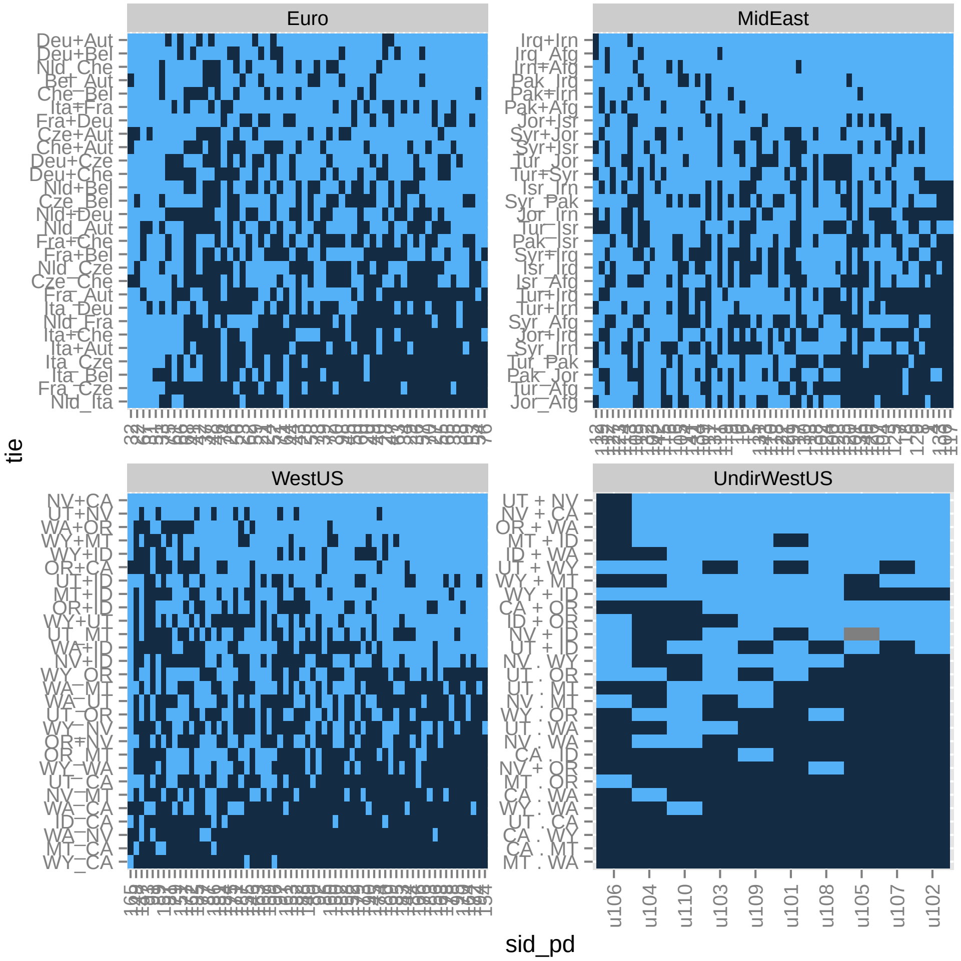


Figure 2.14: Data ‘heatmap’ plots. Rows ordered by average response. Columns ordered by PDGCM abilities, α_i.

2.13.1 Comparison of results

Ability and bias

The posterior densities for the μ_α (ability location) across the two models is not interesting other than that the two models met our expectation that the MidEast dataset has overall low ability experts and that the UndirWestUS had the highest overall ability (Figure 2.10).

Figures 2.15, 2.16, and 2.17 show the posterior distributions for both the GCM/Tie Model and the PDGCM on the σ_α (ability scale), $\bar{\mathbf{G}}$ (mean response bias), and p_Z (consensus tie bias) parameters, respectively. These distributions suggest that the PDGCM is able to determine a greater range of differences in expert ability in the MidEast dataset where the GCM was not. While the central tendency of ability estimates across the two models were comparable, the distribution of abilities is much wider in the PDGCM. The structure of the model makes a trade-off in estimation: when there is lower estimated signal, the data will have greater influence in estimating the guessing biases. The actual tendency for experts to respond "yes (borders)" found in the MidEast is better reflected in the PDGCM since less signal is available, and this shows up in the discrepancy between the two models' estimates of $\bar{\mathbf{G}}$. The extreme response bias detected in the MidEast indicates there must be relatively less information about the consensus bias, p_Z, in the posterior as well. The GCM/Tie Model indicates an extremely high p_Z, but the PDGCM estimate is not so different from the prior. For the MidEast, in particular, the PDGCM does a better job at reflecting underlying response and ground truth biases.

Consensus

The key differences between the two models' posterior consensus estimates lay in the difficult MidEast dataset (Figures 2.11 and 2.12). The other datasets show qualitatively similar

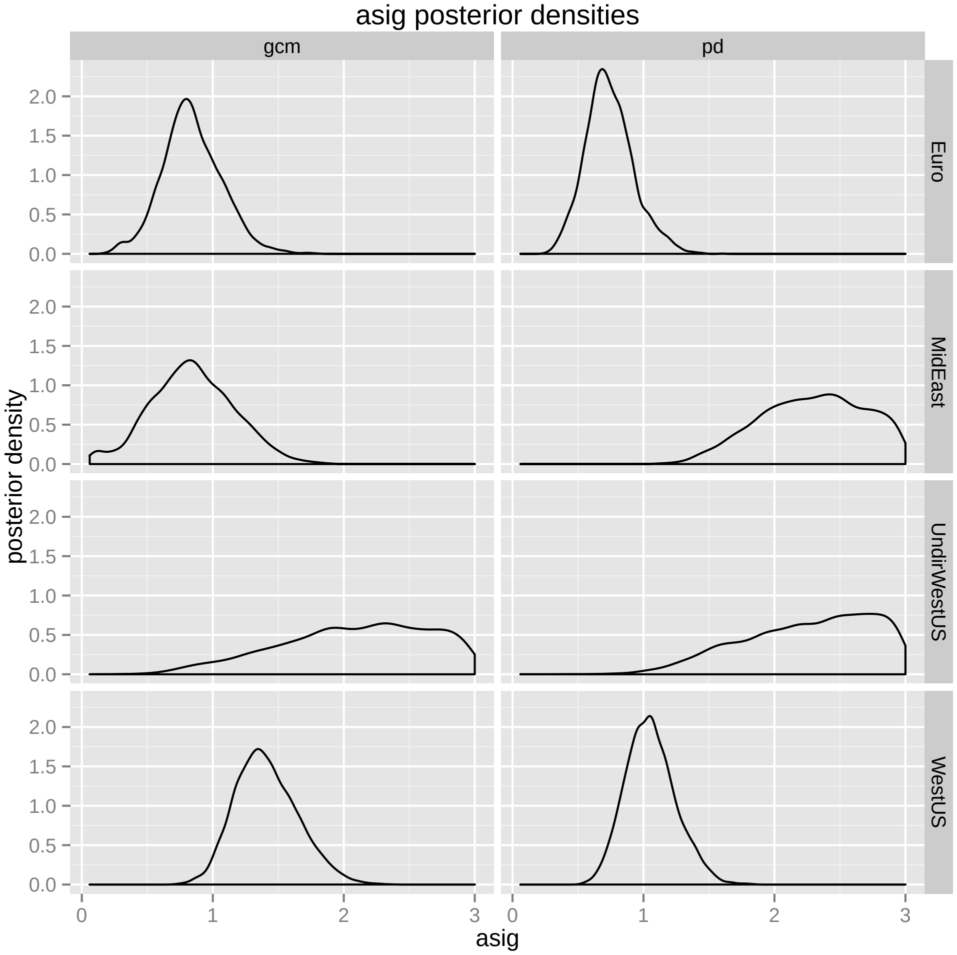


Figure 2.15: σ_α: Interpret this as the measure of breadth of abilities across subjects. For problematic datasets, the PDGCM may be able to capture the range of ability differences that the GCM cannot (i.e. in the MidEast row).

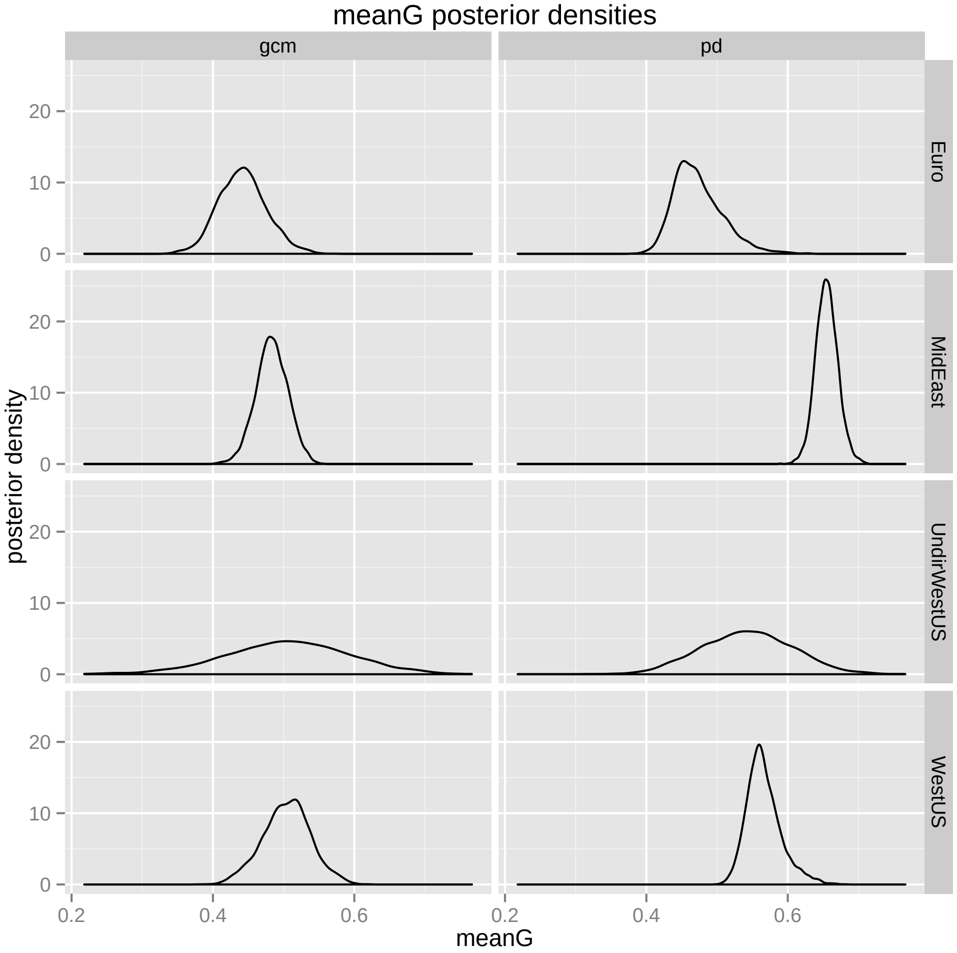


Figure 2.16: $\bar{\mathbf{G}}$: Interpret as overall estimated guessing bias. Notably, the PDGCM is able to capture the experts' bias towards positive (border) connections for the difficult MidEast dataset. The GCM, most likely due to an evenly spread expert competence, does not account for the clear bias in the data. The PDGCM is able to extract a truer estimate of guessing bias because it has identified the low-ability experts and accumulated truly high-positive bias from them. This, despite the overall mean ability being roughly the same for the two models.

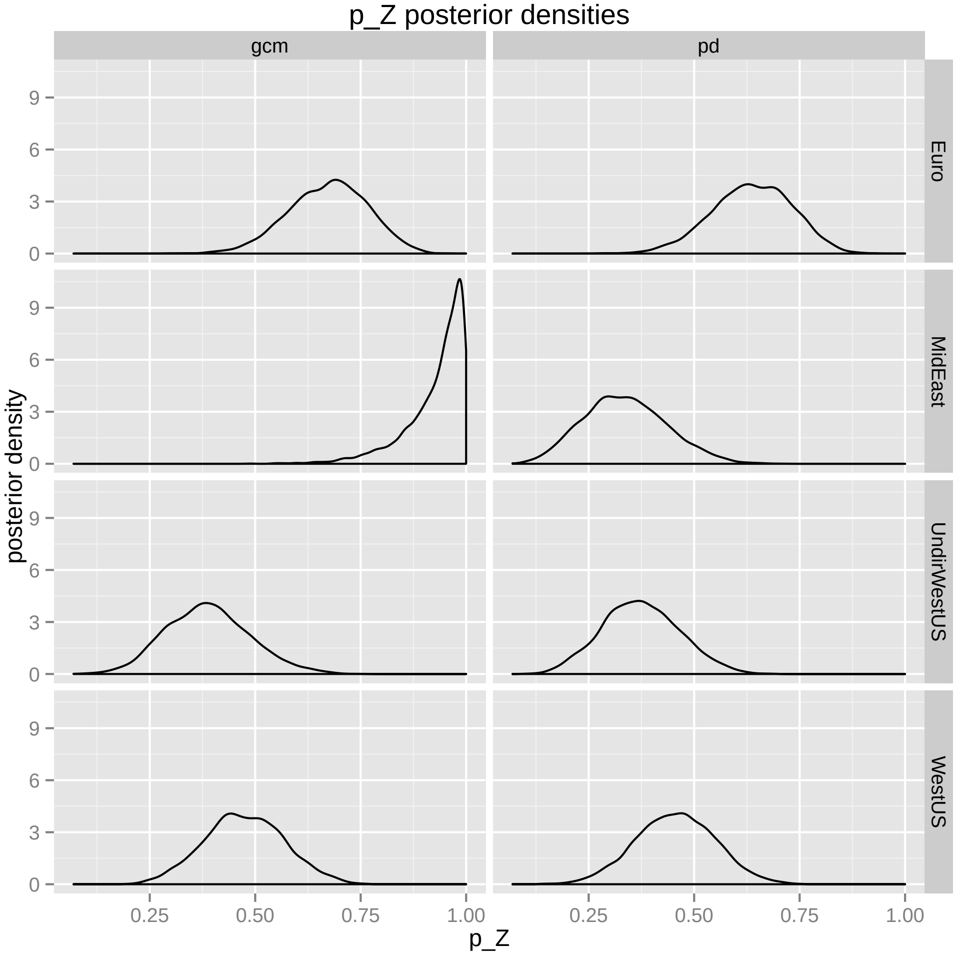


Figure 2.17: p_Z: Interpret this as the overall rate of probability that a tie is positive (borders) in the consensus. The key difference is in the MidEast dataset. The GCM, assuming roughly even abilities across the experts, assumes a fair amount of overall signal. The overall response rate is then interpreted as a rate of positive in the consensus. However the PDGCM is, as discussed in the σ_α and $\bar{\mathbf{G}}$ and μ_α discussion, is able to provide a better estimate of the actual probability that a tie is a border or not.

consensus results across the two models. Modal consensus values differ across the two models on one tie in the UndirWestUS dataset but in three-quarter of the ties (21 of 28) in the MidEast dataset.

When looking at modal marginal tie values, a crisp (modal) consensus is ofter desired by researchers seeking definitive answers to the questions. For UndirWestUS, the UT + ID border lacked clear consensus for both models, and a slight difference in posterior means changed the classification. The Euro and WestUS datasets estimated the same modal consensus tie values for each model (albeit with different tie means).

The most drastic change can be seen in the MidEast dataset. The majority of ties were classified with posterior mean tie values on the opposite end of the unit interval for 16 of them. The PDGCM indicates uncertainty for several ties for which the GCM/Tie Model did not. The posterior consensus for five (six, including one that did not change mode) ties reflect uncertainty that we would expect from the dataset with such poor expert performance.

Two cliques

The PGDCM's estimated that the four easiest nodes are Afghanistan, Iran, Iraq, and Pakistan. These form a clique of bordering states in the modal posterior consensus for the PDGCM. The only other positive tie in the PDGCM is Tur_Jor.

Suppose, instead, that a consensus map is based on posterior mean Z_{jk} rather than the mode, and that a threshold could be varied to accept a tie as "borders" in the consensus. The modal tie value corresponds to a 0.5 threshold. If this threshold were lowered to accept ties with uncertainty in posterior Z_{jk} as bordering, then the other bordering states in the consensus would be Jor+Isr, Syr+Jor, Syr+Isr, Tur_Jor, Tur+Syr, and Tur_Isr. These six ties are a clique of the other states not in the aforementioned clique. That is, the PDGCM identified a partition of the nodes of two cliques of "bordering" countries: One is a near-certain clique of {Afghanistan, Iran, Iraq, Pakistan}, and the other is a less-certain clique of {Israel, Jordon, Syria, Turkey}. All other ties are between these groups and are definitively not-bordering in consensus.

I believe this reflects not the state of bordering in experts memories so much as the state of associations based on news and media reports. Perhaps too, the close names of "Iraq" and "Iran" can be blamed for some responses as well. Based on real geography, the partition evokes "Western" and "Eastern" sets of country nodes.

Even without the lowered threshold for border status in the posterior, there is a clear posterior clique of Afghanistan, Iran, Iraq, and Pakistan that are isolated from the rest of the group. The low difficulty associated with these nodes is highlighted by the PDGCM. A similar, if incomplete, formation is also evident in the average response data.

Ability and performance

A comforting note about these two models is that the estimated expert abilities correlate highly with each other for each of the datasets (Figure 2.18). The datasets of Euro and MidEast, however, show the poor correlation between the expert's performance against the known ground truth and their estimated abilities. For the MidEast, the highest performing expert (objectively) was rated quite low in terms of ability (Figure 2.19). The correlation between expert abilities and their performance improved markedly from 0.095 with the GCM to 0.34 with the PDGCM.

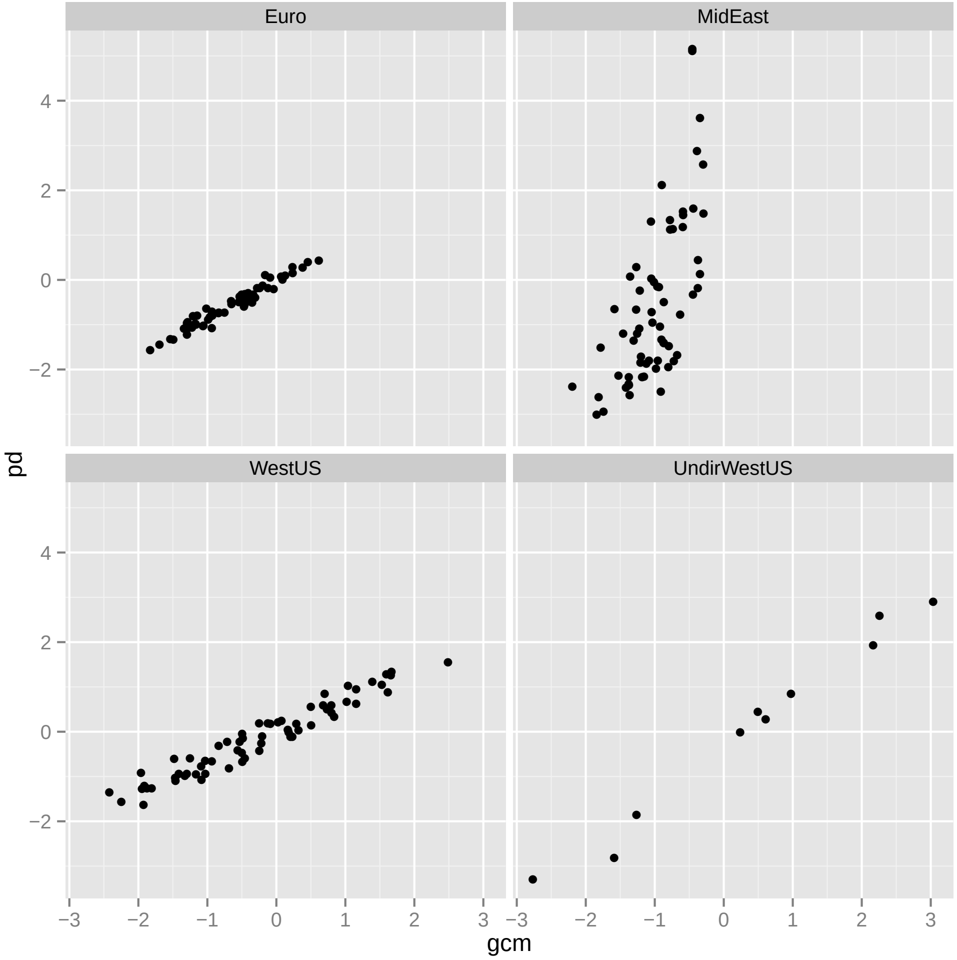


Figure 2.18: Ability estimates comparing means from GCM and PDGCM.

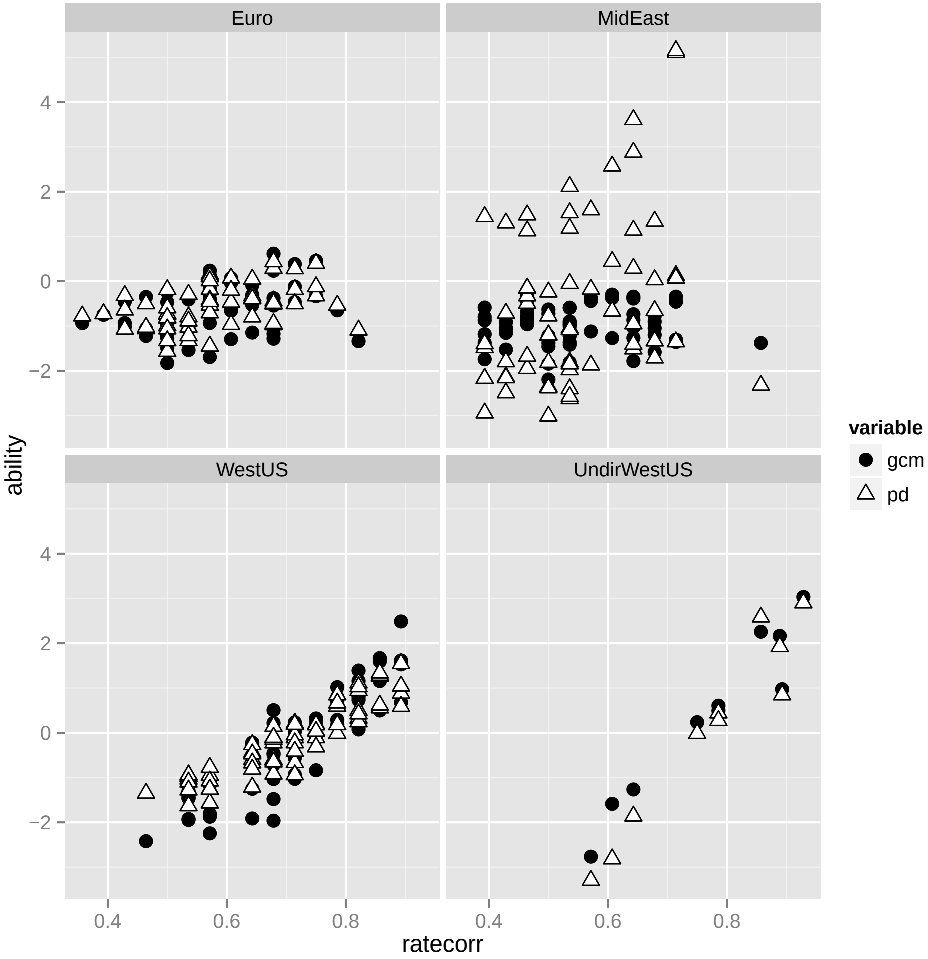


Figure 2.19: Expert performance as rate correct (ratecorr) on the x-axis and posterior mean ability estimate on the y-axis. Notice the high correlation for the WestUS and UndirWestUS datasets and low correlation for the Euro and MidEast for both GCM/Tie Model and the PDGCM.

