Measuring and Modeling Networks of Human Social Behavior

Daniel Mark Wyatt

A dissertation submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

University of Washington

2010

Program Authorized to Offer Degree: Computer Science and Engineering

University of Washington
Graduate School

This is to certify that I have examined this copy of a doctoral dissertation by

Daniel Mark Wyatt

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Co-Chairs of the Supervisory Committee:

Jeff Bilmes

Tanzeem Choudhury

Reading Committee:

Jeff Bilmes

Tanzeem Choudhury

Mark Handcock

Date:

Signature______________________________

Date__________________________________

University of Washington

Abstract

Measuring and Modeling Networks of Human Social Behavior

Daniel Mark Wyatt

Co-Chairs of the Supervisory Committee:
Associate Professor Jeff Bilmes
Department of Electrical Engineering

Assistant Professor Tanzeem Choudhury
Department of Computer Science, Dartmouth College

New technologies have made it possible to easily collect information about social networks as they are acted and observed "in the wild," instead of as they are reported after-the-fact in surveys. This unprecedented access to social behavior data—data that captures the observable actions of multiple people as they interact with one another—provides opportunities to address many new research questions: How does local behavior relate to the global structure of the social network? How does a social network change over time? How can meaningful information be extracted from raw, recordable data? And how can all of this be done while protecting privacy?

With the goal of answering those questions, this dissertation presents new methods for measuring and modeling social networks derived from automatically recorded behavioral data. These techniques are presented in three parts.

First, new methods that use privacy-sensitive audio data to automatically find colocated people, determine who is conversation with whom, and detect who speaks when and how (pitch, rate, etc.) are presented. The use of these methods to gather a data set capturing a year's worth (426 person-hours) of real-world face-to-face conversations within a subject population of 24 graduate students is then recounted.

Second, two new extensions to exponential random graph models are proposed. These extensions exploit the richness of social behavior data and enable the new models to: (i) recover latent networks where hidden social relationships are observable only through noisy behavior data, and (ii) discover long range, high level properties of evolving social networks using time-inhomogeneous models.

Third, an *influence mixture model* is proposed that quantifies the amount of influence each person in a multi-person interaction exerts on the all of others. This measured influence is found to correlate positively with a person's centrality in her social network.

Table of Contents

Page

List of Figures

Figure Number Page

List of Tables

Table Number Page

Acknowledgments

Before anyone else, I want to thank the National Science Foundation for funding almost all of this research under NSF grant IIS-0433637, as well as Intel Labs Seattle for supporting this research in numerous other ways.

I want to thank Tanzeem for putting this project together in the first place and for always finding a way to keep me funded over many years.

I have been lucky enough to attend the meetings of two outstanding research groups: the Machine Learning Lunch run by Pedro Domingos and the netresearch meeting run by Mark Handcock, Martina Morris, and Steve Goodreau. The invaluable feedback and advice I have gotten from both of these groups has allowed me to entirely fake my way through graduate school by constantly stealing ideas from one group and passing them off as my own to the other.

For their help during the long and unpredictable data collection years I want to thank Brian Ferris, Julie Letchner, Alex Stupakov, Lin Liao, Eric Garcia, and especially Jonathan Lester.

Ranran Wang also deserves special thanks for her work on taming a painful data set as well as all of her helpful advice on network modeling.

I want to thank Veneta Tashev and Sarah Tachibana, two undergraduate students with whom I was fortunate enough to work, for all of their help and for showing me why my most of my ideas wouldn't work.

I also want to thank my boy Mikhail at Мишка for keeping me clothed on the cheap. I want to thank whoever has been funding the CSE espresso room for the past seven years: it has been a boon to my progress. And I want to thank Franklin, the UW CSE night watchman, for stopping by to chat during all those late nights.

Finally, I want to thank all of the incredibly smart people I've been around who have provided incredible feedback along the way: Sumit Basu, Gaetano Borriello, Jesse Davis, Dieter Fox, Krista Gile, Ashish Kapoor, Henry Kautz, Stanley Kok, Pavel Krivitsky, James Landay, Daniel Lowd, Anmol Madan, Don Patterson, Sandy Pentland, Matthai Philipose, Gail Potter, Dan Weld, Brian Ziebart, and all the others I'm sure I'm forgetting.

Dedication

to Chrissy, for all of her patience

Chapter 1

Introduction

It is worth beginning at the beginning, with one of the foundational essays on social network analysis. In his 1940 presidential address to the Royal Anthropological Society, Alfred Radcliffe-Brown described his vision for future research:

> If we set out to study, for example, the aboriginal inhabitants of a part of Australia, we find a certain number of individual human beings in a certain natural environment. We can observe the acts of behaviour of these individuals, including, of course, their acts of speech... Direct observation does reveal to us that these human beings are connected by a complex network of social relations. I use the term "social structure" to denote this network of actually existing relations. It is this that I regard it as my business to study. (Radcliffe-Brown, 1940)

This quotation contains two ideas that motivate the work presented here. The first is that social networks are worthwhile objects of study. That point has hopefully been demonstrated by much of the research done on social networks in the last half-century. Various network analyses have led to new insights into the diffusion of ideas (Valente, 1996), the spread of disease (Klovdahl, 1985), the adoption of new behaviors (such as smoking, Ennett and Bauman, 1993), disparities in economic status (Lin, 1999), and even changes in a person's physical and mental health (Smith and Christakis, 2008).