Subset(s) of MidEast

One weakness of the GCM and PDGCM are their susceptibility to colluding experts. Even though the elicitation task reduces this risk, experts can coordinate their responses by simply using an extreme response bias. These experts then seem, to the model, as overly-able.

We ran subsequent estimation procedures on subsets of the MidEast dataset. One subset omitted four experts whose response patterns were extreme at over 0.9 in the affirmative ("borders") or the negative ("not borders"). The second omitted 31 experts whose responded in the affirmative or negative at a rate of 0.7 or more.

The first subset showed no qualitative difference in posterior consensus from the original. The GCM showed less certainty, as would be expected with reduced data.

The second subset showed a marked difference in the GCM posterior consensus but not as much with the PDGCM. The GCM remarkably matches the modal PDGCM tie values in all but one case.

2.13.2 Next steps

Dual bias model

The PDGCM revealed that the four least difficult nodes were the four with the highest connectivity between them in the posterior. Furthermore, subsetting the MidEast data to exclude extreme response patterns brought the GCM and PDGCM estimations somewhat more in line with each other.

Both of these issues suggest that bias is not handled properly by these models. And correlation between the posterior ties and difficulties raise a question: Is an expert's response to a tie dependent on the ease of the tie or nodes? That is, if a tie is easy for an expert, does that mean that it is also a "border" tie? If a tie is difficult, does the expert tend to respond "not borders"? One method of handling this is to allow for bias to depend on item difficulty.

Unit interval parameterization

Crowther et al. (1995) presents a three factor unit-interval parameterization of the kind of linear composition of factors used in the PDGCM. The GCM has benefited from such a parameterization (Karabatsos and Batchelder, 2003); the researcher benefits from more interpretable parameters. Crowther et al talk about the Fuzzy Logic Model of Perception (FLMP) and link it to the Rasch model: "we demonstrate how to reformulate FLMP as a simple logit model.". The two factors for a response of "positive" (in the case of signed graph elicitation) would be based on fuzzy truth (i.e. [0,1] scale) value representing the degree to which each factor contributes to support the classification of "positive" (or "1", "true", "detect", etc.). Such a parameterization would fit into the PDGCM with three factors being the expert, node j, and node k.

Toss out inconsistent digraph arcs

The original digraph data collected for the MidEast, Euro, and WestUS could provide a way to discard data that is logically inconsistent within an expert. Extensions of CCT to accommodate inconsistencies within a survey can take advantage of identifying certain lack of knowledge of the consensus. An easy approach is to simply consider an expert's response to a tie as a missing datum when their response to the corresponding forward and backward arcs are different.

Heatmap visualizations

We expect certain structures to emerge in the arrangements (orders) of expert and item parameters when the data really does come from the proposed generative models.

See Figure 2.20 for "heatmaps" of the response data that take the following form:

1. Obtain the data matrix of expert by items.
2. Reorder the experts to be in the following fashion:
   - Find the expert with the highest estimated ability α_0 and obtain their estimated mean bias $\bar{G}_0$.
   - Partition the experts into two groups: experts with $\bar{G}_i \leq \bar{G}_0$ and those with $\bar{G}_i > \bar{G}_0$.
   - For the first subset, order them from least-to-best ability.
   - For the second subset, order them from best-to-least ability.
   - Concatenate these two lists, first then the second.
   - This is the order to use in the heatmap.
3. Reorder the ties:
   - First by posterior Z_{jk}.
   - Then resolve potential ties by ordering by average response.

The expected look of this should be with the center columns having clearly defined positive ties on top and negative ties on bottom. Then, tapering off to either direction the responses should become more random, but biasing towards either 0 (in one direction) or 1 (in the opposite). Consequently, there should be an inverted "triangle" of high density "1" responses approaching from the top and a triangle of "0" responses rising from the bottom.

While this is qualitative analysis only, the two datasets with the most signal signal show the triangular patterns for both positive and negative ties, whereas the Euro and MidEast datasets suggest only one triangle: a low (0) and high (1), respectively.

2.13.3 Conclusion

This chapter reviewed the GCM/Tie Model as applied to graph data as well as an extension of it (PDGCM) that is addresses the correlation of node difficulty by breaking GCM item difficulty into a sum of constituent nodal difficulties. This paper presents additions to the CCT literature by addressing hierarchical Bayesian modeling for responses on undirected graphs for non-embedded experts. Such additions are notable because they handle item heterogeneity without the use of in-degree or out-degree parameterization as has been used previously. The models handle inherently undirected graphs, something that several other aggregation models do not.

We also present new datasets that can be used for validation of these and other models on undirected tie-based response data. The inclusion of this data is due in part to a new survey design that mitigates the potential dependence of expert responses based on heuristics or memory that the models cannot capture. These data show that experts have imposed a structure (of clusters) on the geographic proximity (i.e. borders) of the MidEast. That is, with CCT, at least for knowledge of geographic border knowledge in undergraduates at UCI, cultural truth shows structure that the ground truth does not really have.

When applied to the datasets, the PDGCM and GCM perform comparably on data where there is strong signal. On a particularly weak-signal dataset with a high degree of overall bias, the PDGCM was able to distinguish biased expert responses better than the GCM.

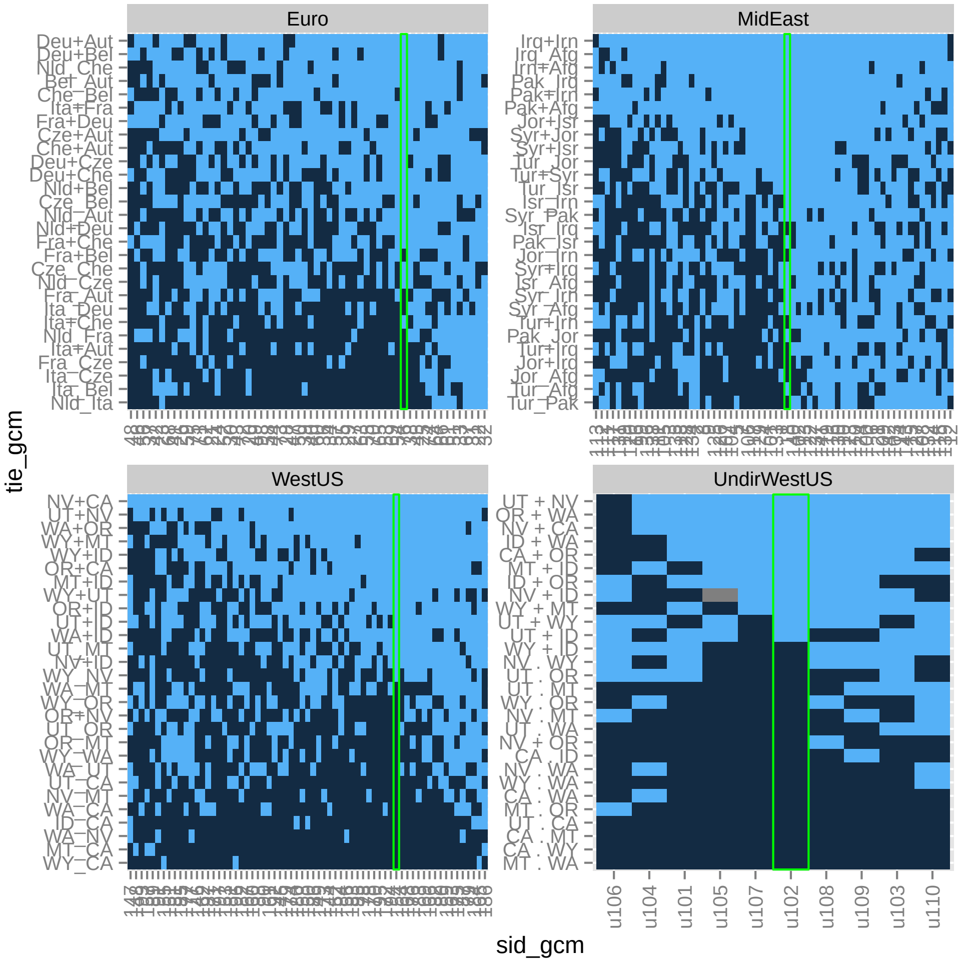


Figure 2.20: Exploratory heatmaps of the data. Order of rows (ties) and columns (experts) is described in the text. The highlighted expert has the highest estimated ability α_i. Experts to the left have lower estimated bias than the best expert. Within the group of experts to either side of the best expert, the abilities are ordered such that they decrease away from the best expert. The result is that the data with the most influence on the consensus is centered around the highlighted column. The least influential data on the estimated consensus is at the left and right extremes, separated by low and high biases, respectively. Rows are ordered by posterior Z_{jk}. The most *uncertain* ties are therefore centered and the most certain ties are at the top and bottom.

An added benefit of the PDGCM over the GCM for graphs is the greatly reduced parameter count with respect to modeling item heterogeneity; the count of difficulty parameters is made linear with respect to the number of items being compared.

Several extensions and suggestions proposed for follow-up would provide even better modeling for these interesting elicitation formats and would continue to build upon the library of CCT formats.

Chapter 3

Binary nodal knowledge models

3.1 Introduction

This chapter introduces Cultural Consensus Theory (CCT) models geared towards pooling experts' responses on undirected pairwise relationships among a fixed set of nodes. Motivated by Batchelder (2009), we add two types of constraints to CCT models. The first type of constraint is on the experts' latent knowledge when responding to questions about the same node within a survey. The second type of constraint is on the structure of consensus graph on which the experts are reporting.

This chapter introduces a discrete (binary) nodal knowledge layer to the GCM/Tie Model and PDGCM approaches to the pairwise judgments discussed in Chapter 2. This knowledge layer is introduced for two models that share the response portion of the model (likelihood) but differ in the Bayesian specification of the consensus prior. To present it in a more straightforward fashion, we will first introduce the overall Nodal Rasch (NR) model. When the consensus is an unconstrained graph, it will be labeled the Unconstrained NR (UNR) model. When the consensus prior imposes a hard constraint of a three-cell partition of nodes on the consensus graph, then it will be labeled the K3NR (for a partition with $k = 3$ cells).

The premise of this chapter is that the GCM/Tie Model and PDGCM response models do not capture certain kinds of undirected pairwise judgments. For example, consider a scenario where items are compared for the purpose of clustering based on similarity (similar/not similar) or proximity (near/far) when cluster names are not known in advance by the researcher or experts. And so a motivation for the nodal knowledge model comes from differentiating these two types of questions given to experts:

1. Affect / connection:
   - "Have Avery and Blake met?"
   - "Does New York state border Vermont?"
2. Class comparison:
   - "Are the cities of Bakersfield and Bend in the same state?"

Both types are factual, signed, and undirected. The first focuses on the connection between nodes, and is served by GCM/Tie Model and the PDGCM. But the second variety invites comparing nodes on some attribute(s), for which the NR models are targeted. The NR models assume that (a) all comparison questions being asked are on the same attributes of comparison and (b) knowledge of the attributes of one node persist throughout all questions related to that node. In the example above, the state in which a city lies is an attribute. When an expert "knows" the attribute of comparison for two nodes being compared, the answer is straightforward. And the longevity of that knowledge of the attributes would reasonably remain with an expert when asked for comparisons that share a previously-compared node. In contrast to the models discussed in the previous chapter, these models concern reports on the comparisons of pairs of nodes rather than on the nature of their relationship.

New in the NR models is a response model that maintains an expert's level of competence with regard to a given node across all questions related to that node. These models assume that each expert either does or does not have in their memory an adequate representation of the comparative properties of each particular node, and these expert-nodal statuses remain fixed throughout the response survey. This is a more realistic model of how comparisons of pairs of nodes are made. We hope that these models that allow for consistent responses across an entire survey are better suited for the type of task described than classic CCT models.

The remainder of the chapter is structured as follows. Section 3.2 presents the NR response model. Section 3.3 adds Bayesian specifications and distinguishes the UNR and K3NR models. Following these, Section 3.5 discusses some of the algorithms involved with Bayesian inference on these models. Section 3.6 provides tests using simulation studies. Section 3.7 applies the models to experimental validation data. Finally, we conclude with general discussion, further steps to undertake, and ideas for future research in Section 3.8.

3.2 Nodal Rasch (NR) model

As with the GCM/Tie Model and PDGCM, let $\mathbf{Z}$ be the adjacency matrix for a complete signed graph on N nodes (1.1). $Z_{jk} = 1$ when tie $\{j, k\}$ is '+' and $Z_{jk} = 0$ when it is '$-$'. $\mathbf{Z}$ is the unobserved consensus graph. Experts' responses graphs are obtained as input data, where $\mathbf{X} = \{X_{i,jk} | 1 \leq i \leq M, 1 \leq j < k \leq N\}$ contains the response graphs from all the experts. This model assumes that for each node, each expert has a latent binary *knowledge* indicator that allows accurate reporting of ties between that node and other *known* nodes.

This process is assumed to be based on knowledge prior to the survey, and before any tie responses are elicited. The tie judgments themselves are made correctly if an expert knows both nodes, or takes a guess otherwise. The survey questions are also expected to emphasize the comparison of the nodes.

At the point of tie judgment, the response model assumes that the expert detects the consensus directly if they can make comparisons on both of the tie's nodes or else they make a biased guess. The response model is very similar to that described by (2.4) in Chapter 2, except that per-expert-per-tie competence is discrete binary. The model specifies parameters for expert nodal knowledge probabilities and guessing probabilities as follows. Let $D_{i,k} \epsilon [0, 1]$ be the probability that expert i has nodal knowledge of node k, and $K_{i,k} \epsilon \{0, 1\}$ be the expert's nodal knowledge indicator for node k, and $G_i \epsilon [0, 1]$ the probability expert i guesses '+' to any tie $\{j, k\}$ in the absence of knowing both nodes (i.e. for ties where either $K_{i,j} = 0$ or $K_{i,k} = 0$). We collect these expert parameters into arrays $\mathbf{D} = \langle D_{i,k} \rangle$, $\mathbf{K} = \langle K_{i,k} \rangle$ and $\mathbf{G} = \langle G_i \rangle$. The response model portion is stated below. As observed data, we have the researcher accumulate responses from M experts to the values of a signed, undirected graph of N nodes. Observed data are therefore a list of M graphs, $\mathbf{X} = \langle \mathbf{X}_i \rangle$, and individual responses by expert i to tie $\{j, k\}$ are $X_{i,jk} \epsilon \{0, 1\}$.

3.2.1 Specification of the NR response model

Axiom 3.1 (Common truth)**.** *All experts respond according to a shared, but unobserved, complete signed consensus graph as in* (1.2)*:* $\mathbf{Z} = \{Z_{jk} | 1 \leq k < j \leq N\}$ *on* N *nodes.* $Z_{jk} = 1$ *when edge* $\{j, k\}$ *is '+' and* $Z_{jk} = 0$ *when it is '−'.*

Axiom 3.1 is identical to Axiom 2.1 from the GCM/Tie Model.

Axiom 3.2 (Conditional independence of experts)**.** *The likelihood of observing all the experts' response graphs given the consensus graph is the product of all the likelihoods of observing each expert's particular response graph given the consensus graph. Namely for all realizations* $\mathbf{x}$ *of* $\mathbf{X}$*,*

$$\Pr(\mathbf{X}|\mathbf{Z}) = \prod_{i=1}^{M} \Pr(\mathbf{X}_i = \mathbf{x}_i|\mathbf{Z}) \ , \tag{3.1}$$

where $\mathbf{x}_i = \{x_{i,jk}|1 \leq j < k \leq N\}$ *is expert i's response graph.*

That is, other than the shared consensus, experts respond independently of one another.

Axiom 3.2 is akin to the conditional independence assumption of the GCM/Tie Model found in Axiom 2.2. In the NR, the responses from all the experts are conditionally independent of *each other*, but not within an expert, given the consensus graph.

Axiom 3.3 (Latent nodal knowledge)**.** *Each expert possesses a latent random binary knowledge indicator for each node, where*

$$K_{i,k} = \begin{cases} 1 & \textit{if expert } i \textit{ knows node } k, \\ 0 & \textit{if expert } i \textit{ guesses on ties with node } k \ . \end{cases} \tag{3.2}$$

And the probability that an expert knows (is able to compare) a node is $D_{i,k}$*, such that*

$$Pr(K_{i,k} = 1) = D_{i,k} \tag{3.3}$$

We will write $\mathbf{K}_i = \langle K_{i1}, \ldots, K_{iN} \rangle$ as expert i's *knowledge vector*, and array $\mathbf{K} = \langle \mathbf{K}_1, \ldots, \mathbf{K}_M \rangle$ for all knowledge.

There are several ways one might parameterize an expert's probability of knowing a node, $D_{i,k}$. Here, we utilize the Rasch model again.

Axiom 3.4 (Rasch model of nodal knowledge)**.** *The probability of expert i knowing node k is given by*

$$D_{i,k} = \frac{1}{1 + exp(-(\alpha_i - \tau_k))} \ , \ \alpha_i, \tau_k \ \epsilon \ \Re \ , \tag{3.4}$$

where $\alpha_i \ \epsilon \ \Re$ is a measure of an expert's ability, and $\tau_k \ \epsilon \ \Re$ is a measure of difficulty for a node.

Axiom 3.4 reduces the number of expert-node parameters from $M \times N$ to $M + N$ and is a typical modeling step in CCT and Item Response Theory (IRT) (Fischer and Molenaar, 1995). The result is that the logit expert-node competence $D_{i,k}$ of expert i being able to compare node k is a linear function of the expert's ability and the node's difficulty, i.e. (2.20). Therefore, the expert-node competence may be construed as a sort of competition between the expert and the node where only their *relative* difference matters.

Axiom 3.5 (Marginal tie responses)**.** *An expert either correctly reports $X_{i,jk} = Z_{jk}$ if they know both nodes, or guesses $X_{i,jk} = 1$ with probability G_i if they do not. This is given by*

$$\begin{aligned} &\Pr(X_{i,jk} = x_{i,jk} | Z_{jk}, \mathbf{K}_i, G_i) = \\ &\quad (x_{i,jk}) \left[K_{i,j} K_{i,k} Z_{jk} + (1 - K_{i,j} K_{i,k}) G_i \right] + \\ &\quad (1 - x_{i,jk}) \left[K_{i,j} K_{i,k} (1 - Z_{jk}) + (1 - K_{i,j} K_{i,k})(1 - G_i) \right] \ . \end{aligned} \tag{3.5}$$

This equation (3.5) makes use of the multiplication of two binary numbers as the logical equivalent to the "and" operation (conjunction), where $K_{i,j} K_{i,k} = 1$ iff both $K_{i,j} = 1$ and $K_{i,k} = 1$.

The marginal response probability of (3.5) can be viewed as the sum of the mutually exclusive events connected to responses of $x_{i,jk} = 1$ (yes/positive) and $x_{i,jk} = 0$ (no/negative) crossed with consensus values $Z_{jk} = 1$ and $Z_{jk} = 0$. The probability of responding yes/positive is exactly 1 if both nodes are known and the consensus is yes/positive or equal to the guessing bias otherwise. For example, the probability of responding no/negative is exactly 100% if both nodes are known and the consensus is no/negative or one minus the guessing bias otherwise.

Axiom 3.5 is analogous to Axiom 2.3 except the expert-item competence in the GCM/Tie Model is continuous and here its place is held by the binary product of $K_{i,j}K_{i,k}$. In other words, this model's marginal tie response is based on an all-or-none expert-item competence driven by the newly-added *knowledge layer* $\mathbf{K}$.

3.2.2 Likelihood of observed data

The response likelihood is the likelihood of the observed data given the immediate conditioned parameters. There are four possible partial likelihood values for a given i, j, k triple. The resulting probability $\Pr(X_{i,jk} = x_{i,jk}|Z_{jk}, K_i, G_i)$ can be zero, one, G_i, or $1 - G_i$. An expert's joint response probability is the product of these item response probabilities.

Below we provide each of the possible response probabilities, $\Pr(X_{i,jk}|Z_{jk}, K_{i,j}, K_{i,k}, G_i)$, for all combinations of $K_{i,j}K_{i,k}$, $X_{i,jk}$, and Z_{jk}:

$K_{i,j}K_{i,k}$	$X_{i,jk}$	Z_{jk}	$\Pr(X_{i,jk}\|Z_{jk}, K_{i,j}, K_{i,k}, G_i)$
0	0	0	$1 - G_i$
0	0	1	$1 - G_i$
0	1	0	G_i
0	1	1	G_i
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

(3.6)

Importantly, when $K_{i,j}K_{i,k} = 1$, the likelihood is zeroed out whenever any expert's response is not equal to the consensus. That is, the likelihood is zero when $X_{i,jk} \neq Z_{jk}$.

This table also highlights that response probabilities are unaffected by the consensus tie value in the absence of knowledge: the value remains either G_i or $1 - G_i$ for those (rows 1-4). In the case of knowledge, the consensus tie value determines whether that marginal probability is either zero or one (rows 5-8).

There is something to note about the likelihood of parameter sets that vary only in the Z_{jk} (consensus) parameter. For parameter arguments that are the same except for some difference in the consensus Z_{jk}, the product likelihood is either zero, a single constant between zero and one, or one. The only way that $\mathbf{Z}$ affects the likelihood is when it zeros it out because there is a conflict between the answer key and an expert's response when that expert has knowledge for the conflicting tie ($K_{i,j}K_{i,k} = 1$). This will have an implication in the estimation algorithm to be discussed later.

3.3 Bayesian specification of the UNR and K3NR

In this section we state the specific parameters, priors, hyperpriors, and hyperparamaters to complete the Bayesian graphical model. We use a mixed hierarchical model, where certain parameters' distributions are drawn from a hyperdistribution of fixed effects. The two variants of the NR model are characterized here by the consensus graph prior. A similar form of this model was used in Agrawal and Batchelder (2012). In contrast to that model, this paper includes the additional knowledge layer for both the UNR and K3NR models. Borrowed from Agrawal and Batchelder (2012) is an updated variant of the constrained consensus prior for the K3NR model.

To proceed, we first state the response model described thus far by using standard Bayesian Graphical Model distribution notation. Gelman et al. (2004) and Jackman (2009) use this notation and provide good background on the hierarchical Bayesian modeling used here. The response model is followed by specification of additional priors needed to complete the model including the two consensus priors that complete the two NR models.

In general, prior distributions are chosen to reflect a researcher's prior understanding of parameters involved. When there is a lack of prior knowledge, one arguably leans towards using a so-called "non-informative", or "flat", prior which reflects the researcher's uncertainty in the distribution of the parameter's values (e.g. all values are equally probable).

3.3.1 Response model

The response model of Axioms 3.2 and 3.5 and may be rewritten in distribution notation without changing or adding to the model. In distribution notation, we write the probability that a given response is positive, i.e. $\Pr(X_{i,jk} = 1)$, as

$$\text{Response:} \qquad X_{i,jk} \sim \text{Bernoulli}((K_{i,j}K_{i,k})Z_{jk} + (1 - K_{i,j}K_{i,k})G_i) \tag{3.7}$$

3.3.2 Knowledge model

Modeling the *knowledge layer* is akin to the competence model for the GCM.

The Rasch equation (3.4) of Axiom 3.4 that remains as written since it is not a random variable.

$$\text{Knowing a node:} \qquad K_{i,k} \sim \text{Bernoulli}(D_{i,k}) \ . \tag{3.8}$$

3.3.3 Expert bias, node difficulty, and expert ability prior distributions and hyperdistributions

Expert guessing bias is set to match Bayesian GCM variants such as Karabatsos and Batchelder (2003), where

$$\text{Expert guessing bias:} \qquad G_i \sim \text{Uniform}(0,1)\ . \tag{3.9}$$

Following Batchelder and Anders (2012), we note that prior research in uses of the GCM suggests that guessing biases tend to be estimated near 1/2 and we infrequently find real *informative* experts providing extreme tendencies in overall response (i.e. responding all 'positive' or all 'negative') or extreme estimated biases.

Expert abilities are drawn from a Normal distribution centered on 0, and with a latent random standard deviation of its own distribution,

$$\text{Expert ability:} \qquad \alpha_i \sim \text{Normal}(0,\sigma^2)\ . \tag{3.10}$$

Each expert's ability is therefore drawn from a population-level mean of 0, but with a latent scale. The zero-centered ability identifies the model and is needed since the Rasch function (3.4) does not vary when the same value is added to both the ability and difficulty parameters. The shared scale parameter floats as a random variable with its own hyperdistribution:

$$\text{Ability scale upper limit:} \qquad \kappa = 3 \ , \tag{3.11}$$

$$\text{Scale of abilities:} \qquad \sigma \sim \text{Uniform}(0, \kappa) \ . \tag{3.12}$$

This choice of a uniform distribution over the scale parameter is borrowed from Jackman (2009); the uniform is useful because it is straightforward to interpret and has easy-to-set hyperparamaters. The specific value of $\kappa = 3$ was chosen as a reasonable upper limit to the deviation around the mean.

The choice of nodal difficulty prior and hyperprior distributions is analogous to the expert abilities in that they have complementary influence on the resulting probability of expert-node competence in the Rasch model (3.4). However, unlike the abilities, which are centered at 0, each node's difficulty is drawn from a Normal distribution with another latent scale parameter and latent population mean,

$$\text{Difficulty scale upper limit:} \qquad \kappa = 3 \ , \tag{3.13}$$

$$\text{Scale of difficulties:} \qquad \lambda \sim \text{Uniform}(0, \kappa) \ , \tag{3.14}$$

$$\text{Location of node difficulties:} \qquad \theta \sim \text{Uniform}(-2, 2) \ , \tag{3.15}$$

$$\text{Node difficulty:} \qquad \tau_k \sim \text{Normal}(\theta, \lambda^2) \ . \tag{3.16}$$

The location of the difficulty population, θ, can be uniform without affecting identification of the model because the abilities have a fixed mean (at zero). The choice of the $[-2, 2]$ range for the distribution is based on testing that shows that the data sizes we use do not benefit from extending the range further.

While different from some other CCT models, the priors and hyperdistributions chosen here are easy to interpret and work with and should apply to a wide range of populations.

3.3.4 Consensus tie priors

This subsection adds the *U* in the *UNR* and the *K3* in the *K3NR* model names. Both consensus priors are introduced here, and discussion and application are taken up later.

The UNR model

The UNR model assumes a prior on consensus tie values as conditionally independent, and therefore the whole consensus may be factored as a product of their respective conditional (tie) prior distributions, where each tie of the graph independently has the same probability of being 1 (or 0). The expectation of the density of positive ties in the graph is a hyperparameter in $[0, 1]$. With conditionally independent ties, we have

$$\Pr(\mathbf{Z} = \mathbf{z}) = \prod_{j=2}^{N} \prod_{k=1}^{j-1} \Pr(Z_{jk} = z_{jk}) \ . \tag{3.17}$$

This is a standard CCT assumption, where each consensus item Z_{jk} is modeled as a Bernoulli trial with some expected base rate of being one answer or another. We model this with a hierarchical uniform prior distribution on the single probability,

$$\text{Consensus graph positive density:} \qquad p_Z \sim \text{Uniform}(0,1)\ , \tag{3.18}$$

$$\text{Consensus tie value:} \qquad Z_{jk} \sim \text{Bernoulli}(p_Z)\ . \tag{3.19}$$

The choice of a uniform distribution on p_Z is typical (e.g. Batchelder and Anders, 2012), though other modelers may decide to put a hierarchical distribution on p_Z in the presence of more information. In the present application, we assume lack of knowledge about the overall expected density of positive ties in the consensus graph. This implies that each configuration of consensus tie values is equally likely *a priori*.

The K3NR model

The K3NR model follows Agrawal and Batchelder (2012) and extends it to clusters of arbitrary size (2, 3, ...). The implementation discussed here fixes the number of clusters enforced to $k = 3$.

The K3NR model assumes that the nodes belong to a partition such that each node belongs to one and only one cell. In the consensus graph, nodes in the same cell of the partition all have positive ties between them and no negative ties, and nodes in different cells have negative ties between them. In general, the labels of partition cells are in $\{1, \ldots, N\}$. In K3NR, we maintain each node's cell label in a partition vector, $\mathbf{W}$, where $W_k \in \{1, 2, 3\}$.

For example, for $N = 4$ nodes, the partition vector $\mathbf{W} = \langle 1, 2, 1, 1 \rangle$ has two cells whereas $\mathbf{W} = \langle 3, 2, 4, 1 \rangle$ has the maximum possible of four cells (one for each node). This last $\mathbf{W}$ is equivalent to the *partition* $\mathbf{W} = \langle 1, 2, 3, 4 \rangle$, though the *labeling* differs. This redundant labeling is sometimes called *label-switching* problem and must be addressed in order to interpret results. Without worrying about label switching redundancies, there are K^N possible node label patterns.

However, if labeling is restricted to a *canonical* order, where labeling moves from left-to-right, drawing from the smallest available cell number first, and if partitions are composed of exactly k non-empty cells, then the number of partitions of exactly k non-empty subsets (cells) is the *Stirling Number of the Second Kind*, denoted $S(N, k)$.

Let $\mathcal{K}(\mathbf{w})$ give the number of different label levels used in $\mathbf{w}$, that is the number of cells used. In this case of the K3NR, we ensure that the consensus partition has exactly three non-empty cells $\mathcal{K}(\mathbf{w}) = 3$ by imposing a set of restrictive priors such that partitions that do not have exactly three non-empty cells have a zero probability.

For the supported partitions, the researcher is left to set the relative non-zero prior probabilities. We chose a simple, "flat" prior on the remaining supported partitions by making them equally probable in the prior:

$$\text{Exact number of cells to enforce:} \qquad k = 3 \tag{3.20}$$

$$\text{Consensus node cell assignment probabilities:} \qquad \Pr(\mathbf{W} = \mathbf{w}) \begin{cases} = 0, \text{if } \mathcal{K}(\mathbf{w}) \neq k, \\ \propto 1, \text{otherwise.} \end{cases} \tag{3.21}$$

Notice that the non-zero probability in (3.21) is left proportional to a constant for the specification. This is because the algorithm does not require exact conditional distributions since only ratios of prior probabilities are computed. Under this specification, the relative proportion is always unit, which slightly reduces the work of the inference algorithm.

The consensus partition in the K3NR model implies a corresponding consensus graph **Z** that can be derived from **W** and is partitionable into exactly three non-empty cells of positive tie cliques:

$$\text{Consensus tie value based on partition:} \qquad Z_{jk} = \begin{cases} 1, \text{if } W_j = W_k, \\ 0, \text{otherwise.} \end{cases} \tag{3.22}$$

3.4 Properties of the NR models

The NR models both share the common truth of the GCM/Tie Model and PDCM, as seen in Axiom 3.1 and Axiom 2.1. However, in Axiom 3.2 this model departs from the GCM/Tie model in that the expert responses are no longer conditionally independent within an expert, but only between experts.

Axiom 3.3 provides each expert a within-survey realization of their competence about node k, called their "knowledge" of that node. While this knowledge is probabilistic, the realized latent binary indicator $K_{i,k}$ is applied to all tie judgments with that node which provides a constancy of specific knowledge throughout the elicitation procedure that is new to CCT. Each knowledge vector $\mathbf{K}_i$ specifies which nodes are known by expert i, and therefore the ties to which they necessarily respond correctly. The known nodes in $\mathbf{K}_i$ form a knowledge *clique* for the expert where *all ties* between known nodes are known and therefore reflect the consensus exactly. When such cliques partially or fully overlap, any pairs of nodes known by multiple experts must have matching responses since they are linked through the consensus graph, **Z**.

Axiom 3.4 applies the Rasch model to detection probabilities, just as Axiom 2.4 does for the GCM/Tie Model. In this case, the probability that an expert knows a given node (i.e. detection, $D_{i,k}$) is determined by pitting the expert's ability against the node's difficulty. For example, a node might be inherently more or less difficult to observe than others, while an expert may may be more or less observant than her peers. Applying the Rasch model results in a great reduction of parameters and simplifies the model in a sensible way.

There is a parallel between this model's Axiom 3.5 and Axiom 2.3 of the GCM/Tie Model and PDGCM, as well. Both use the same two high threshold (2HT) detection model. However, in the GCM/Tie model/PDGCM, the ability for an expert to answer a particular question item is continuous in $(0, 1)$, but is a discrete $\{0, 1\}$ in the NR models. Furthermore, the NR models apply the 2HT to the intermediate knowledge layer rather than an actual response.

The possibility of zero likelihood, given certain configurations of data and parameters, further distinguishes the NR models from other CCT models. This may occur under one circumstance for each tie, for each expert: it is when a single expert's response to a tie is "known", yet the consensus tie value differs from their recorded response. That is, if $K_{i,j} = K_{i,k} = 1$ and $X_{i,jk} \neq Z_{jk}$, then the likelihood will be zero. A more intuitive occasion when this happens (though not necessary for the zero condition) is when two experts "know" the value to the same tie, yet have different responses in the data. Note that since the consensus value for the tie is either of two exclusive values, such a case presents a contradiction where one of the experts "knows" the tie and actually matches the answer key while the other expert "knows" the same tie but responds differently from both the other expert and the consensus.

For the UNR, nodal knowledge can be interpreted as expert i's ability to make comparisons of node k with other known nodes. For the K3NR, this nodal knowledge may be interpreted as the expert's knowledge of a node's cell within a partition (such as a node's latent class). For example, suppose a given expert may be presented pairs of basketball players, with each question asking if they are on the same team or not. The expert may know the team names (cell labels, classes) for only some of the nodes but not for others. The expert would therefore be able to correctly respond to the questions about "same team" class membership when they can access and compare the team names for both players ("Lakers" is not the same as "Bulls"). In the current NR models, the expert would be forced to guess if they only know the team membership for one of them. In other CCT models, the specific knowledge used for answering one question does not carry over to other questions as it does here.

Finally, we note that the consensus partition of the K3NR model must have three non-empty cells represented. In this way, we get around the notoriously difficulty problem of estimating the number of clusters based on the data – the number of clusters is specified ahead of time. Researchers interested in estimating the number of clusters should resort to other methods first. Presently, only three-cell partitions have been reviewed, though adjusting the model to fit up to N clusters is technically trivial.

3.5 Inference and estimation

CCT models are now often analyzed in the Bayesian framework. Beyond a model likelihood, this requires further specification of model priors and sometimes special methods for parameter inference. For example, Karabatsos and Batchelder (2003) provide a Markov Chain Monte Carlo (MCMC) methods algorithm for Bayesian inference on the GCM, Butts (2003) details the steps for directed CCT on digraphs, and Agrawal and Batchelder (2012) describe an algorithm for CCT on graphs with a two-cell consensus partition. We follow this newer tradition by using MCMC methods to estimate posterior distributions of the model parameters, given collected data.

We use the Metropolis-Hastings (M-H) and Gibbs algorithms to estimate statistics for parameters of the model. When combined, Jackman (2009) calls theses algorithms "Metropolis-

Hastings-within-Gibbs" (MH-within-Gibbs) hybrid algorithms. These work by using a computer program to iteratively sample from the joint posterior distributions of the parameters by sampling from more easily sampled conditional distributions.

Bayes' rule states that the joint posterior probability of observing the data given the parameters is proportional to the product of the probability of observing the data given the model parameters and the probability of those parameters. Let $\mathbf{v}$ and $\mathbf{v}^*$ representing a generic vector of model parameters, and X the data. Bayes' rule gives $\Pr(\mathbf{v}|X) = \frac{\Pr(X|\mathbf{v})\Pr(\mathbf{v})}{\Pr(X)}$. The MH-within-Gibbs algorithm iteratively draws random samples from the posterior distribution, $\Pr(\mathbf{v}|X)$, by comparing the last sampled parameter vector, $\mathbf{v}$, with a candidate sample parameter vector, $\mathbf{v}^*$. More specifically, the algorithm uses the ratio of $r = \frac{\Pr(\mathbf{v}^*|X)}{\Pr(\mathbf{v}|X)}$ as part of determining whether candidate parameter vectors are to be included in the accumulated samples or not. Fortunately, one does not need to solve for the normalizing constant $\Pr(X)$ when using the Bayes' rule substitution because it cancels out in the ratio yielding:

$$r = \frac{\Pr(X|\mathbf{v}^*)\Pr(\mathbf{v}^*)}{\Pr(X|\mathbf{v})\Pr(\mathbf{v})} \tag{3.23}$$

The algorithm can be implemented because the remaining likelihood and priors are straightforward to evaluate for data and any two given parameter vectors.

Another aspect of the the MH-within-Gibbs algorithm is that the initial parameter vector must be chosen such that it does not produce a zero likelihood or come from unsupported space in the joint prior. That is, $\Pr(\mathbf{v}|X) \neq 0$ must be the case for an initial choice of $\mathbf{v}$. Initial parameter vectors must be randomly chosen, but do not need to satisfy any strict distribution requirements. Each run of the algorithm over a sequence of iterations of samples is called a *chain*. Theoretically, each chain's samples are from the posterior distribution *in the long run.* Practically, the chains may not well represent the posterior distribution within a finite time, and researchers rely on *multiple* chains that use different random initial parameter vectors and then both (a) assess how converged the chains are and (b) combine samples from all chains into one pool of samples.

General purpose software for Bayesian model estimation based on the Gibbs algorithm (among others) exist and are useful for continuous and mixture models (e.g. BUGS, WinBUGS, OpenBUGS, JAGS, and Stan).[1] However, these softwares do not lend themselves to efficient sampling for the constrained discrete parameters of the NR models. Therefore, the author wrote a custom framework for implementing MH-within-Gibbs algorithms in the R statistical computing language along with implementations for the UNR and K3NR models.[2]

3.5.1 NR models estimation

The MH-within-Gibbs algorithm breaks the problem of sampling from the joint posterior distribution into separate steps of sampling from conditional distributions. Each of these makes use of candidate generator distributions that are much easier to sample from than the joint. In addition to the likelihood and priors already described, the algorithm needs the probability of drawing a candidate given the parameters of the previous iteration and the reverse probability of drawing the previous iteration when starting with the candidate. Since the Metropolis, Metropolis-Hastings, Gibbs, and MH-within-Gibbs algorithms are well covered in other texts (e.g. Gelman et al., 2004; Jackman, 2009), we choose to highlight a few of the more interesting aspects of the steps used for the NR models. We will focus on initialization and sampling steps for the unconstrained consensus graph $\mathbf{Z}$, the partition-constrained

[1] See Lunn et al. (2000) for a review of BUGS-type software and Stan Development Team (2014) for the manual of a newer probabilistic programming language called Stan.

[2] The framework is available as a set of R code with PDF documentation from the author. Interested readers can download or request the framework to aid in writing their own MCMC samplers. Latest code and documentation will be posted online at `http://sites.uci.edu/kalinagrawal/`, `http://kalinagrawal.com`, or `http://www.socsci.uci.edu/~kagrawal/`. Contact the author by email at `kalin.agrawal@gmail.com`, `kagrawal@uci.edu`, or `kalin_agrawal@alumni.brown.edu`.

consensus $\mathbf{W}$, and the binary knowledge $\mathbf{K}$. Sampling for the continuous parameters is straightforward and follows standard procedures outlined in the previously-mentioned texts.

Canonical order of the partition vector

During sampling of the K3NR model, the partition vector is relabeled at all times using a canonical ordering to eliminate “label switching” issues of redundant coding of partitions. Under the canonical format, for nodes $\{1, \ldots, N\}$ and cell labels $\{1, \ldots, K\}$, nodes are labeled sequentially using cell labels sequentially as well. That is, given a partition vector, lowest cell labels are first encountered before any higher cell numbers when reading the nodes sequentially.

Initializing consensus for UNR and K3NR

Since knowledge vectors, $\mathbf{K}$, compatible with any consensus and any response data can always be found (e.g. $K_{i,k} = 0, \ \forall i, k$), we begin by initializing the consensus parameters before the knowledge parameters.

For the UNR, the first step is to initialize the consensus graph $\mathbf{Z}$. Since the consensus is unconstrained and is conditional only on the probability of a tie being positive (p_Z); this consists of straightforward draws from Bernoulli trials.

The K3NR requires first generating a consensus partition, $\mathbf{W}$, of the N nodes with a guarantee of three cells represented. This is accomplished by first assigning the three cells to three randomly chosen nodes followed by random cell assignment for the remaining nodes. The corresponding consensus graph, $\mathbf{Z}$, is computed based on $\mathbf{W}$ using (3.22).

Initializing knowledge, given the consensus graph for UNR and K3NR

Since the knowledge layer $\mathbf{K}$ does not directly depend on the partition $\mathbf{W}$ in the K3NR model, initialization for $\mathbf{K}$ is shared by the UNR and K3NR models. Recall that the response model is the same for both and only takes into account the consensus graph $\mathbf{Z}$. Given the potential for zero-likelihood situations for either of these models, the algorithm must also start with a set of parameters (initialization conditions) that are also non-zero-likelihood.

Initializing $\mathbf{K}$ assumes $\mathbf{Z}$ has already been initialized and conditions on it. Knowledge vectors are determined for each expert separately such that no expert's knowledge vector causes conflict between their responses and the consensus condition.

The first step is to determine which *ties* an expert must *not* know given the initial consensus and the expert's responses. This is accomplished by identifying all responses that do not match the consensus.[3] For an expert, a *must-not-know* graph, $\mathbf{C}_i$, is computed, where

$$C_{i,jk} = \begin{cases} 1 & \text{if } X_{i,jk} \neq Z_{jk}, \\ 0 & \text{if } X_{i,jk} = Z_{jk} \ . \end{cases} \tag{3.24}$$

The key point in determining a knowledge vector that satisfies the responses and consensus is that two "known" nodes cannot be set for a tie with a response different from the consensus. And only through such *must-not-know* ties will nodal knowledges be constrained. The next step is to identify the component sets of nodes in the must-not-know graph $\mathbf{C}_i$.

[3] The implementation also allows for "must know" ties for an expert, though this is not actually necessary for the initialization step and is beyond the scope of this discussion.

This algorithm then chooses a single *seed* node of potential knowledge for each of the components of $\mathbf{C}_i$. These seed nodes are randomly assigned knowledge values.[4]

Using the must-not-know graph $\mathbf{C}_i$, a Depth-First-Search (DFS) (e.g. Cormen et al., 2009) is performed on each component, beginning with visiting each seed. As the search follows must-no-know ties to unvisited nodes, the node knowledge are either assigned by logical necessity or randomly[5] otherwise. Logical necessity dictates, for example, that a node by following a must-not-know tie from a known node cannot also be known (due to the conjunctive response model of Axiom 3.5 and equation 3.6).

Once all nodes for expert i are visited, $\mathbf{K}_i$ is initialized for that expert and the algorithm moves on to initialize any remaining experts' knowledge vectors. The result of initialization is a randomized $\mathbf{K}$ that does not violate constraints based on the responses, the model, and the given consensus.

Sampling the consensus graph given knowledge and data for UNR

We now turn to the sampling algorithm for the consensus graph in the UNR model. During sampling, a proposed modification to the previous iteration's consensus is made and considered for acceptance as a sample. The consensus is constrained by the expert knowledges and responses.

Must-be-known ties are any ties for which *any* expert has both nodes "known"; i.e. tie $\{j, k\}$ is a "must-be-known" tie if $\exists\ i | K_{i,j} = 1$ and $K_{i,k} = 1$. Since the sampler only proposes changes to the consensus, any *must-be-known* ties cannot be changed in the current iteration without zeroing the likelihood.

[4] I.e. a fair coin tossed for whether $K_{i,k} = 1$ or $K_{i,k} = 0$ for each seed.

[5] "Randomly", here, again means a fair coin toss for whether $K_{i,k} = 1$ or $K_{i,k} = 0$.

For each sampler iteration, the UNR consensus sampler works by identifying one tie in $\mathbf{Z}$ that is *not must-be-known* (i.e. that *nobody* has any knowledge of) and proposing that its consensus value be flipped. If no ties are eligible, then the proposal is the same as the original, and is accepted anyway.

Recall (3.6) and that the only probabilities that affect the sampling are the prior tie probabilities. The responses — in particular the knowledge model represented by $\mathbf{K}$ — has no influence on the sampling in this stage. So long as knowledges, responses, and biases are constant, the likelihoods of any two consensus values that are also compatible with the knowledges cannot be probabilistically distinguished. And any consensus vectors that are not compatible with the knowledges will have a zero (0) likelihood in any case.

The result is that when considering a proposed consensus against an existing consensus, we can only differentiate them in terms of their prior probabilities. The effect is that tie values are sampled given the probability of a positive tie, p_Z, so long as $\mathbf{Z}$ does not conflict with the currently-conditioned knowledges. The candidate generator for the UNR consensus chooses ties that are eligible for changing (recall this will not affect the likelihood) and to propose those tie values be altered.

Sampling the consensus partition given knowledge and data for K3NR

Sampling for $\mathbf{W}$ involves choosing a candidate partition that is a modification of the given previous iteration's partition. The algorithm here uses a "split and merge" technique that borrows from Ranganathan et al. (2006): a cell of a given partition is chosen to first "split" into two smaller cells, and then one of these smaller cells is merged with another cell. Unlike Ranganathan et al. (2006), which allowed the number of cells to vary through *either* splitting *or* merging at each iteration, this algorithm maintains the required $k = 3$ cells specified in the prior. The algorithm proceeds by first enumerating all possible "split and merge" changes that could be made to the cells of the given partition after taking knowledge and response constraints into account. One of these moves is chosen uniformly at random to execute and thereby create a candidate partition. At that point, the M-H-within-Gibbs algorithm computes ratios of the prior distributions to determine acceptance of the candidate, as described earlier.

Without the nodal knowledge layer, the split-merge procedure is straightforward: 1) find cells that have two or more nodes, 2) compute the number of ways to partition each such cell into two non-empty cells, and 3) for each of these possible splits, there are $2(k-1)$ ways to merge (one of) those into remaining cells.

The knowledge layer, however, requires certain nodes to either remain in the same cell or remain in separate cells which reduces the number of possible moves just described. Consider two nodes that an expert "knows" to have a positive tie based on the conditional $\mathbf{K}$ and given data $\mathbf{X}$. These must be in the same cell (because positive ties only occur within a cell) in any proposed changes to the partition. This requirement constrains the candidate partitions. Likewise, experts' knowledge of a negative tie will limit which cells can be merged since certain nodes must remain in separate cells.

The knowledge-constrained split-merge partition exploration algorithm is thus:

1. Determine a graph, $\mathbf{P}$, that has ties between nodes where at least one expert "knows" the tie and that tie is positive in $\mathbf{X}$.
2. Determine a graph, $\mathbf{Q}$, that has ties between nodes where at least one expert "knows" the tie and that tie is negative in $\mathbf{X}$.
3. Find all cells that have at least two (possibly-atomic) components in $\mathbf{P}$ (these necessarily never span cells).
4. For each $\mathbf{P}$-component within these cells, count the number of other cells that it could be moved to without violating ties in $\mathbf{Q}$ (i.e. without merging nodes that have a $\mathbf{Q}$-tie).

One of the remaining split-merge moves is chosen and a candidate partition is generated and evaluated and forward and reverse transition probabilities are calculated to determine an acceptance probability.

Sampling knowledge, given the consensus graph for UNR and K3NR

Recall that the response model for K3NR does not condition on the underlying partition. Since the M-H-within-Gibbs algorithm is based on conditional sampling, this step (as with others) regards other parameters as fixed. Since we regard the the consensus graph $\mathbf{Z}$ as fixed, the UNR and K3NR can share the same sampling procedure for experts' knowledge vectors.

During sampling, for each expert's $\mathbf{K}_i$ vector, we consider whether to increase or decrease the amount of knowledge they have, as measured by the number of known nodes. That is, one $K_{i,k}$ in $\mathbf{K}_i$ will be proposed for changing from 0 to 1 or from 1 to 0. Not all additions of knowledge are possible, given the other parameters and data, but we may always consider a subtraction of knowledge when the expert has at least one known node. This is because the likelihood (3.6) will always be non-zero for an expert even if setting all $K_{i,k} = 0, \forall k$. Indeed any given data and parameters can always be accounted for by guessing, by letting $K_{i,k} = 0, \forall k$.

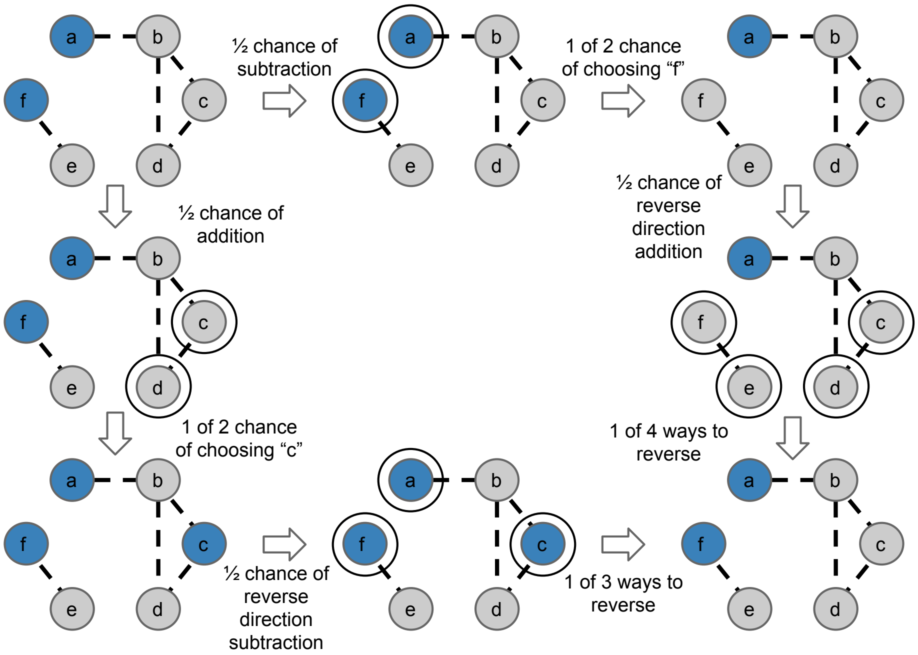
This property of allowing all $K_{i,k} = 0$ with a non-zero likelihood means the sampler can reach all possible configurations for $\mathbf{K}$ for which there is a non-zero likelihood. There are no "islands" of configurations of $\mathbf{K}$ separated by zero-likelihood configurations, and this is essential to the M-H-within-Gibbs algorithm.

Choosing nodes in a $\mathbf{K}_i$ vector that do not zero the likelihood is equivalent to exploring independent sets of nodes on a graph of *must-not-know* ties. An *independent set* of nodes on a graph is a set of nodes that are non-adjacent on the graph. For a given expert, we construct a must-not-know graph, $\mathbf{C}_i$ as in (3.24), where ties in $\mathbf{C}_i$ correspond to ties for which the expert's response does not match the consensus. The nodes for which $K_{i,k} = 1$ in vector $\mathbf{K}_i$ constitute an independent set on $\mathbf{C}_i$ when the likelihood is non-zero.

Rather than drawing independent samples of independent sets (a hard problem), we use an incremental conditional sampler that uses an add-or-subtract method. Subtracting knowledge can always be done to yield another independent set. The sampler may also make a small step of adding one node of knowledge such that the added node is not adjacent to another "known" node in $\mathbf{C}_i$. There are times when one (or both) of the steps may not be possible in which case the sampler simply "stays put".

For either a proposed removal of knowledge or a proposed addition of knowledge, the ratio of probabilities of transitioning in both directions (i.e. from original to candidate and candidate back to original) needs to be calculated. Therefore a set of possible nodes that can have knowledge added needs to be determined for either a proposed addition or subtraction.

Figure 3.1 shows examples of generating a knowledge addition and subtraction given a $\mathbf{K}_i$ configuration. In general, for the add-or-subtract sampler, a coin toss determines whether an addition or subtraction will be attempted. If addition is chosen, then one of the nodes that could possibly be added is chosen with uniform probability. The candidate knowledge vector is then generated as $\mathbf{K}_i^*$ with a forward transition probability of 0.5 divided by the number of possible nodes that could have been added as "known". Given a candidate coming from a single node addition, there is only one subtraction that could reverse the process. Therefore, the reverse probability comes from the coin flip's 0.5 probability divided by the number of known nodes that could be subtracted from $\mathbf{K}_i^*$. In practice, the coin flip probabilities of 0.5 in either direction cancel out when used in the M-H-within-Gibbs algorithm. The same sort of calculation is carried out in the case that subtraction is randomly chosen for generating a candidate.



a
b
½ chance of subtraction
1 of 2 chance of choosing "f"
f
c
e
d
½ chance of addition
½ chance of reverse direction addition
1 of 2 chance of choosing "c"
1 of 4 ways to reverse
½ chance of reverse direction subtraction
1 of 3 ways to reverse

Figure 3.1: Two examples of candidates for sampling $\mathbf{K}_i$ (a potential candidate for both adding and subtracting knowledge are shown). Unknown nodes are in gray, known nodes in blue. Potential nodes for addition or subtraction are circled. The diagram begins in the top-left. A sampling step starts with a knowledge configuration for expert i of known nodes "a" and "f" (in blue) and a *must-not-know* graph for expert i (dashed lines) in the top-left corner. The sampler has a 1/2 chance of either adding (moving down along left) or subtracting (moving right along top) knowledge. If addition is chosen, a candidate is chosen (bottom-left), and then the probability of returning back to original state (bottom-right) is determined. In the addition step shown, the probability of adding knowledge "c" is $(1/2)(1/2) = 1/4$ and the reverse probability is $(1/2)(1/3) = 1/6$. If subtraction is chosen, then the probability of subtracting "f" from original knowledge is $(1/2)(1/2) = 1/4$ and the reverse probability of adding back "f" would be $(1/2)(1/4) = 1/8$. In practice, at each iteration for expert i, the initial configuration (top-left) is given and only one of either the top-right or bottom-left candidates is generated for consideration; though both backwards and forwards transition probabilities are calculated to determine the sampler's probability of accepting the candidate.

3.6 Simulation studies

3.6.1 Bayesian Software Validation

Since the sampler algorithms used here for estimation are custom designed and written by the author without the benefit of a software development team or distributed quality assurance as found in Open Source Software (e.g. the base R program and and many of its packages), it is recommended that implementations of the algorithms be validated using automated validation techniques.

One method is to use the model priors and hyperpriors to repeatedly draw simulated data using sampled (and recorded) parameter values and then to compare those with posterior estimates of those parameter values.

The intuition is that the posterior estimates for parameters should approximate the generating parameter values, albeit with random noise. As Cook and Gelman (2006) point out: when using valid software, "the resulting posterior inferences will be, on average, correct." and that "50% and 95% posterior intervals [of the estimated parameters] will contain the true parameter values with probability 0.5 and 0.95, respectively." The key issue with complex hierarchical models is that the expected random error associated with recovery is not intuitive nor easy to determine analytically. Cook and Gelman (2006) rightfully point out that an author of estimation algorithms may lack incentive, ability, or resources to additionally commit to testing their model estimation software for validity. Some have provided software tools for such general-purpose testing of inference algorithms to reduce the difficulty in performing this due diligence.

We ran Bayesian Software Validation (Cook and Gelman, 2006) on the UNR and K3NR models, as implemented in an R package called BayesValidate (Cook, 2006). Twenty sets of parameters were sampled and recorded along with corresponding twenty sets of simulated data of 12 experts and 6 nodes (15 ties) for each of the response models. The program then applied the author's respective inference algorithms to each of these simulated datasets to obtain posterior samples (and statistics) of model parameters. BayesValidate takes care of keeping track of the simulation parameters and the posterior samples.

One output of BayesValidate is an "exploratory" plot of z statistic absolute values that indicate how different the posterior estimate for a parameter (or block of parameters) is from the parameter value(s) used for generating the data. Figures 3.14 and 3.15 show the plots of these statistics produced by the program. This software is applicable toward continuous parameters, so the $\mathbf{Z}$, $\mathbf{W}$ and $\mathbf{K}$ parameters can be ignored. We look for where absolute value of z statistics are within 2 (details can be found in the accompanying articles). Based on reviewing these outputs, we may wonder if the σ parameter is correctly programmed since it has absolute value z statistic near 2 for the UNR model.

Concern over the σ parameter is confirmed by anecdotal reviews of posterior sampling distributions when looking at real data. The distributions were sometimes truncated by the upper bound of the prior distribution for σ, occasionally showing a "pile up" near the upper limit. Hence the prior distribution may be misspecified or, more likely, the sampling methods may not be efficient. Exploring these issues is left for future work.

3.6.2 Other simulation studies

Artificial data designed with known expert abilities and/or node difficulties recover the generating patterns well. Importantly, all simulated datasets showed perfect recovery of consensus values in the posterior modal distributions. Traceplots of the sampler values show there is very good mixing in the $\mathbf{K}$ parameter which bodes well for the custom discrete candidate generator.

Working with early version of the NR models and algorithms, traceplots of real data indicated issues with convergence of sampling chains. The issue was especially apparent when looking at the posterior distributions of the binary Z_{jk} parameters. Some chains indicated a posterior mass of Z_{jk} on a single value (i.e. all on either 0 or 1). In some cases, not all the chains agreed on which value all the posterior mass should be placed for a given tie. If, having run four chains, one chain's mass was all on 0, and the other three chains' masses were all on 1 for a tie, then the standard practice of merging the chains' samples would result in a posterior mean of precisely 3/4. The fluke for such binary parameters is that the ratio of *chains* settled onto one value will very nearly determine the posterior point estimate for that parameter. This highlights a weakness in the discrete sampling: such algorithms can force the distributions to vary significantly across chains, yet the number of chains is typically kept low (compared to the number of iterations per chain).

Based on these initial simulation studies, the author made two notable changes to the initial versions of the model and sampling algorithm. First, the original implementation of the UNR algorithm proposed changing one consensus tie at a time for each iteration when generating candidates. The current algorithm proposes changing a *minimum* of one consensus tie value (if possible) and up to as many that could possibly change. The adjustment was geared towards allowing better mixing in the posterior consensus exploration for the UNR. Simulation studies on artificial, similarly-sized datasets indicate that continuous parameter mixing and convergence improved (i.e. $\hat{R}$ values were consistently close to 1.0).

The second change was to adjust the model to separate the parameters for ability and difficulty scales, as the model currently specifies. A shared scale parameter between abilities and difficulties would assume the population of nodes was the same as the experts, which is unrealistic. Prior to separating the two scale parameters, the sampler chains were not mixing due to some chains being "stuck" in local maxima which also affected the recovery of consensus values.

We created several datasets based on simulated responses with certain characteristics that should show up in the posterior estimates. The simulations used $M = 20$ experts and $N = 6$ nodes (15 ties). Under conditions where experts' responses exactly match a consensus (i.e. they all respond exactly the same), we find all of the expected patterns for posterior estimates. Difficulty parameter estimates pushed the lower bounds of the finite interval since the items were infinitely easy. The associated scale parameter stretched to its finite bound as well, perhaps somewhat accommodating the infinitely easy ties and nodes. The posterior recovery of the consensus was perfect and $\hat{R}$ values were low for all continuous parameters.

With ties simulated as progressively more difficult starting with those adjacent to node 1 and increasing difficulty through node 6, we find that the estimated node difficulty increases. That is, node 1 is correctly identified as the "easiest" node and node 6 is identified as the "hardest". Correspondingly, the posterior consensus Z_{jk} estimates for ties with node 1 are generally well-recovered, though confident ground truth recovery gets very poor for the higher-numbered nodes. The convergences of chains measured by $\hat{R}$ was also good (low).

Thinning the samples used for parameter estimation can reduce autocorrelation in the resulting chains which otherwise can lead to poor mixing and convergence. No thinning corresponds to a thinning value of 1, which retains every sample. Simulations showed that thinning of 4 (i.e. retaining every fourth sample) sufficiently reduced the $\hat{R}$ values and was used in subsequent analyses.

3.7 Application

In these subsections we apply the UNR and K3NR to two datasets. The data and descriptive statistics of the data are given first followed by results of the model estimation.

3.7.1 Data

Two datasets were collected by issuing surveys (questionnaires) to undergraduate students in an upper-division psychology course at University of California - Irvine in the Winter Quarter of 2015. The two surveys addressed, respectively, inherently tie-based judgments and inherently pairwise comparisons of nodes.

One dataset, a *Celebrities* questionnaire, asked experts ($n = 38$) about whether pairs of named celebrities appeared as coaches or judges on the same singing competition show on network television. Response options were forced-choice "yes (same show)" or "no (different shows)". Three judges from each of three different shows were chosen as the nodes to query experts about, focusing on a common time frame of 2013-2014:

- *The X-Factor*: Season 3, September 11, 2013 – December 19, 2013
  - Simon Cowell
  - Demi Lovato
  - Kelly Rowland
- *The Voice*: Season 6, February 24, 2014 – May 20, 2014
  - Adam Levine
  - Shakira
  - Usher
- *American Idol*: Season 13, January 15, 2014 – present
  - Jennifer Lopez
  - Keith Urban
  - Harry Connick, Jr.

All 36 pairs of the nine judges were presented using the survey design described in Section 2.6 of the previous chapter. The survey design mitigates some of the potential for an expert to use logical inference to respond rather than their actual knowledge. The shows on which the celebrities participated provided a ground truth partition that can be used to validate the model's recovery of latent consensus partitions. The survey design also randomized the inherently directed question phrasing as well as the order of pair presentation. The response forms always had "yes (same show)" as the left-hand response option and "no (different shows)" as the right-hand response option.

The second dataset, *Borders*, provided the same format of data for analysis as the Celebrities data, but the questions and node set were different. The Borders questionnaire was issued to different (disjoint) set of students in the same class where Celebrities data was collected. The Borders questions were tie-based, asking whether each pair of nine states "share a land border?" Response options were forced-choice "yes (they border)" or "no (they do not border)". The states are those discussed in Chapter 2 and include one more. As with the first dataset, all 36 pairs on nine nodes were presented using the same survey design. They are states in the western USA:

- Arizona
- California
- Idaho
- Montana
- Nevada
- Oregon
- Utah
- Washington
- Wyoming

To provide a general idea of the performance and item difficulty, we present the *true positive rate* (TPR, or *hit rate*) and *false positive rate* (FPR) distributions in Figures 3.2 and 3.3. The FPR are plotted against the TPR in Figure 3.3. Right away, it is apparent that the 2HT assumption of TPR being greater than FPR for experts is present in the Borders data but absent in the Celebrities data. We should expect poor recovery of the consensus from the Celebrities data. The overall performance (rate correct of responses) for Borders is 0.715 and for Celebrities is 0.682, also shown in Table 3.1.

We expect that the two datasets will differ in their inherent structure given our (researcher's) knowledge of the ground truth. The Celebrities data should hint at the three cells of TV show participation while the Borders should show none. One way to visualize this is to look at the summed adjacency matrices across experts, with rearranged columns and rows highlighting the block structure. See Figure 3.4 and 3.5. The Borders data shows an imperfect single block structure that "fades" away into noise, typical unstructured data. The Celebrities data, however, shows three distinct highly-connected blocks with few positive ties between them.

Dataset	Overall TPR	Overall FPR	Correct	UNR TPR	UNR FPR	UNR Correct	K3NR TPR	K3NR FPR	K3NR Correct
Borders	0.276	0.12	0.715	0.333	0.0278	0.833	0.25	0.0833	0.722
Celebs	0.126	0.194	0.682	0.25	0.167	0.833	0.222	0.0833	0.889

Table 3.1: Overall expert performance measures for each dataset along with performance rates for each model's consensus (modal) against the ground truth.

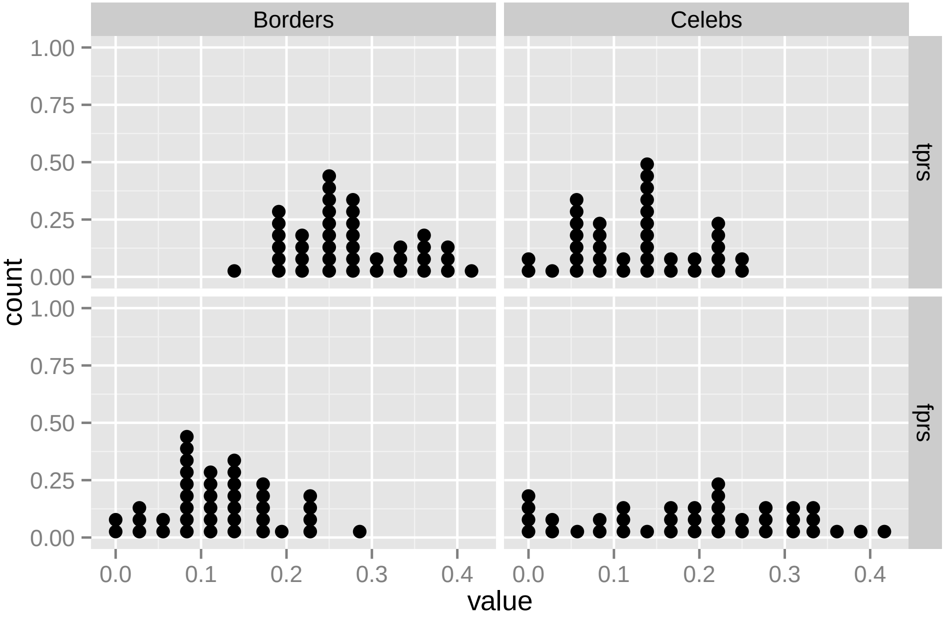


Figure 3.2: Dotplots of true positive rate (TPR, top row) and false positive rates (FPR, bottom row) for each dataset.

3.7.2 Determination of single-factor structure and sufficient signal

We begin analysis by screening the available datasets for evidence of a single-factor solution to the inter-expert response correlation and for overall sufficient signal in the data. We can safely expect a single ground truth for the questions, though experts' responses may reveal that cultural knowledge is different than the ground truth.

The UNR and K3NR models were applied to the two datasets and then postpredictive data was generated using samples from the posterior parameter distributions. Post-predictive checks help assess whether the model is appropriate for given data. Gelman et al. (2004), Gelman and Hill (2006), and Jackman (2009) discuss these procedures in more detail. Here we examine the following statistics that the model should recapture in post-predictive simulations.

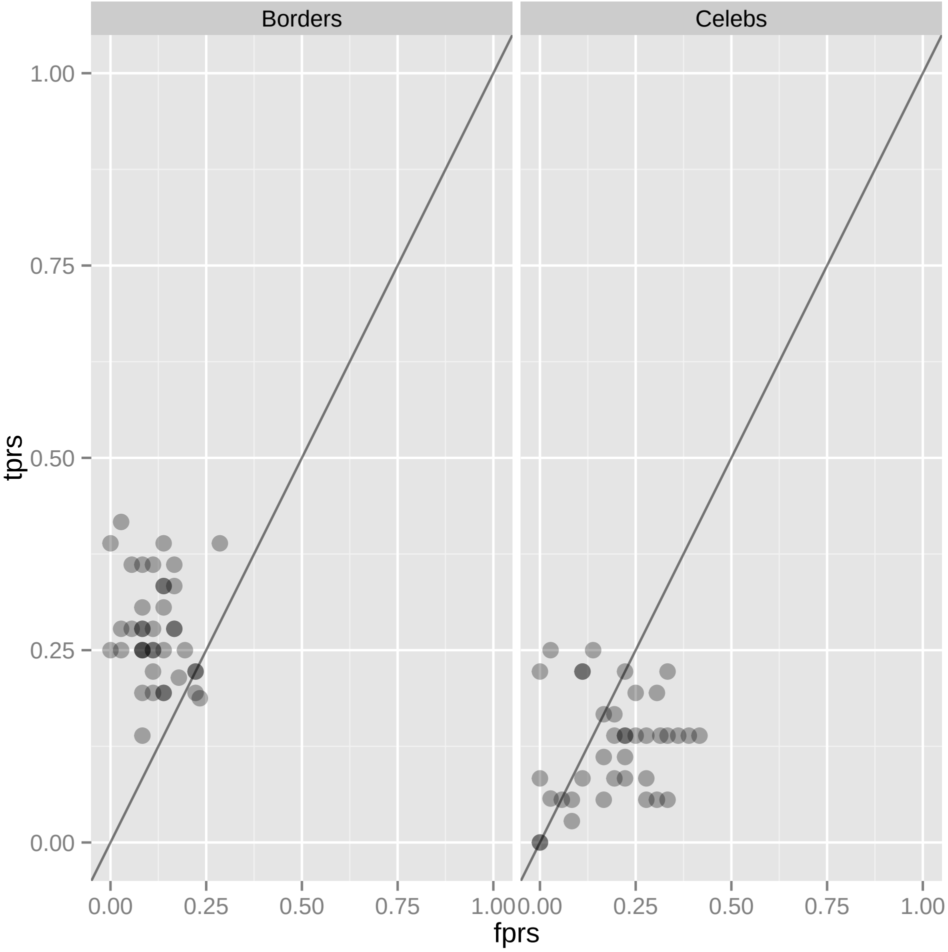


Figure 3.3: Scatterplots of true positive rate (TPR) and false positive rate (FPR) for each dataset, per expert. Darker circles represent multiple experts with the same TPR and FPR.

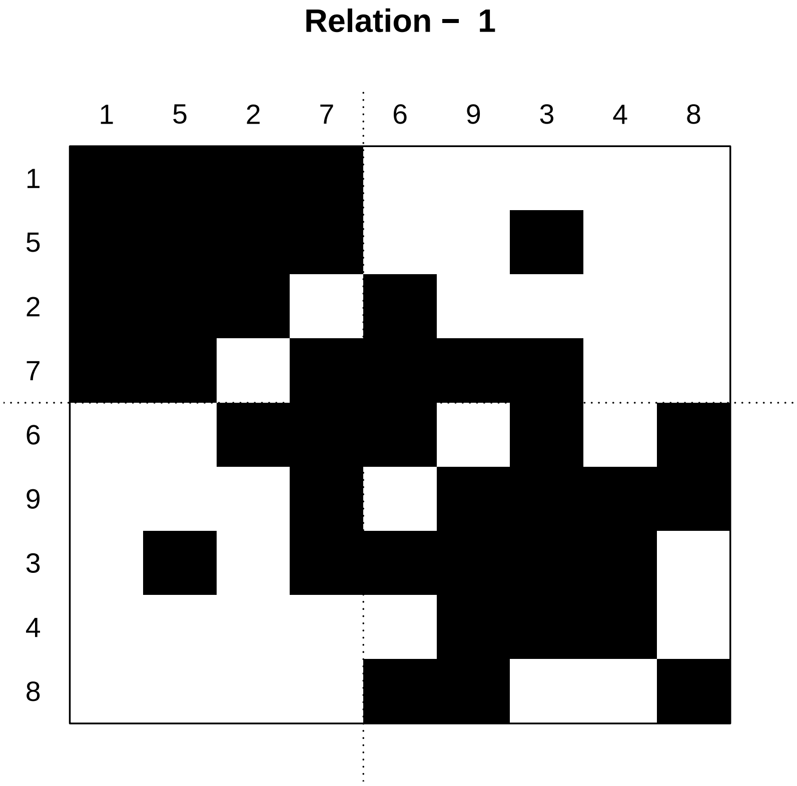


Figure 3.4: Blockmodel plot for Borders data. There is no obvious “block structure” based on mean responses for ties greater than a 0.5 threshold.

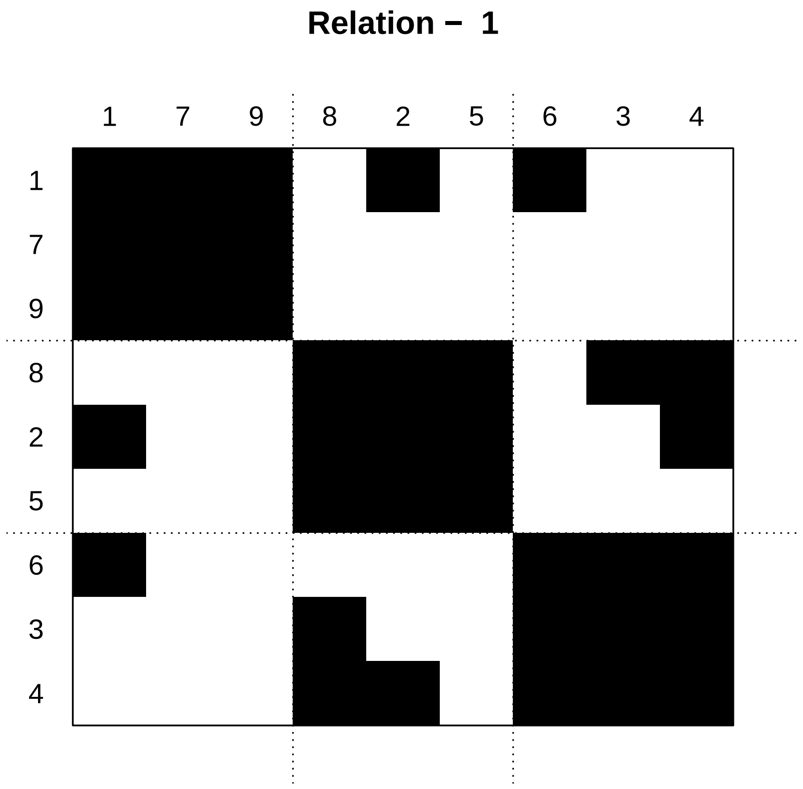


Figure 3.5: Blockmodel plot for Celebrities data. The data shows the expected three-block structure based on mean responses for ties greater than a 0.5 threshold.

Scree plot

Aside from a minor “hump” in the real eigenvalues for the Celebrities data, Figure 3.6 suggests that the Celebrities data real eigenvalues fall within the postpredictive cloud. The Borders data suggest a model mismatch where the data generated based on estimated parameters parameter indicated less of a single culture than the real data.

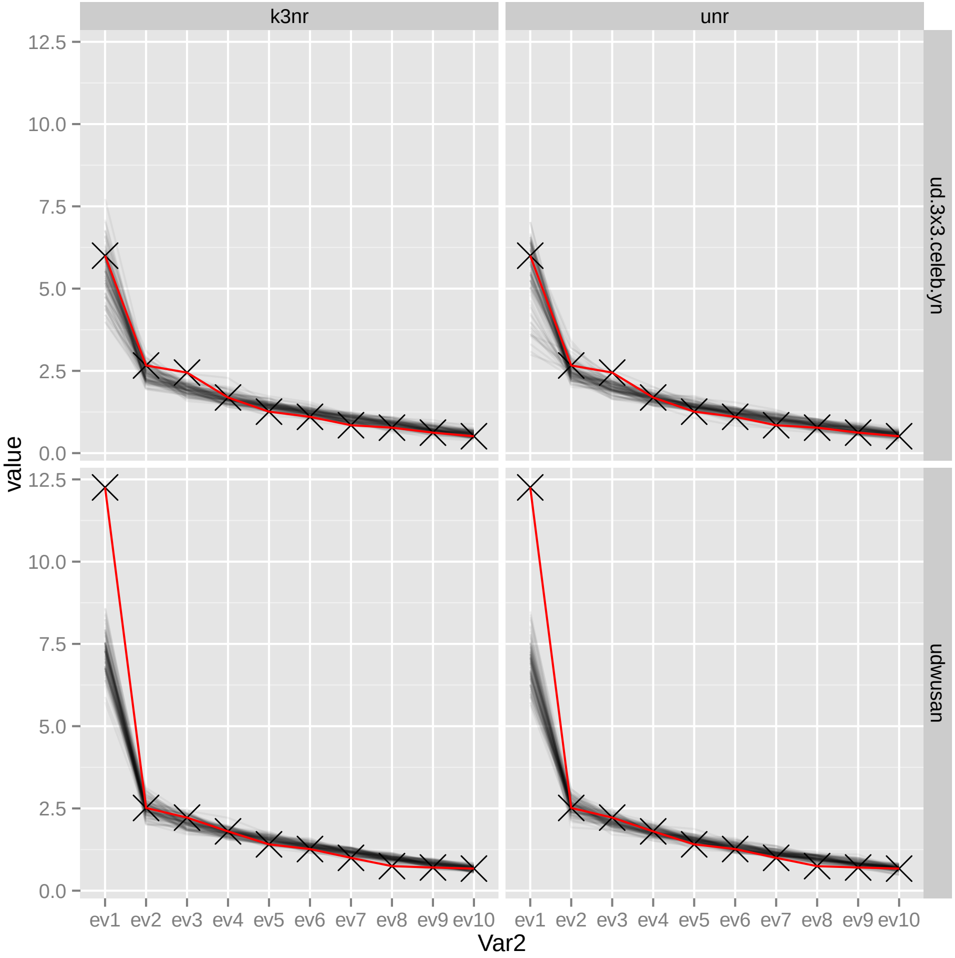


Figure 3.6: Scree clouds from postpredictive data from the UNR and K3NR models for each dataset (Celebrities in the top row, Borders in the bottom row). The red line with black “X” marks indicate the scree plot for each dataset such that column 1 and column 2 are the same within each row.

First-to-second eigenvalues ratio

We may also look at the postpredictive distribution of the ratios of the first to second eigenvalues of the expert-to-expert correlation matrices. For each of the datasets along with each of the models' postpredictive data for them, Figure 3.7 shows postpredictive distributions of these ratios using the UNR and K3NR models along with the ratio obtained from the original data. The Celebrities dataset suggests model fit for both models using this test, where model fit comes from actual ratio falling within the highest-probability-density interval. The discrepancy between the Borders data ratio and the postpredictive ratios is evident evident in these plots as well.

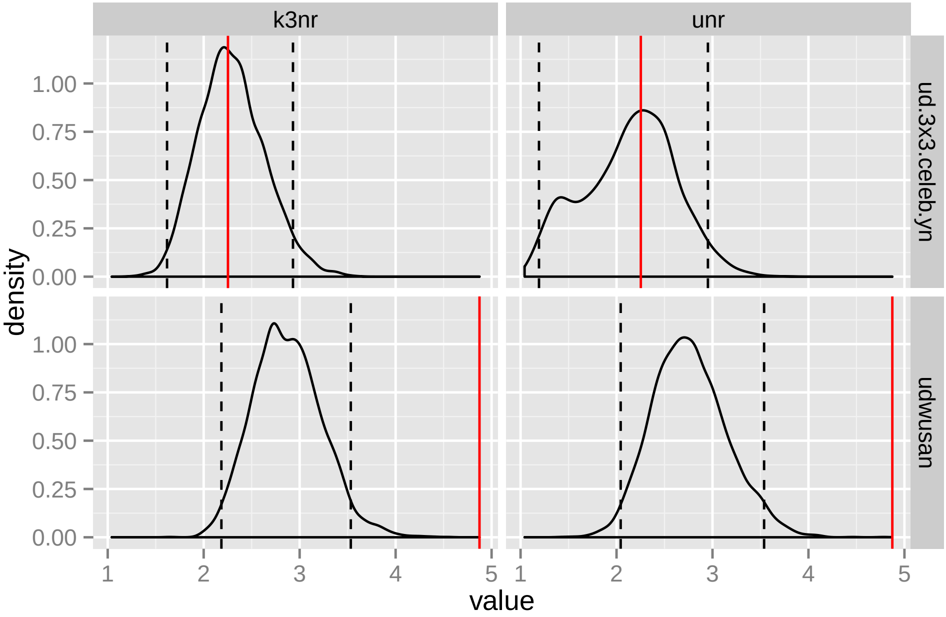


Figure 3.7: Distribution of ratios of first-to-second eigenvalues of inter-expert correlation matrices based on postpredictive data for each model and each dataset (Celebrities in the top row, Borders in the bottom row). Dashed black vertical lines indicate the 95% highest probability density (HPD) bounds. Solid red vertical lines indicate the actual data first-to-second eigenvalue ratios.

Mean posterior competence and knowledge

For the UNR model, the posterior $\bar{\mathbf{D}}$ for Borders and Celebrities are 0.53 and 0.43, respectively. For the K3NR model, the $\bar{\mathbf{D}}$ for Borders and Celebrities are 0.47 and 0.39, respectively. These numbers are the mean estimated competences with respect to experts' ability to make comparisons against a given node.

These numbers cannot be directly compared with the $\bar{\mathbf{D}}$ of the GCM since these are one level removed from the actual responses and is the mean probability that experts have for "knowing" a node in the NR sense. The same pattern shows up in the posterior distribution of the number of nodes "known" by the experts overall. The $\bar{\mathbf{K}}$ for the UNR Borders posterior is 0.51. The $\bar{\mathbf{K}}$ for the UNR Celebrities posterior is 0.39. The $\bar{\mathbf{K}}$ for the K3NR Borders posterior is 0.48. The $\bar{\mathbf{K}}$ for the K3NR Celebrities posterior is 0.39.

3.7.3 Partitionable triads

Counting triads of responses (i.e. three ties sharing the same three nodes) that cannot be partitioned into $k = 3$ cells may indicate a mis-application of the model. It would be especially worrisome if the experts responded to the Celebrities data with more un-partitionable triads than expected. Figure 3.8 shows that the postpredictive distributions of un-partitionable triad counts are all generally more than the response data itself, and there are no surprisingly high numbers of un-partitionable triads in the data.

3.7.4 Posterior analyses

As it turns out, separating the two scale parameters (used for ability and difficulty) is useful for accommodating the two different populations' ranges of values. Figure 3.9 shows the

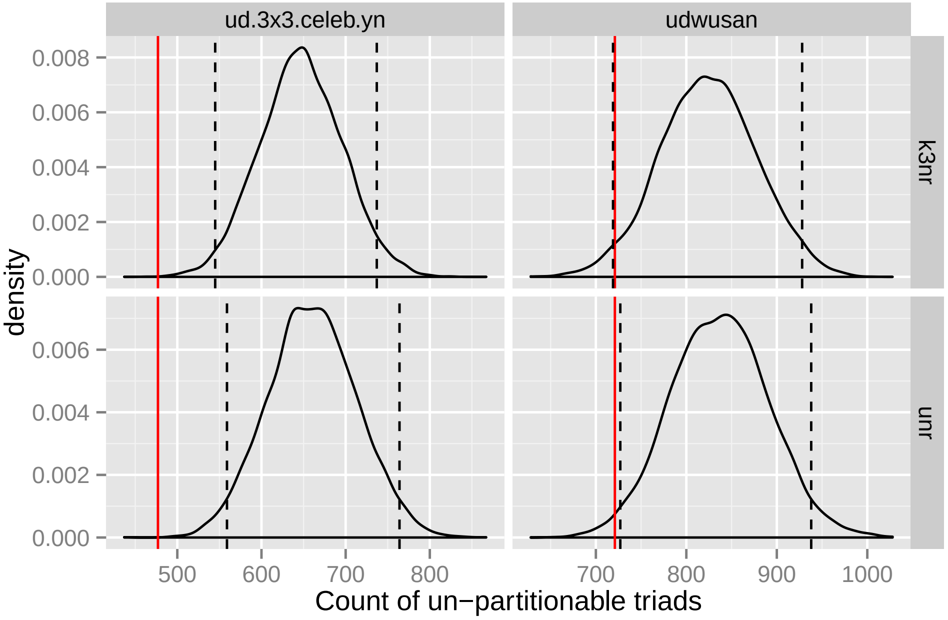


Figure 3.8: Plots of postpredictive triad counts and 95% HPD interval. Dashed black vertical lines indicate the 95% highest probability density (HPD) bounds. Solid red vertical lines indicate the statistic from the data.

posterior distributions of each scale parameter, for each model and dataset. The Celebrities nodes' difficulties had nearly the same posterior scale as the abilities of the experts. However, with the Borders dataset, there is much less variance in the experts' abilities than in the node difficulties.

Consensus estimates are plotted in Figures 3.10 and 3.11. In these plots, the marginal mean posterior Z_{jk} value is plotted for the UNR model and the modal $\mathbf{Z}$ graph is plotted for the K3NR. For the average responses and the UNR point estimates, extreme values near 0 or 1 indicate high confidence and values near the center lines indicate low confidence. As with other similar datasets, the Borders data shows a wide range of marginal average response per tie. The Celebrities dataset notably does not present any pair of nodes for which experts are in high agreement.

Considering overall performance, we expected that the K3NR partitioning model would do best with the Celebrities data and that the UNR model would do well with the Borders data

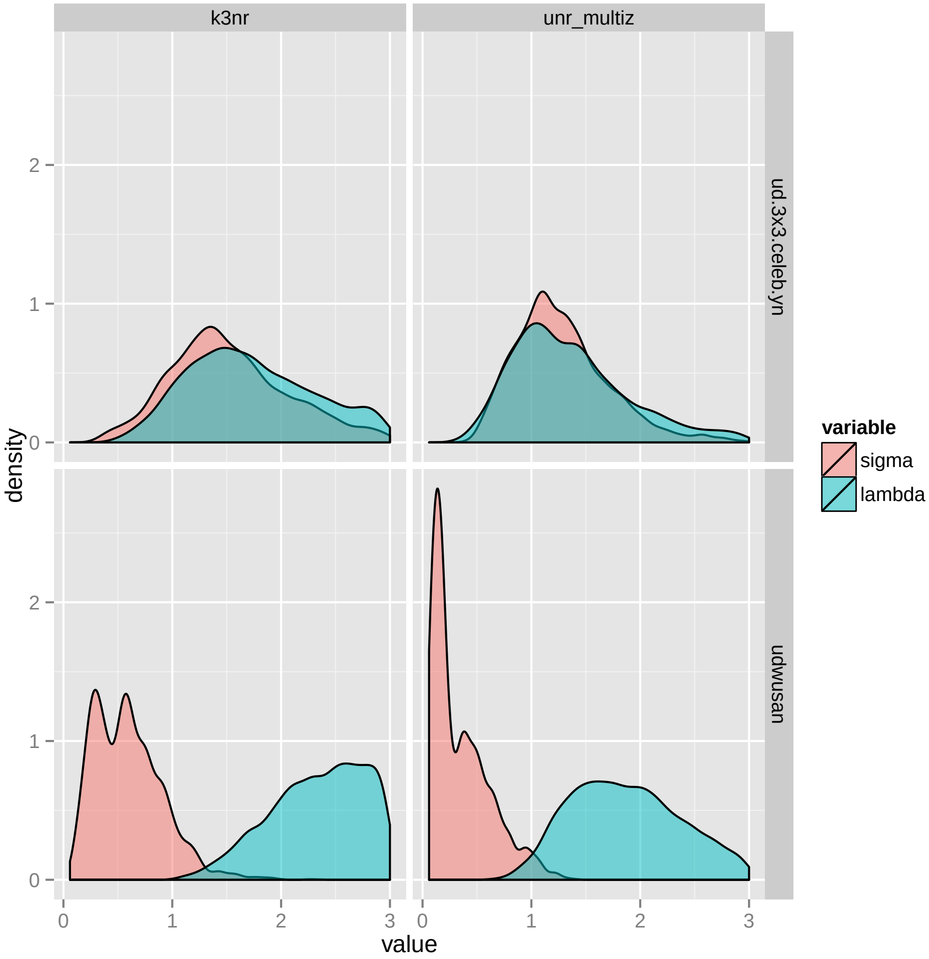


Figure 3.9: Posterior density plots of ability and difficulty scales of σ and λ, respectively.

and also to do well with Celebrities data if there is strong signal. In particular, the K3NR should perform poorly with respect to the ground truth when applied to the Borders data since there is no partition in the ground truth.

Individual estimates of posterior node difficulty are in line with expectations and presented in Tables 3.2 through 3.5. Each model resulted in similar node difficulty estimates and fulfilled expectations that California students would know the most about California geography (though Nevada seems to be an important cultural geographic marker as well).

Looking to the Borders data, the confidence in marginal posterior estimates is exaggerated compared to the raw responses and is typical of CCT model estimates. The UNR model illuminates the anticipated ease with which experts respond to questions involving California and the difficulty of the states Utah, Idaho, and Oregon. Oregon's difficulty is surprising given its true adjacency to California.

More importantly for the Borders is that the UNR did not provide strong consensus for the posterior estimates. Contrasting it with the GCM and PDGCM on similar data makes the many ambiguous posterior tie values disconcerting (See Figure 3.11).

At face value, the relative high posterior mass on the Borders partition "w4" (see Figure 3.12) suggests that a partition with cells of {AZ, CA, NV, UT}, {ID, MT, WA, WY}, and an isolated {OR} is probable. This will be challenged in discussion below.

The UNR model further emphasizes the strong cultural assumption that NV and OR do not border each ocher (when, in fact, they do). That the K3NR isolated OR as a singleton cell stems from the low knowledge of Oregon's borders coupled with the pressure of the mandated cell count. The $k = 3$ cell partition prior forces the ties adjacent to OR to be negative given the other strong border relationships.

Turning to the Celebrities data, we see that the UNR reveals a strange phenomenon where many of the marginal posterior tie values hover at 1/4. This situation is, indeed, improbable and an explanation is given below.[6] In any case, the structure in the ground truth came through in a small way: the UNR picked out the {AL, S, U} ground truth triad as an isolated clique of celebrities. The K3NR also identified this clique as a cell and resolved the separation of the other nodes into two cells – something the UNR could not do. Figure 3.12 shows the different patterns of partitions identified in the samples, placing the modal partition on the right hand side. Despite the node SC (of the X-Factor) being identified with celebrities of cell 2 in the modal partition "w8", the *marginal* modes of each W_k result in a partition that matches the ground truth (and the same as "w7" in the figure). This difference of SC's cell in this $k = 3$ cell partition corresponds to five tie differences (negations) and is the sole source of error in Figure 3.10.

With respect to estimating consensus, the constraint imposed by the K3NR model is both a strength and a weakness. Certainly, when misapplied to unpartitioned data, as is the case with Borders, the consensus graph will necessarily be flawed. For partitionable consensus graphs, however, the constraint can significantly clean up poor data to recover ground truth. Experience with authoring and analyzing the constraints of the binary knowledge layer and of the partition made clear that there are some underlying issues that make some of the posterior estimates questionable.

[6] This phenomenon is because multiple sampler chains provide the exploration of the consensus distribution instead of within a given chain. This is not desired behavior and has serious implication for the use of the results.

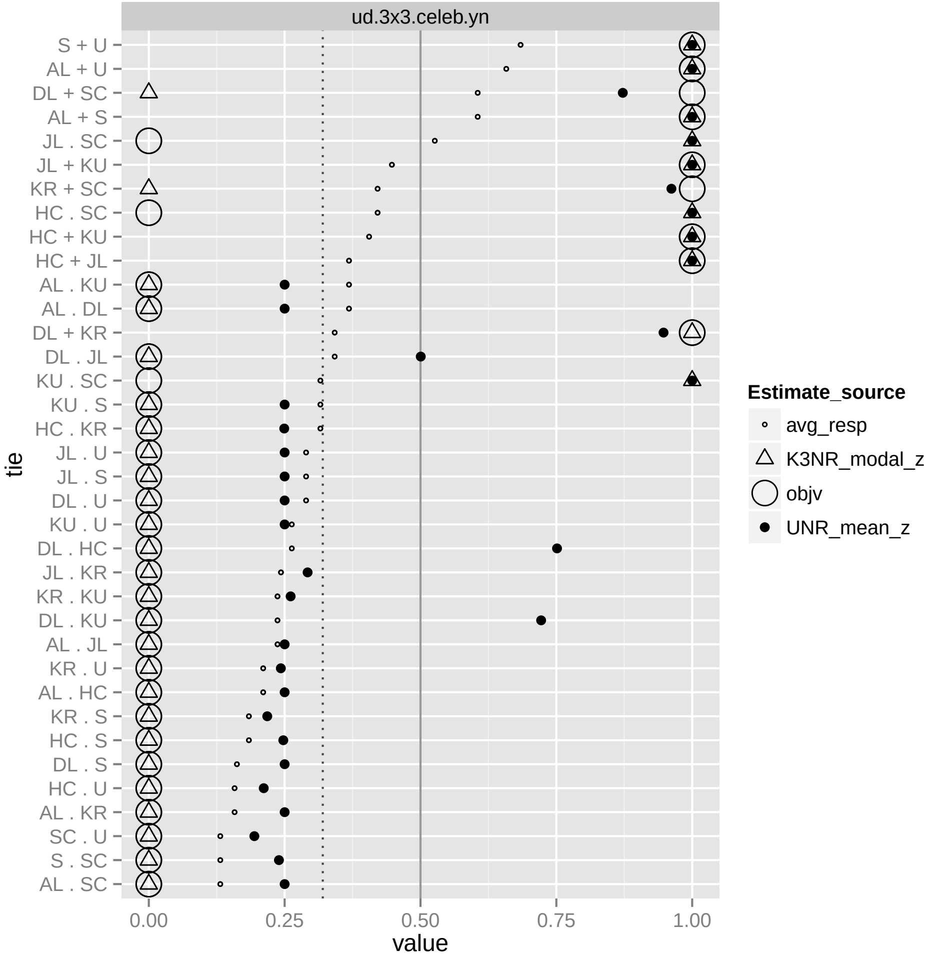


Figure 3.10: Celebrities posterior consensus: Posterior marginal mean Z_{jk} values for the UNR and posterior modal $\mathbf{Z}$ for the K3NR are presented simultaneously. The dotted line represents the overall mean response, to be compared against marginal mean responses. The solid line is at 0.5 and can be compared against posterior mean Z_{jk} values. These lines may be used for decision thresholds for tie values if a discrete posterior estimate for $\mathbf{Z}$ is desired.

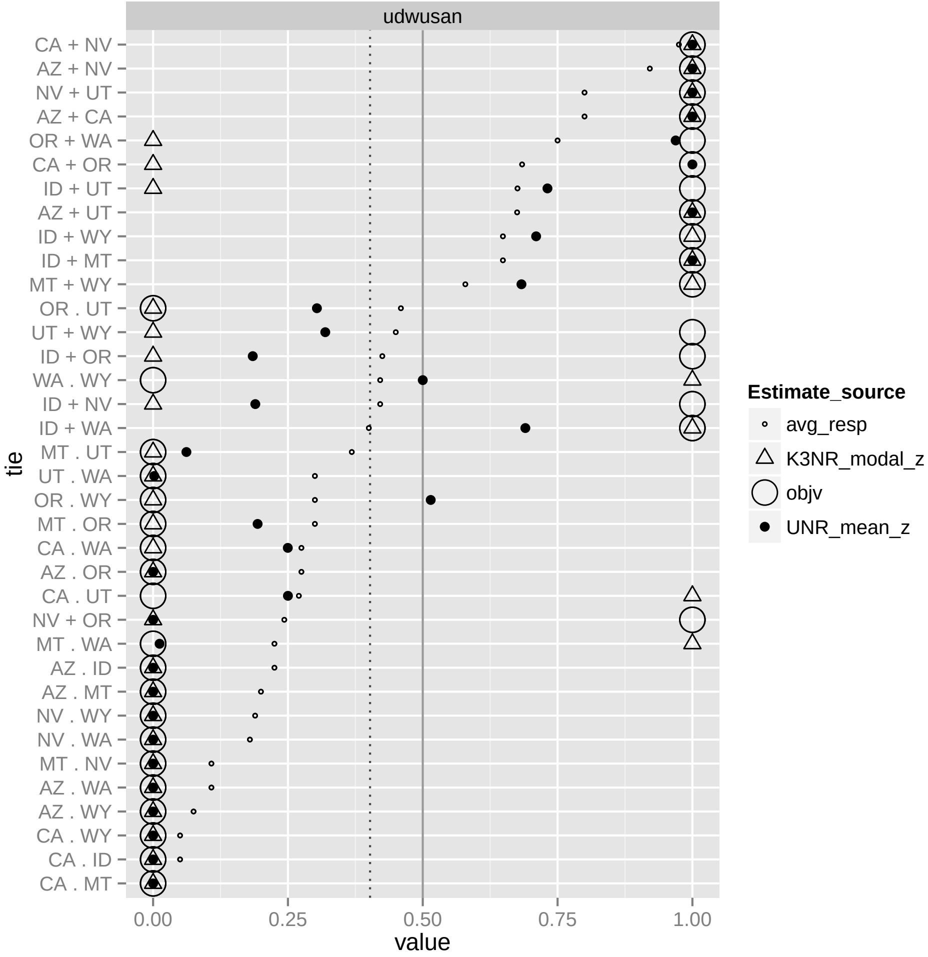


Figure 3.11: Borders posterior consensus: Posterior marginal mean Z_{jk} values for the UNR and posterior modal $\mathbf{Z}$ for the K3NR are presented simultaneously. The dotted line represents the overall mean response, to be compared against marginal mean responses. The solid line is at 0.5 and can be compared against posterior mean Z_{jk} values. These lines may be used for decision thresholds for tie values if a discrete posterior estimate for $\mathbf{Z}$ is desired.

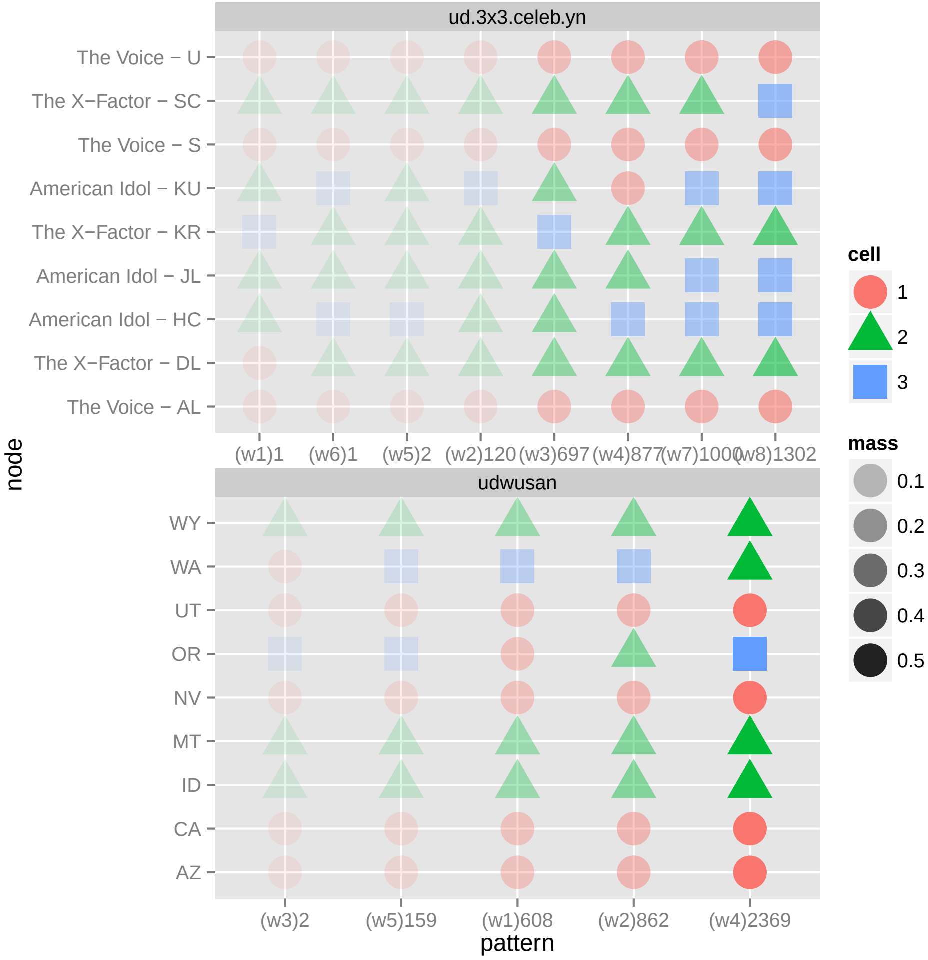


Figure 3.12: For each dataset (panel), all posterior partition patterns from the K3NR samples are shown. Each pattern is shown as a vertical column labeled with a pattern number, “w(1)”, “w(2)”, etc., followed by the frequency of that pattern in the posterior samples. The patterns are arranged in increasing frequency from left to right. Column (pattern) transparency is proportional to that pattern’s posterior mass, and so the darkest column has most mass. For each dataset, posterior cell values are assigned a color/shape symbol for ease of cell differentiation. For a given panel, the right-most column is the modal posterior partition and therefore the algorithm’s best guess at the underlying partition. Node names are on the y axis. The Celebrities dataset includes the ground truth cell identities for each node, while the Borders dataset does not have any ground truth partition. The marginal modes of the cell labels differ from the modal partition as a whole for only one node: *The X-Factor - SC* of the Celebrities nodes is in cell “2” (green triangle) under the posterior marginal modal cell.

ID	node	B
5	NV	-2.341
2	CA	-1.861
1	AZ	-1.512
4	MT	-0.1882
9	WY	0.3606
8	WA	0.7355
6	OR	1.074
3	ID	1.103
7	UT	1.419

Table 3.2: Estimated difficulty for each node of Border data using the UNR model.

ID	node	B
9	The Voice - U	-0.5773
1	The Voice - AL	-0.4198
7	The Voice - S	-0.1529
8	The X-Factor - SC	0.4126
4	American Idol - JL	1
3	American Idol - HC	1.093
6	American Idol - KU	1.124
2	The X-Factor - DL	1.662
5	The X-Factor - KR	1.746

Table 3.3: Estimated difficulty for each node of Celebrities data using the UNR model.

ID	node	B
5	NV	-3.369
2	CA	-2.337
1	AZ	-1.612
4	MT	-0.5608
9	WY	-0.06113
3	ID	1.495
7	UT	1.917
8	WA	2.493
6	OR	4.085

Table 3.4: Estimated difficulty for each node of Borders data using the K3NR model.

ID	node	B
9	The Voice - U	-0.8181
7	The Voice - S	-0.5695
1	The Voice - AL	-0.3912
8	The X-Factor - SC	0.1303
3	American Idol - HC	1.362
4	American Idol - JL	1.483
5	The X-Factor - KR	1.767
6	American Idol - KU	1.814
2	The X-Factor - DL	1.833

Table 3.5: Estimated difficulty for each node of Celebrities data using the K3NR model.

3.8 Discussion

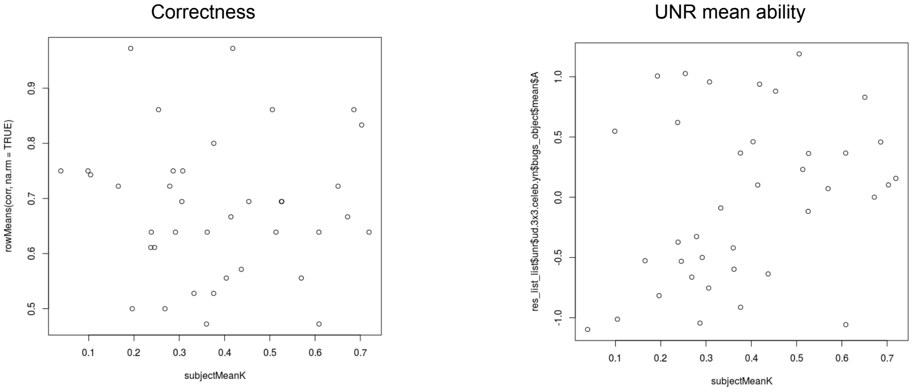
3.8.1 Benefits of the Knowledge Layer

When looking at posterior expert abilities, we see the same behavior as in Chapter 2. Two related models estimate highly correlated abilities for the experts but are not highly correlated with the expert's performance against the ground truth.

The knowledge layer acts as a selection mechanism that picks a subset of information or subset of responses that will be used for the estimation of the answer key. All other information not selected by $\mathbf{K}$ is relegated to biased guessing.

When looking at the knowledge layer's meaning as a cognitive parameter, we want to check if it correlates with some real performance measures. In this case, knowledge is compared with expert performance against the ground truth, since that has been naturally manipulated. Scatterplots of the UNR mean $\mathbf{K}_i$ estimates are plotted against the rate of correct response per expert (a) and the expert's mean estimated ability (b) are depicted in Figure 3.13. There is very little structure to the plots that would be expected if the knowledge layer indicated real-world knowledge.

A single $K_{i,k} = 1$ for a given expert does not change the likelihood of their responses from that of guessing since the hard knowledge "conjunction" requires for two nodes to be known in order to attribute the response to knowledge. In sampling, a poor-performing expert (with respect to the rest of the experts) will exhibit a "bubbling" of knowledge that itself never amounts to their responses necessarily matching the consensus. Indeed, even knowledge of two node only accounts for a single tie being attributed to true knowledge. Therefore, the $\mathbf{K}$ parameter is less telling when only one or two are turned on.



(a)

(b)

Figure 3.13: Averaged $\mathbf{K}_i$ for each expert is plotted along the horizontal axis for both plots. (a) plots averaged $\mathbf{K}_i$ versus expert performance against the ground truth. (b) plots averaged $\mathbf{K}_i$ against the expert abilities.

3.8.2 Lack of convergence

Further analysis of the samples indicates that chains tend to fixate on certain patterns for the discrete parameters. An important lesson learned from the implementation of the algorithms is that the discrete nature of the answer key on which all the latent parameters are conditioned makes estimating these very difficult. This lack of convergence seems to occur even after multiple levels of burn-in and after the noted adjustments to the sampling algorithm (where multiple ties are proposed to change during each sampling iteration). We have, in effect, a machine that explores a space of multiple consensus graphs that may not be "close" to each other in the posterior distribution. With multimodal distributions, MCMC method sampling algorithms tend to leave chains sampling only from local maxima and not the whole space. The consequence is that the diversity and distribution of the initialization provides the exploration of the space rather than the chains. This property of inadequate mixing seriously undermines the results of these samplers.

3.8.3 Limitations

Logical exclusions are possible with the particular questions

It may be that the Celebrities dataset allowed for certain logical comparisons that we had not anticipated. At least one expert reported that they did not see the cell membership as equally likely. The expert pointed out that *The Voice* singing competition show will only showcase musicians as celebrity judges. And, if one knows that Simon Cowell, for example, is not a singer (i.e. he does not sing), then the expert may logically exclude him from appearing alongside any other node that is known to belong to *The Voice* cell.

Cell membership may not be exclusive

We strove for exclusive cell membership when creating the Celebrities datasets. However, one expert pointed out that at least one of the judges may be on one or more of the TV shows we listed. In theory, this should not be a problem for the UNR model, since the consensus graph does not have exclusivity as a constraint. However, the K3NR model constrains the posterior partition to be exclusive: a given node can only be in one cell or another. In the posterior, though, since cell membership *is* conditionally independent, such cases are expected to generate a multi-modal posterior cell membership for the given multi-member node.

One course of action is to use different datasets that do not have this problem, and where the cell membership may be less ambiguous. We do have access to other validation data that queried experts on whether two automobile models are produced by the same automobile maker and plan to analyze them in the future.

Multiple consensus may need other aspects of mixture models

The K3NR model may actually be a choice of which cultural consensus is being conditioned on. If that is the case, then it may be beneficial to have separate ability, difficulty, bias, etc. that is conditioned on which cultural truth is being used.

3.8.4 Lessons learned and future work

One of the most important lessons learned in this process is that constrained discrete models and complex mixture models are difficult beasts to tame. Such models have well-deserved reputations for being unwieldy due to lack of continuity and explosion of sample space with large data.

More realistic biases in the response model

The single bias parameter for lack of knowledge of both nodes is not realistic. Under the models, the following guessing biases are at play under the four knowledge conditions for a given tie response.

$K_{i,j}$	$K_{i,k}$	NR biases	Proposed biases	Tie type
0	0	G_i	G_{i1}	Outside knowledge clique
0	1	G_i	G_{i2}	Frontier
1	0	G_i	G_{i2}	Frontier
1	1	NA: correct response	NA: correct response	Within knowledge clique

Table 3.6: Dual biases proposal.

Table 3.6 also shows the suggested two bias parameters per expert. Bias G_{i1} would account for tie responses for which the expert knows neither of the nodes (i.e. complete ignorance of the tie). The other, G_{i2} accounts for response bias when exactly one of the two nodes is known.

Recall the "knowledge clique" concept introduced earlier. The UNR and K3NR models specify that only within the knowledge clique (ties between known nodes) will the expert always respond precisely with the consensus tie value, and that all other ties are guesses with the same bias. The proposed additional bias parameter distinguishes the bias used for nodes within the clique, ties completely outside the clique, and the "frontier" ties connecting nodes in the clique those outside it.

Such a addition to the UNR/K3NR models would be important for modeling an expert's knowledge of a node's class. It is realistic to assume that an expert who generally knows about nodes belonging to one class would also believe that unknown nodes are not part of that class. For example, an expert may know the *Sentra* is a car model made by Nissan, and would likely claim that the *Camry* is not also made by Nissan, even if they did not know what company makes a Camry (for they would know about Nissans, generally).

Combine the PDGCM with the UNR model

One possible issue is that the NR models utilize a very strict sense of "knowledge" and have a lot of parameters for which there is not very much data to inform. Furthermore, in the case of K3NR in particularly, the restrictions cause a headache when creating sampling methods for the models.

Important steps in Gibbs sampling for flexible model development using BUGS, WinBUGS, OpenBUGS, JAGS, and `stan` have been made in the last 10-20 years that have opened up great pathways in cognitive modeling (perhaps to the dismay of mathematicians). Leaning on the JAGS approach, for example, we may introduce a continuous version of the NR models.

The NR models have the strict binary operator on the nodal knowledges. However, we can parameterize a smooth function of the two knowledges that we could then estimate that would provide a way to continuously differentiate between the combinations of K_j and K_k. A flat square with a spike near K_1K_2=(1,1) would represent the current UNR setup. A new model could then allow for that function to smoothly change to a "pit" near (0,0) and high plateau elsewhere. This provides an "OR" operation (in contrast to the "AND" of the NR models). This parameter should then inform the researcher of the *nature* of the nodal relationship on a continuous scale. Such parameterizations would provide insight into inhomogeneous nature of some networks.

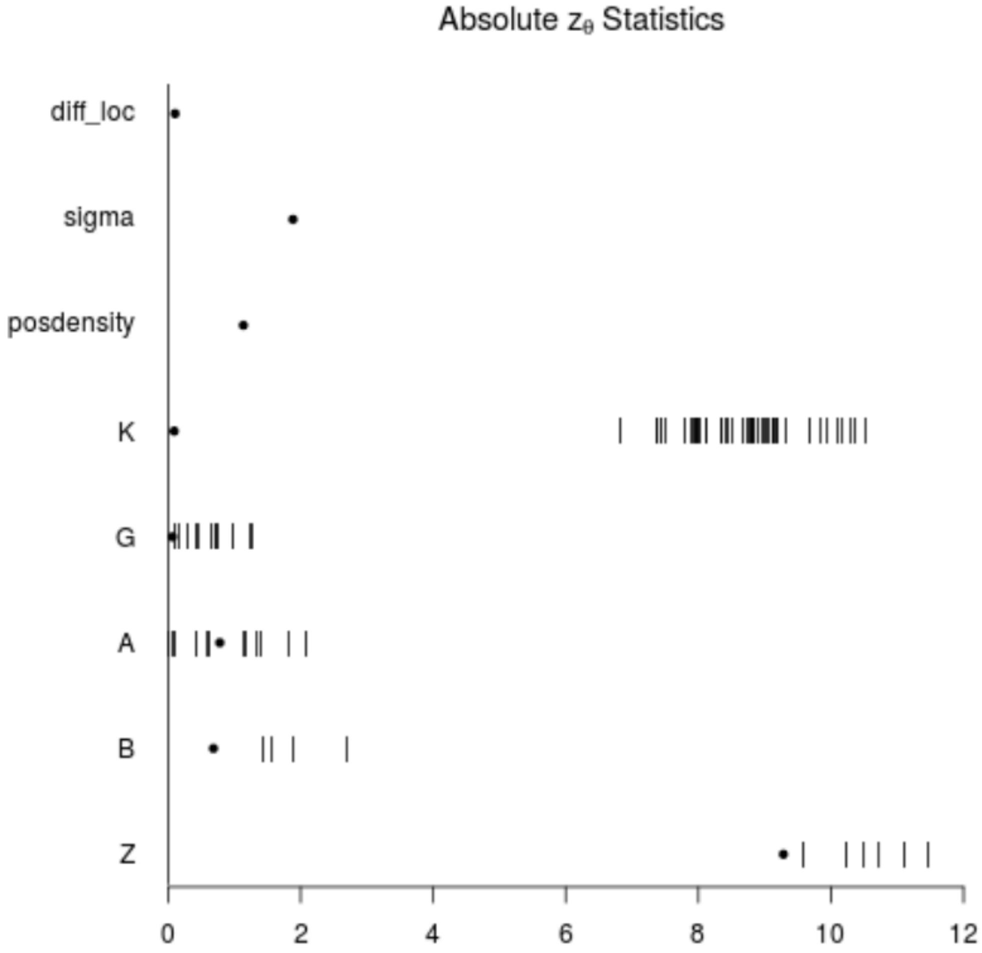


Figure 3.14: Output from the BayesValidate software library for R as applied to the UNR model.

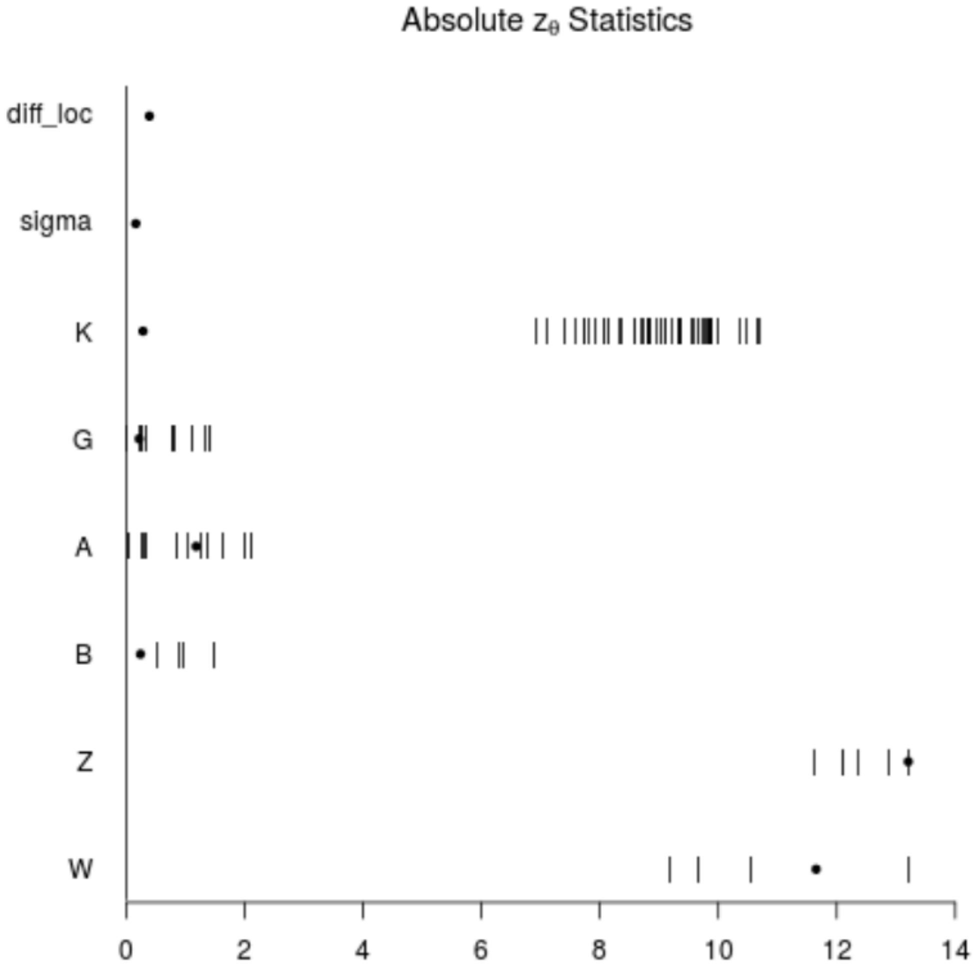


Figure 3.15: Output from the BayesValidate software library for R as applied to the UNR model.

Chapter 4

Incomplete design

4.1 Introduction

This chapter explores using the survey design discussed earlier in this document (Chapter 2, Section 2.6) to provide an incomplete data design for use with Cultural Consensus Theory (CCT) models. The survey design algorithm chooses a randomized sequence of ties to present an expert when eliciting the values of ties in a graph of N nodes (see Figure 2.2 for an example of generating such a sequence). The survey design offers access to randomized incomplete data that is free from breaks in conditional independence assumptions built into many CCT models. These incomplete datasets can be used for model estimation in the same way as complete data, though with the accompanying issues of less data. A data collection plan utilizing the incomplete design offers benefits of reduced burden for experts.

While all the models discussed in this dissertation can accommodate incomplete data, the GCM/Tie Model discussed in Chapter 2 is used here for the sake of simplicity. Likewise, the UndirWestUS "Borders" data of Chapter 2 is presented as an example application. The properties of using incomplete data for estimation should be analogous when used with the other models discussed in this dissertation.

4.2 Phases of data

Recall that the survey design allowed separating the data from a given expert into three disjoint phases that differ in whether the expert can logically infer a response based on their previous responses. Given N nodes and the presentation of all $N(N-1)/2$ unordered pairs of them, the design distinguishes these phases based on the number and size of cycles of question-ties completed in each phase.

When sequencing all the ties, we consider ties that are presented and those that are not yet presented at a given index in the sequence of questions. At the beginning of the sequence, no ties have been presented, and therefore the graph of presented ties is empty. Moving forward along the sequence of ties, the graph of presented ties fills in until it is the complete graph where each question (tie) has been presented exactly once.

- *Phase 1*: The first phase presents N pairs of nodes such that a cycle of all N nodes has been presented after the first N questions. The first $N - 1$ questions form a path connecting all N nodes with the two end nodes adjacent to 1 node each and other nodes adjacent to 2 nodes each. The Nth tie in the design completes the first cycle, with a cycle length of N. This is the largest possible cycle of N nodes that could be presented.

- *Phase 2*: The second phase consists of ties following the first phase, but before the third phase. It consists of remaining ties such that the expert will not have any triads present in the presentation graph after the last question of this phase is presented.

- *Phase 3*: The third phase consists of the remaining ties that complete the presentation graph. Each question in Phase 3 is the third leg of at least one triad.

For $N = 8$ nodes, the first phase has 8 questions, the second phase has 8 questions, and the third phase has 12 questions.

The most important difference to examine is between using only Phase 1 ("P.1") and using Phases 1-3 ("P.All"). In terms of size of data, the P.1 data will contain MN responses, linear with respect to the number of nodes. And the the complete P.All data contains $MN(N-1)/2$ responses, which is quadratic with respect to the number of nodes. Excluded phases are set as missing data in their respective analyses.

Posterior parameter distributions will be more dispersed using only P.1. In particular, we expect similar modal consensus estimates using the decimated data.

4.3 Simulation studies

To explore the effect of P.1 incomplete design on estimation, simulations were conducted on four sizes of datasets. The four sizes derived from the combinations of setting M to 7 or 21 experts and N to 6 or 8 nodes (15 or 28 ties, respectively). Ten P.All datasets were simulated for each size using the GCM/Tie Model priors and hyperpriors. For each of these data matrices, a corresponding data matrix was created that only retained (randomized) P.1 responses for each expert. The GCM/Tie Model was applied to each simulated dataset using 8000 iterations and 7000 burn-in per each of four chains.

The results confirmed that the models can technically work under such decimated data input. The main sacrifices, of course, are that when less data is used, the priors of the model have more influence on the posterior distribution of the model parameters than if all the data had been used. The GCM/Tie Model uses non-informative priors for the model parameters which generally have high standard deviations when compared to their usual estimated posterior distributions. Estimates on P.1 incomplete data (compared to corresponding P.All data) showed higher standard deviations for model parameters, as expected. Mean standard deviations from the posterior samples of important parameters are shown in Figure 4.1.

The main question is whether the recovery of consensus will suffer greatly with the incomplete design. Fortunately, the recovery rate of the ground truth in simulations were impressive considering the greatly reduced data. Table 4.1 gives the recovery rates (i.e. rate of posterior Z_{jk} matching the Z_{jk} used for data generation) for the various simulations. It is clear (and expected) that using P.All will yield higher recovery rates than without. It is surprising, however, that the rates of recovery are so high using the incomplete design ($\geq 80\%$).

The table also gives a hint at what course of action a researcher may want to take if they wanted to increase the consensus recovery rate given P.1 as a baseline. Two options present themselves: increase the number of experts surveyed or switch to using more questions per

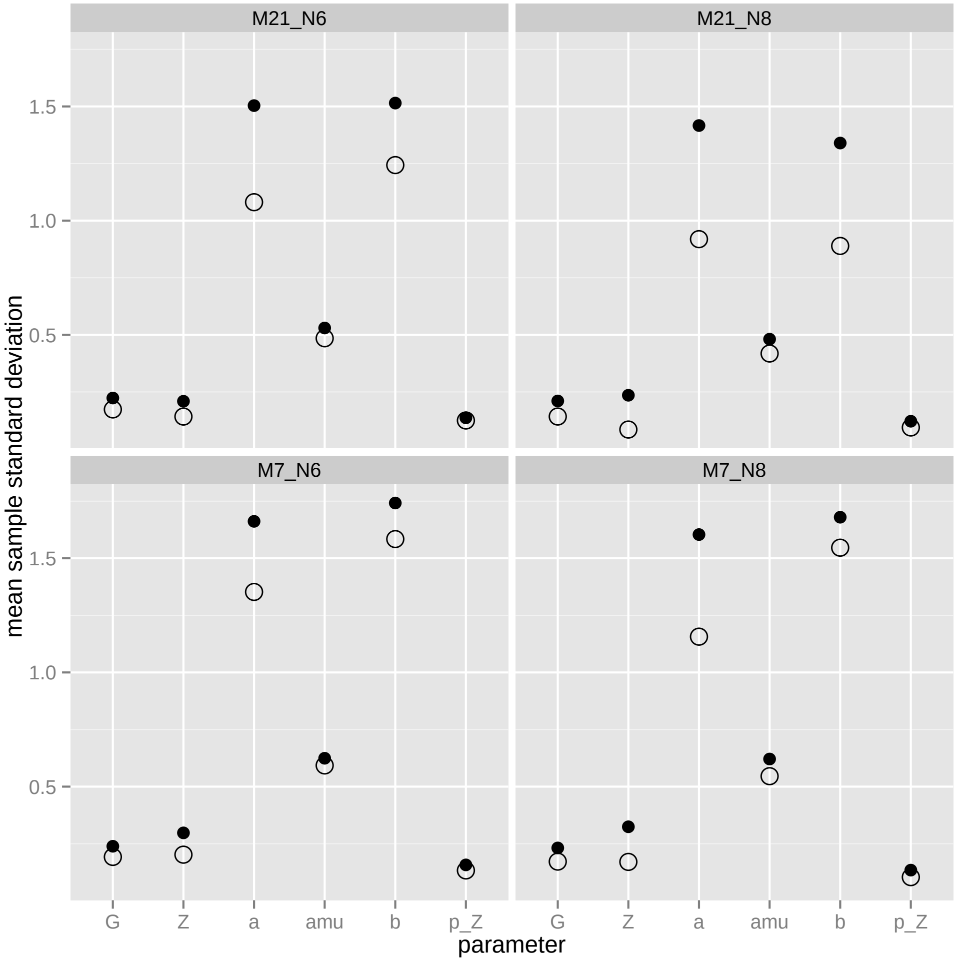


Figure 4.1: Mean standard deviations across simulated datasets of each size. Dataset sizes are labeled according to the number of experts and number of nodes. Solid dots plot the P.1 values and circles plot the P.All values.

expert. Given the limited set of simulations presently available, the greatest benefit comes from using complete data collection for the original number of experts instead of, in this case, tripling the number of experts being surveyed. That is, for the given number of nodes (either 6 or 8), the greatest increase in simulated recovery rates comes from switching from P.1 to P.All rather than tripling the number experts used.

Size	M experts	N nodes	Ties per expert	Phases	Recovery rate
M7_N6	7	6	6	P.1	0.85
M7_N6	7	6	15	P.All	0.95
M21_N6	21	6	6	P.1	0.87
M21_N6	21	6	15	P.All	0.91
M7_N8	7	8	6	P.1	0.80
M7_N8	7	8	28	P.All	0.92
M21_N8	21	8	8	P.1	0.89
M21_N8	21	8	28	P.All	0.96

Table 4.1: Recovery rate of Z_{jk} consensus tie values using posterior marginal modes across 10 simulated datasets.

4.4 Deep versus wide data collection

For another approach to the question of whether to prefer P.All over P.1 or to prefer greater numbers of experts over fewer, we suppose an applied CCT researcher may only ask a fixed number of questions for their entire survey. They must decide whether to survey a large number of experts with P.1's N questions or to survey a smaller number of experts with P.All's $N(N-1)/2$ questions.

The next simulation addresses a single constraint of 280 questions to be asked for eliciting an $N = 8$ graph from a set of M experts. A "wide" dataset would consist of P.1 data from $M = 35$ experts while a "deep" dataset would consist of P.All data from $M = 10$ experts. One size of dataset, "sm", simulates $M = 10$ experts while the "lg" size simulates $M = 35$ experts. Twenty P.All datasets were simulated for each size. For each of these, corresponding data was created that only retained (randomized) P.1 responses for each expert. The *deep* and *wide* datasets are therefore *sm-P.All* and *lg-P.1*, respectively, and these two have the same number of responses included in the model estimation (i.e. 280). The GCM/Tie Model was applied to each simulated dataset using 8000 iterations and 7000 burn-in per each of four chains.

Based on these simulations, we find that the recovery rates of the underlying ground truth (simulated) for the four dataset/phase conditions show that, of the *deep* and *wide* approaches, it is unclear which is better to use. Table 4.2 shows the recovery rates for each dataset/phase condition. With the same recovery rates, neither the deep nor wide approaches actually offers a clear benefit over the other.

Nickname	Size	M experts	N nodes	Ties per expert	Phases	Recovery rate
n/a	sm	10	8	8	P.1	0.86
“deep”	sm	10	8	28	P.All	0.94
“wide”	lg	35	8	8	P.1	0.94
n/a	lg	35	8	28	P.All	0.99

Table 4.2: Recovery rate of Z_{jk} consensus tie values using posterior marginal modes of 20 simulated datasets. $N = 8$ nodes are elicited, with 8 ties elicited for P.1 data, and 28 ties for P.All data. $M = 10$ experts were simulated in the “sm” datasets and $M = 35$ in the “lg” datasets.

4.5 Real data

We now turn to an application of P.1 data analysis. The UndirWestUS data from Chapter 2 was used to demonstrate recovery of ground truth consensus in a real survey of general knowledge under the incomplete and complete data conditions. The data used are shown in Figures 4.2 and 4.3 where the decimation of data is evident. We began by estimating the model parameters for both forms of data.

With some exceptions, the characteristic moderation of tie values (i.e. closer to 0.5) in the P.1 data is evident, indicating there is less certainty of those values in the posterior consensus (see Figure 4.4). This lack of certainty results from less data being used which imposes less influence on the non-informative priors. Also stemming from the decimated data are four cases of “reversals” where P.1 modal tie values are different than those from P.All: WY + ID, UT + ID, NV . MT, NV + ID.

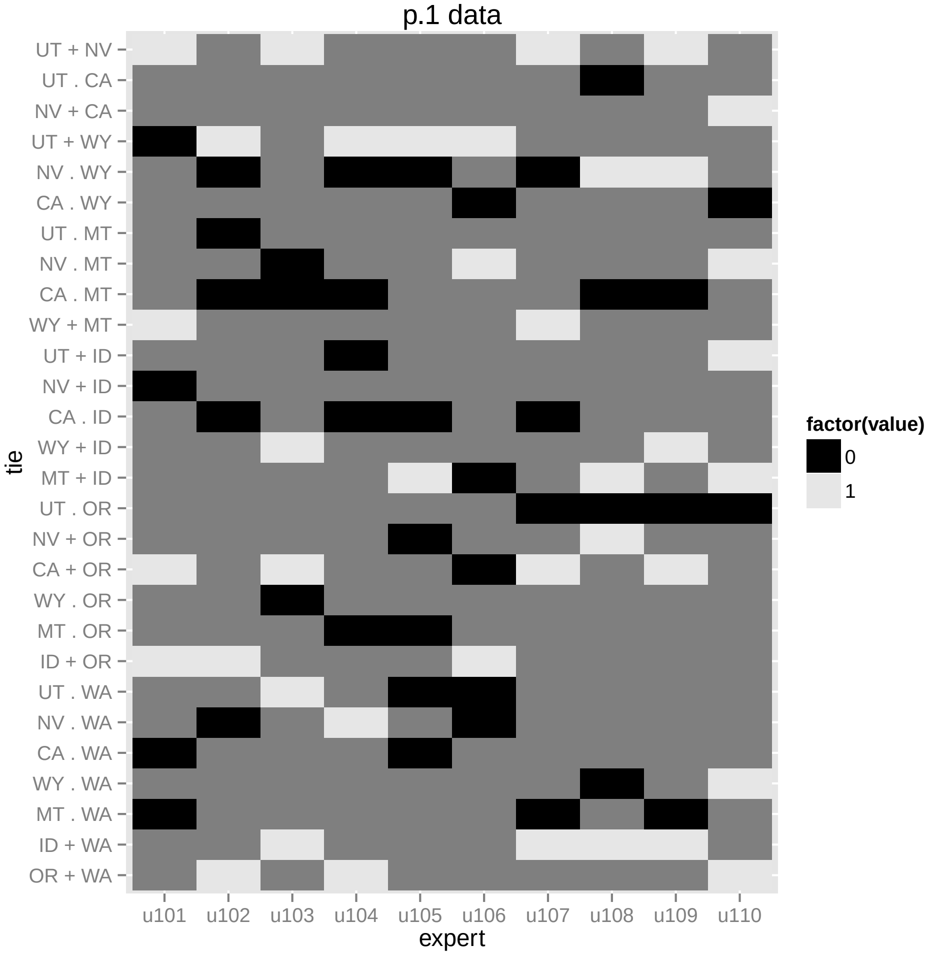


Figure 4.2: Incomplete data matrix (P.1). Middle gray tiles represent missing data. Note the eight data points per expert (column).

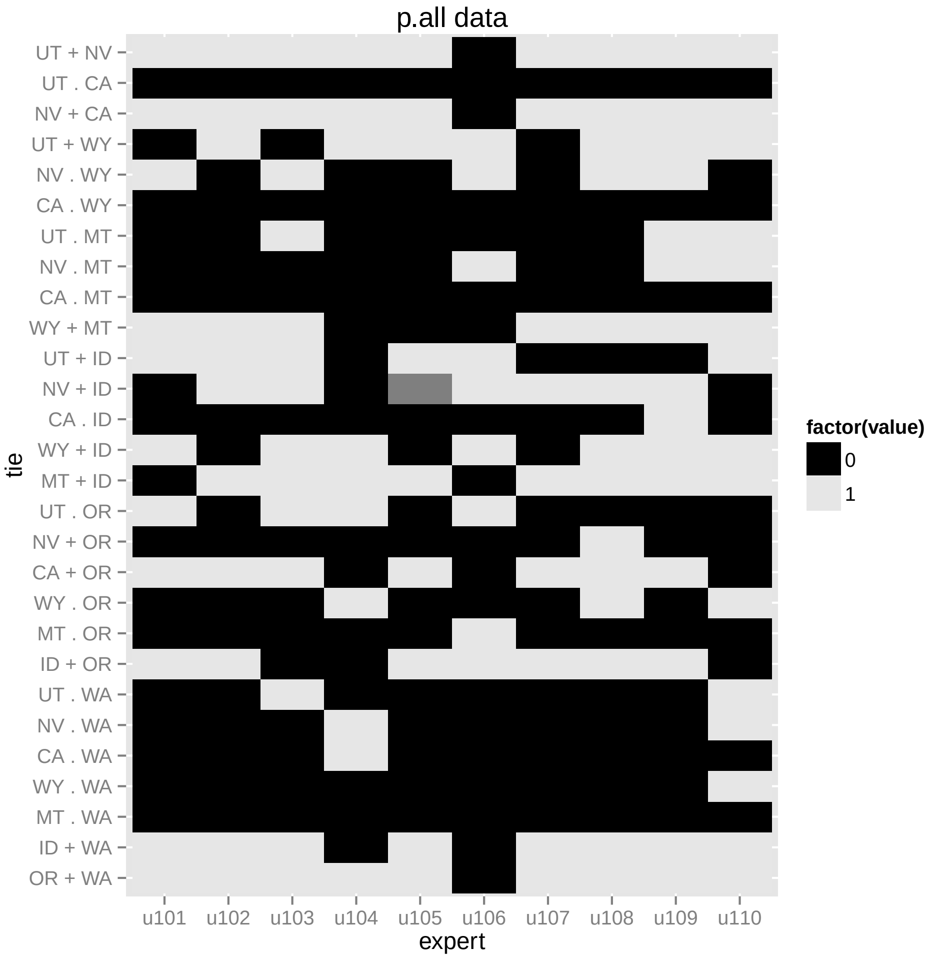


Figure 4.3: Complete data matrix (P.All) with one missing datum (in middle gray).

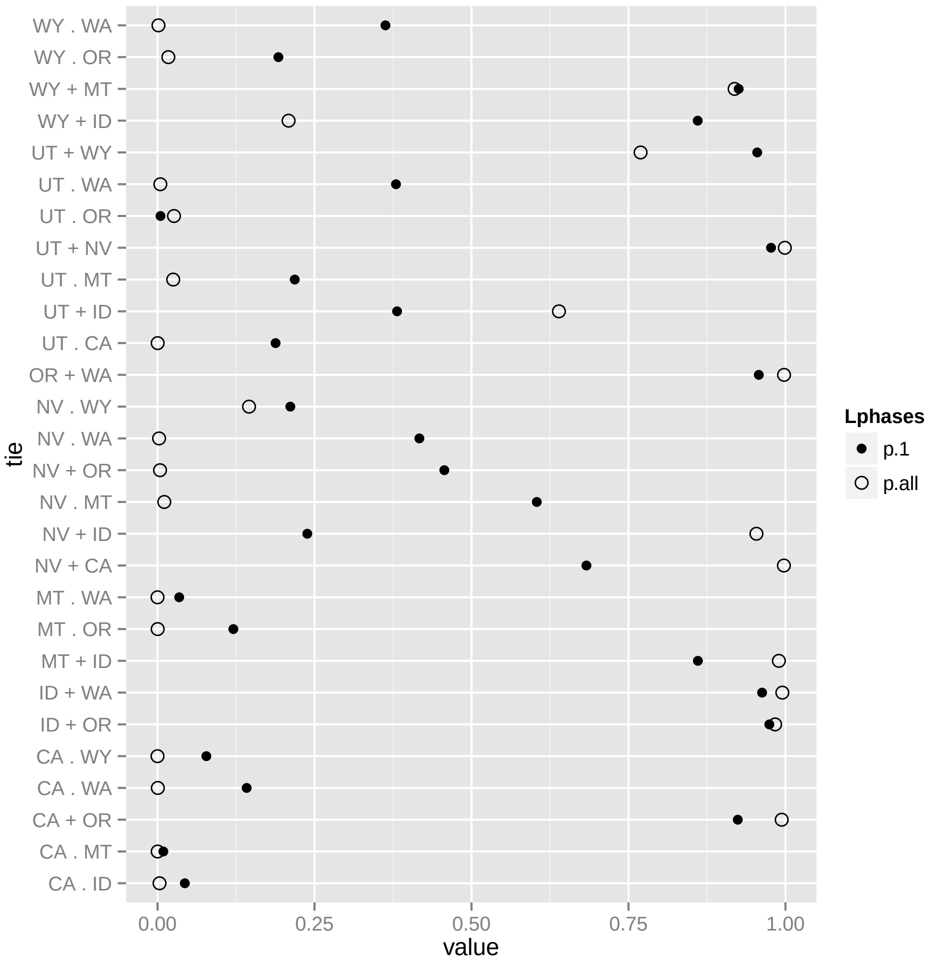


Figure 4.4: Posterior Z_{jk} consensus tie values for P.1 and P.All data.

Posterior estimates for abilities using the incomplete design (P.1) are all generally close to zero, which is the expected value of ability in the prior (see Figure 4.5). Only when all the data is included (P.All) are the nuances of the experts' differing abilities shown. As mentioned in Chapter 2, the experts u102, u105, and u107 show the most consistent reporting pattern with the highest ability estimates when all the data is included, but no possible consistency can be seen in their P.1 data.

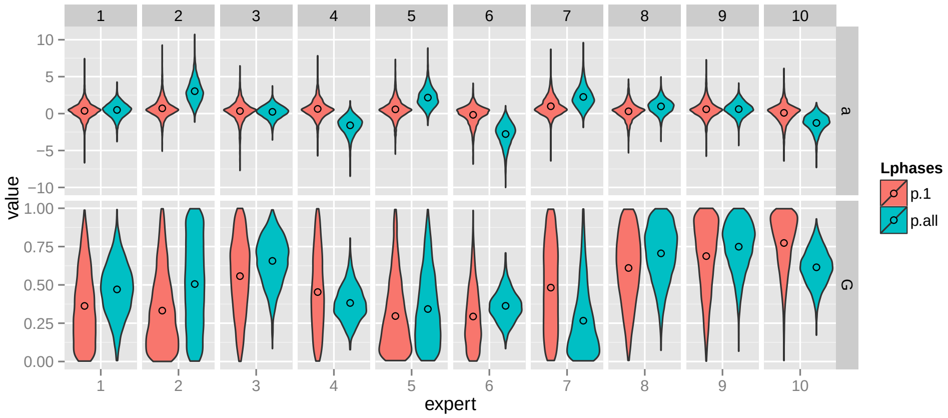


Figure 4.5: Posterior distributions and means for expert parameters (ability and guessing bias) for each of P.1 and P.All.

Lower dispersion and nuanced location of the tie difficulty parameters come with the additional data of P.All (see Figure 4.6) in the same manner as the expert parameters.

4.6 Discussion

The main benefit of using the survey design discussed in Chapter 2, Section 2.6 is the maintenance of conditionally independent responses from experts who may otherwise use some heuristics. In particular, the first phase of N questions in that design are nearly guaranteed to be free of such issues. This additional benefit of the design that allows a truncated survey of N questions per expert would reduce the load on expert who may tire after long surveys, especially when N is large.

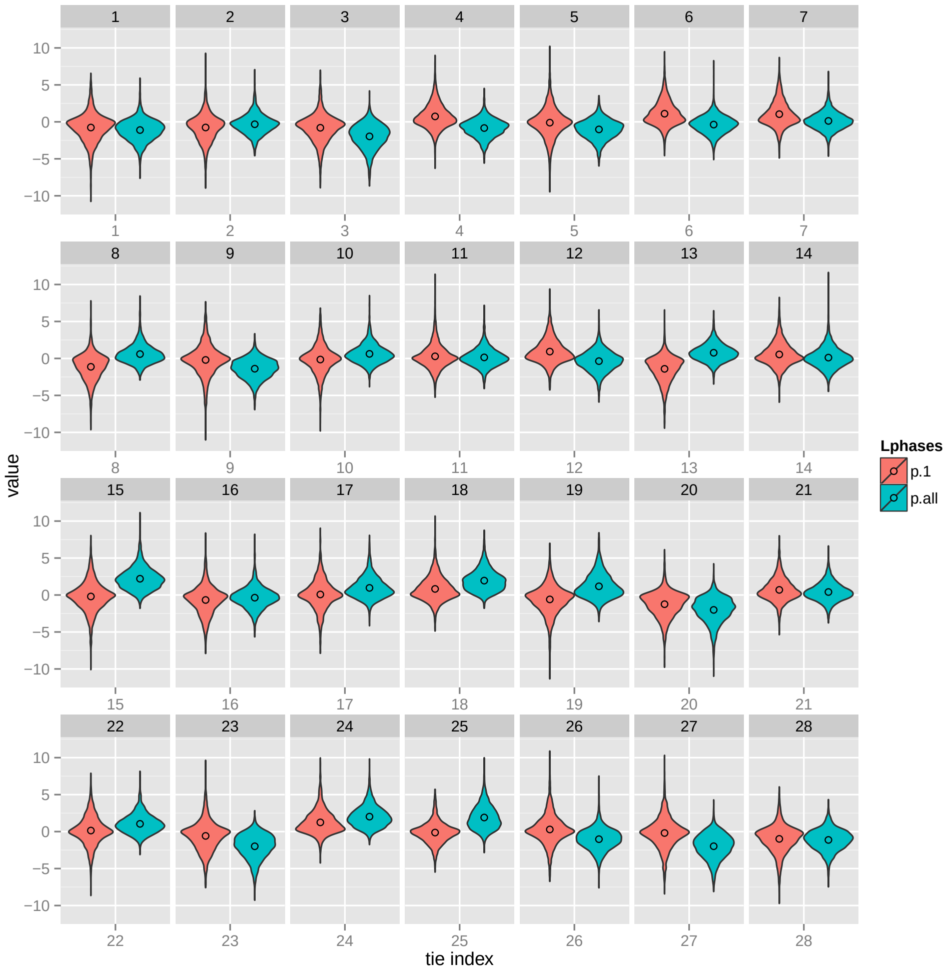


Figure 4.6: Posterior distributions and means for tie difficulty parameters for each of P.1 and P.All.

However, using P.1 data alone should be done with caution. When possible, the applied researcher is advised to survey experts *deeply* for these graph elicitations rather than widely (and shallowly) across multiple experts. Simulation studies and the example application discussed in this chapter have suggested that the model parameters may suffer from the degraded data of a P.1 analysis.

On the other hand, P.1 data analysis can estimate a remarkable rate of success in estimating consensus. That is, one can reasonably estimate complete graphs with substantial reduction in expert burden, research time, and costs. This is the case despite that other model parameter posterior distributions will not be much different than their priors.

The the models discussed in this dissertation require that experts not base their response to question on their responses to previous questions. If experts are at all going to break this assumption, then the incomplete survey design given here can mitigate that since the models can technically accommodate the missing data. Likewise, under model assumptions, using the P.1 data allows for shorter surveys that still provide good estimations of the primary target of applied CCT researchers: estimates of the ground truth.

Chapter 5

Conclusion

This dissertation has extended the application areas and contexts of application for Cultural Consensus Theory (CCT) models in several ways. Besides this, some important technical lessons have been addressed with respect to CCT modeling of graphs. Chapter 2 began by introducing a model that decomposes the difficulty of tie relationships into the difficulties of its adjacent nodes. A new survey design algorithm was also introduced that allows for simple graph elicitation while mitigating certain aspects of within-expert response dependence that any graph elicitation (in addition to those described) will benefit from.

Chapter 3 broached a challenging and new realm of highly constrained discrete CCT models in two ways. A new discrete parameter "layer" was added to coordinate expert responses and is unusual compared to the typical CCT assumptions of conditionally independent responses. This layer added pairwise node comparisons to the array of question formats now handled by CCT models. Furthermore, Chapter 3 extended previous work of imposing hard constraints on CCT answer keys by extending the bipartition constraint on consensus graphs (Agrawal and Batchelder, 2012) to that of a more general k-partition. The discrete response model and constrained answer key are only examples of a broad range of constraints that can use custom sampling algorithms to address their complex parameter support spaces.

Some important broad lessons have been learned in undertaking the modeling described in this dissertation. CCT models for graphs, discrete responses, and hard constraints are generally not as straightforward to estimate as CCT models for independent responses and primarily continuous responses. Despite the difficulties, this work shows that there is promise for adapting CCT models for special formats and constrained response and consensus patterns (such as clusters).

Sampler chain convergence can be difficult (with finite computation time) due to the great changes in model likelihoods that come from small changes to discrete parameter vectors. Future models may benefit from reconsidering what discrete parameters provide the researcher that a continuous version cannot. For example, some discrete decisions (such as classification) can be done at the interpretation level rather than within the model itself.

Writing computer code that is mostly isolated from other researchers can lead to computer "bugs" and design problems. For probabilistic programming, errors may be even more difficult to catch. Team coders may have fewer problems with this due to organizational pressure and peer accountability, and so it is highly suggested that mostly independent coders adhere to some wisdom from the professional software development industry. One strategy is to use automated software testing tools when available. The BayesValidate (Cook and Gelman, 2006) software used in this dissertation gives a way of detecting certain probabilistic errors in ones code. Another suggestion is to "be appropriately lazy" (Andries van Dam, personal communication, 1997). This entails, among other things, writing only modifications to existing software libraries and tools that already benefit from shared development and debugging or longevity. Indeed, one path only just begun by the author was to extend the JAGS framework (previously mentioned).

In an effort to help other researchers be appropriately lazy when developing constrained CCT models, the author has provided code to readers in two ways. The appendix contains JAGS code for parameter estimation of the GCM/Tie Model and the PDGCM of Chapter 2. Additionally, a Bayesian estimation framework mentioned in Chapter 3 is available to interested readers by contacting the author or downloading it from the author's web page (please see the footnote in Chapter 3, Section 3.5 for specifics).

Bibliography

Agrawal, K. and Batchelder, W. H. (2012). Cultural Consensus Theory: Aggregating Signed Graphs under a Balance Constraint. In Yang, S. J., Greenberg, A. M., and Endsley, M., editors, *Social Computing, Behavioral - Cultural Modeling and Prediction*, volume 7227 of *Lecture Notes in Computer Science*, pages 53–60. Springer, Heidelberg.

An, W. and Schramski, S. (2015). Analysis of contested reports in exchange networks based on actors credibility. *Social Networks*, 40:25–33.

Anders, R. and Batchelder, W. H. (2012). Cultural consensus theory for multiple consensus truths. *Journal of Mathematical Psychology*, 56(6):452–469.

Anders, R. and Batchelder, W. H. (2015). Cultural Consensus Theory for the Ordinal Data Case. *Psychometrika*, 80(1):151–181.

Batchelder, W. H. (2009). Cultural Consensus Theory: Aggregating Expert Judgments about Ties in a Social Network. In Liu, H., Salerno, J. J., and Young, M. J., editors, *Social Computing and Behavioral Modeling*, pages 1–9. Springer US, Boston, MA.

Batchelder, W. H. and Anders, R. (2012). Cultural Consensus Theory: Comparing different concepts of cultural truth. *Journal of Mathematical Psychology*, 56(5):316–332.

Batchelder, W. H., Kumbasar, E., and Boyd, J. P. (1997). Consensus analysis of three-way social network data. *The Journal of Mathematical Sociology*, 22(1):29.

Batchelder, W. H. and Romney, A. K. (1988). Test theory without an answer key. *Psychometrika*, 53(1):71–92.

Batchelder, W. H., Strashny, A., and Romney, A. K. (2010). Cultural Consensus Theory: Aggregating Continuous Responses in a Finite Interval. In Chai, S.-K., Salerno, J., and Mabry, P., editors, *Advances in Social Computing*, volume 6007 of *Lecture Notes in Computer Science*, pages 98–107. Springer, Heidelberg.

Bearman, P. S., Moody, J., and Stovel, K. (2004). Chains of Affection: The Structure of Adolescent Romantic and Sexual Networks. *The American Journal of Sociology*, 110(1):44–91.

Brusco, M., Doreian, P., Steinley, D., and Satornino, C. B. (2012). Multiobjective Blockmodeling for Social Network Analysis. *Psychometrika*, 78(3):498–525.

Butts, C. T. (2003). Network inference, error, and informant (in)accuracy: a Bayesian approach. *Social Networks*, 25(2):103–140.

Cartwright, D. and Harary, F. (1956). Structural balance: a generalization of Heider's theory. *Psychological Review*, 63(5):277–293.

Comrey, A. (1973). *A first course in factor analysis.* Academic Press, New York.

Comrey, A. L. (1962). The minimum residual method of factor analysis. *Psychological Reports*, 11:15–18.

Cook, S. (2006). *BayesValidate: BayesValidate Package.* R package version 0.0. http://CRAN.R-project.org/package=BayesValidate.

Cook, S. and Gelman, A. (2006). Bayesian software validation. *R News*, 6(1).

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009). *Introduction to Algorithms.* The MIT Press, Cambridge, Mass., 3rd edition.

Crowther, C. S., Batchelder, W. H., and Hu, X. (1995). A measurement-theoretic analysis of the fuzzy logic model of perception. *Psychological Review*, 102(2):396–408.

Fischer, G. H. and Molenaar, I. W., editors (1995). *Rasch Models: Foundations, Recent Developments, and Applications.* Springer-Verlag.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2004). *Bayesian data analysis.* Chapman & Hall/CRC, Boca Raton, FL, 2nd edition.

Gelman, A. and Hill, J. (2006). *Data Analysis Using Regression and Multilevel/Hierarchical Models.* Cambridge University Press, 1st edition.

Harary, F. (1959). On the Measurement of Structural Balance. *Behavioral Science*, 4(4).

Herrera-Viedma, E., Cabrerizo, F. J., Kacprzyk, J., and Pedrycz, W. (2014). A review of soft consensus models in a fuzzy environment. *Information Fusion*, 17:4–13.

Jackman, S. (2009). *Bayesian Analysis for the Social Sciences.* Wiley.

Johnson, V. E. and Albert, J. H. (1999). *Ordinal Data Modeling.* Statistics for Social and Behavioral Sciences. Springer-Verlag, New York.

Karabatsos, G. and Batchelder, W. H. (2003). Markov chain estimation for test theory without an answer key. *Psychometrika*, 68(3):373–389.

Krackhardt, D. (1987). Cognitive social structures. *Social Networks*, 9(2):109–134.

Lee, M. D. (2011). How cognitive modeling can benefit from hierarchical Bayesian models. *Journal of Mathematical Psychology*, 55(1):1–7.

Lee, M. D. (2014). *Bayesian Cognitive Modeling: A Practical Course.* Cambridge University Press.

Lunn, D. J., Thomas, A., Best, N., and Spiegelhalter, D. (2000). WinBUGS - A Bayesian modelling framework: Concepts, structure, and extensibility. *Statistics and Computing*, 10:325–337.

Macmillan, N. A. and Creelman, C. D. (2004). *Detection Theory: A User's Guide*. Psychology Press, 2nd edition.

Oravecz, Z., Anders, R., and Batchelder, W. H. (2013). Hierarchical Bayesian Modeling for Test Theory Without an Answer Key. *Psychometrika*, 80(2):341–364.

Oravecz, Z., Vandekerckhove, J., and Batchelder, W. H. (2014). Bayesian Cultural Consensus Theory. *Field Methods*, 26(3):207–222.

Plummer, M. (2003). JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling.

Plummer, M. (2014). *rjags: Bayesian graphical models using MCMC*. R package version 3-14. http://CRAN.R-project.org/package=rjags.

Plummer, M., Best, N., Cowles, K., and Vines, K. (2006). Coda: Convergence diagnosis and output analysis for mcmc. *R News*, 6(1):7–11.

R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Raîche, G., Walls, T., Magis, D., Riopel, M., and Blais, J.-G. (2013). Non-graphical solutions to Cattell's scree test. *Methodology: European Journal of Research Methods for the Behavioral and Social Sciences*, 9:23–29.

Ranganathan, A., Menegatti, E., and Dellaert, F. (2006). Bayesian inference in the space of topological maps. *Robotics, IEEE Transactions on*, 22(1):92–107.

Revelle, W. (2014). *psych: Procedures for Psychological, Psychometric, and Personality Research*. Northwestern University, Evanston, Illinois. R package version 1.4.8. http://CRAN.R-project.org/package=psych.

Romney, A. K., Weller, S. C., and Batchelder, W. H. (1986). Culture as Consensus: A Theory of Culture and Informant Accuracy. *American Anthropologist*, 88(2):313–338.

Smith, J. B. and Batchelder, W. H. (2010). Beta-MPT: Multinomial processing tree models for addressing individual differences. *Journal of Mathematical Psychology*, 54(1):167–183.

Stan Development Team (2014). *Stan Modeling Language Users Guide and Reference Manual, Version 2.5.0.*

Su, Y.-S. and Yajima, M. (2015). *R2jags: A Package for Running jags from R*. R package version 0.05-01. http://CRAN.R-project.org/package=R2jags.

Weller, S. C. (2007). Cultural Consensus Theory: Applications and Frequently Asked Questions. *Field Methods*, 19(4):339–368.

Appendix A

GCM/Tie Model and PDGCM code listings in JAGS format

The following are the JAGS (Plummer, 2003) code texts used for parameter estimation for the GCM/Tie Model and PDGCM described in the text.

The `data` block for each model sets up some fixed parameters for the model such as the number of experts and the number of items (ties and nodes). Only the data matrix `X` need be supplied. The `X` matrix must be in a format where each row consists of an expert's responses and each column represents responses to a tie. The ties must be in a specific order. This is the `lower.tri` order and consists of the lower triangle of the adjacency matrix for all the nodes, excluding the main diagonal, reading in column-first order. This is the same order found when using the `R` program to obtain (or fill) the `lower.tri` portion of a given matrix.

For a set of four nodes, labeled 1, 2, 3, and 4, the order of ties is the following which uses {row, column} indexing notation into the adjacency matrix (as in the `R` program).

```
{2,1} (tie 1, column 1)

{3,1} (tie 2, column 2)

{4,1} (tie 3, column 3)

{3,2} (tie 4, column 4)

{4,2} (tie 5, column 5)

{4,3} (tie 6, column 6)
```

The order can also be visualized in this 4 node by 4 node matrix:

$$\begin{pmatrix} . & . & . & . \\ 1 & . & . & . \\ 2 & 4 & . & . \\ 3 & 5 & 6 & . \end{pmatrix}$$

Here is an example `X` matrix for 3 experts and 6 ties:

	tie1	tie2	tie3	tie4	tie5	tie6
expert1	0	1	1	0	1	0
expert2	0	0	1	0	0	1
expert3	1	1	1	0	1	1

A.1 Code listing for GCM

```
### GCM Estimation ###

data
```

```
{
    xdim <- dim(X)
    M <- xdim[1]
    nties <- xdim[2]
    n_items <- nties    # Items are Ties in the GCM
} # end data block
model
{
        # Population Ability mean and precision
        amu ~ dnorm(0, 2)
        asigkap <- 3
        asig ~ dunif(0, asigkap )
        atau <- 1 / pow( asig, 2)

        # Consensus
        p_Z ~ dunif( 0, 1 )

        # Population Difficulty mean and precision
        bmu <- 0
        bsigkap <- 3
        bsig ~ dunif( 0, bsigkap )
        btau <- 1 / pow( bsig, 2 )

        # Summaries for evaluation
        meanD <- mean(D[,])
        meana <- mean(a[])
        meanb <- mean(b[])
```

```
meanG <- mean(G[])

# Per-subject (per-expert) parameters
for ( i in 1:M ) {
    # Bias
    G[i] ~ dunif(0,1)

    # Ability
    a[i] ~ dnorm( amu, atau)
}

# Per item (per tie or per node, depending)
for ( itemi in 1:n_items ) {
    # Difficulty
    b[itemi] ~ dnorm( bmu, btau )
}

# Per tie difficulty for the GCM uses each tie difficulty
for ( itie in 1:nties ) {
    # Use the Tie difficulty stored in T which has been
    # previously assigned.
    T[itie] <- b[itie]
}

# Per tie (item) parametes
for ( itie in 1:nties ) {
    # Consensus
```

```
        Z[itie] ~ dbern( p_Z )
    }

    # Responses for each tie
    for ( i in 1:M ) {
        for ( itie in 1:nties ) {
            # Rasch-based competence
            D[i,itie] <- 1 /
                ( 1 + exp( -(a[i] - T[itie] ) ) )

            # Probability of responding 1
            pX[i,itie] <-
                ( D[i,itie] * Z[itie] ) +
                ( ( 1 - D[i,itie] ) * G[i] )

            # Response
            X[i,itie] ~ dbern( pX[i,itie] )
        }
    }
} # end model block
```

A.2 Code listing for PDGCM

```
### PDGCM Estimation ###
data
```

```
{
    xdim <- dim(X)
    M <- xdim[1]
    nties <- xdim[2]
    N <- 0.5 * ( 1 + sqrt( 1 + ( 8 * nties ) )  )
    n_items <- N    # Items are nodes in the PD model
} # end data block
model
{
        # Population Ability mean and precision
        amu ~ dnorm(0, 2)
        asigkap <- 3
        asig ~ dunif(0, asigkap )
        atau <- 1 / pow( asig, 2)

        # Consensus
        p_Z ~ dunif( 0, 1 )

        # Population Difficulty mean and precision
        bmu <- 0
        bsigkap <- 3
        bsig ~ dunif( 0, bsigkap )
        btau <- 1 / pow( bsig, 2 )

        # Summaries for evaluation
        meanD <- mean(D[,])
        meana <- mean(a[])
```

```
meanb <- mean(b[])
meanG <- mean(G[])

# Per-subject (per-expert) parameters
for ( i in 1:M ) {
    # Bias
    G[i] ~ dunif(0,1)

    # Ability
    # dnorm(mean, precision)
    a[i] ~ dnorm( amu, atau)
}

# Per item (per tie or per node, depending)
for ( itemi in 1:n_items ) {
    # Difficulty
    # dnorm(mean, precision)
    b[itemi] ~ dnorm( bmu, btau )
}

# For each j (row) node from 1 to
# just before the last row which is just an
# empty diagonal (i.e go to N-1)
for ( j in 1:(N-1) ) {
    # For each k (col) node starting
    # to the side of diagonal to the
    # last (i.e. go to N).
    for ( k in (j+1):N ) {
        # For each j (row) from k+1 to N (i.e. each
        # lower.tri row in the given column).
        # Combined, additive difficulty
        # Use the Tie difficulty stored in T which has been
        # previously assigned.
        T[ ((k-j)+(N*(j-1))-(j-1)*j/2)] <-
            mean( b[j] + b[k] )
    }
}

# Per tie (item) parametes
for ( itie in 1:nties ) {
    # Consensus
    Z[itie] ~ dbern( p_Z )
}

# Responses for each tie
for ( i in 1:M ) {
    for ( itie in 1:nties ) {
        # Rasch-based competence
        D[i,itie] <- 1 /
            ( 1 + exp( -(a[i] - T[itie] ) ) )

        # Probability of responding 1
        pX[i,itie] <-
            ( D[i,itie] * Z[itie] ) +
          ( ( 1 - D[i,itie] ) * G[i] )

        # Response
        X[i,itie] ~ dbern( pX[i,itie] )
      }
    }
} # end model block
```