The second idea from Radcliffe-Brown that motivates this work is his assertion that social networks are empirically measurable via *social behavior*: the observable actions of two or more interacting people. This

Parts of this chapter were previously published in (Wyatt, Choudhury, Bilmes and Kautz, 2007), (Wyatt, Choudhury and Kautz, 2007), (Wyatt, Choudhury and Bilmes, 2007), (Wyatt, Choudhury, Bilmes and Kitts, 2008), (Wyatt, Choudhury and Bilmes, 2008), (Wyatt, 2009) and (Wyatt et al., 2010).

measurability makes possible the empirical and quantitative analysis of social networks and brings them (in Radcliffe-Brown's estimation) into the domain of the natural sciences.

Our ability to observe social networks, however, has not been as powerful as first imagined. The ethnographic "direct observation" approach suggested by Radcliffe-Brown is too laborious to employ for even moderate sized groups. Consequently, social network research has generally relied on data collected via surveys instead of direct observation. Often these surveys ask subjects to recall their previous social interactions (e.g. Lazega and van Duijn, 1997). But when self-reports of recalled interactions are compared to independent observations, the reliability of subjects' answers has been found to be shockingly poor (Killworth and Bernard, 1976; Bernard and Killworth, 1977; Killworth and Bernard, 1979; Bernard et al., 1980, 1982). One early study came to the dire conclusion that "people do not know, with any accuracy, those with whom they communicate" (Bernard and Killworth, 1977). Later studies found that durable, long-term patterns of communication are reliably reported, but moment-to-moment social interactions are not (Freeman et al., 1987). More troubling for research into network structure, individuals have been found to "fill in" non-existent interactions if they would increase the transitivity of the network (Freeman, 1992). Faced with these doubts about their data, some researchers lamented that "unfortunately, most naturally occurring interactive behavior (the stuff of which networks are built) is neither observable nor conveniently recorded in some automated fashion" (Killworth and Bernard, 1979).

That statement is no longer true.

1.1 Measuring Social Behavior

New technologies have made it possible to easily collect information about social behavior as it is enacted, instead of as it is recalled after-the-fact. Phone calls, text messages, emails, instant messages, on-line chat sessions, social media posts, and any other kind of electronically mediated communication can all be automatically recorded for large groups of people, over long periods of time. Advances in wearable sensors and ubiquitous computing have even made it possible to automatically record face-to-face conversations. Portable audio recording devices have grown in capacity while becoming smaller, cheaper, and more powerful. It is now possible to record all of the spontaneous, real-world speech for an entire group of people for a long period of time. These new recording methods finally provide the "direct observation" of social behavior—even of "acts of speech"—desired by Radcliffe-Brown.

1.1.1 Situated Speech Data

The automated recording of real-world speech remains important because, despite the rise in on-line interactions, face-to-face communication is still people's primary mode of social interaction (Baym et al., 2004). A corpus containing the "acts of speech" of a subject population would provide information about perhaps an equivalently "primary" expression of their social network.

Such data would also be unlike any other speech data previously recorded. It would capture truly spontaneous speech that arises *in situ* as people enact their actual, lived relationships. For that reason, we refer to such data as *situated speech data*—data that is gathered "in the wild"—to contrast it with other speech data recorded in constrained or contrived settings.

Of course, obstacles to gathering situated spontaneous speech still remain, and perhaps no other obstacle is as prominent as privacy. To collect data that captures truly natural interactions while providing a full picture of a social network, people must be recorded as they freely go about their lives. Requiring such unconstrained recording gives rise to two problems. First, uninvolved parties could be recorded without their consent—a scenario that, if raw audio is involved, is always unethical and often illegal. Second, people may change their behavior if they know they are being recorded. For both of those reasons, a level of privacy must be maintained. Ideally, a privacy-sensitive recording technique will process incoming audio in order to discard any information deemed too invasive while still preserving data useful for sociological inquiry. (One such method is described in Chapter 3.)

This necessity to discard audio information illuminates what is perhaps a fundamental trade-off between quantity and quality when automatically recording "direct observations" of social behavior. Subjects are unlikely to consent to large-scale, unrestricted recording of their behavior. In order to gather enough data to reliably observe an entire network (perhaps over a long period of time) some potentially useful information must be destroyed.

1.1.2 Measurement Error

That trade-off between the quality of data collected—how rich and detailed it is—and the quantity is also an expression of the fact that with any new measurement method there come new sources of measurement error. Situated speech data, and social behavior data in general, present both new challenges and opportunities with regard to measurement error.

First, as mentioned above, subjects may change their behavior if they know they are being observed. That is true for any method of data collection, but it may be present in varying degrees depending on the method. Determining exactly how and how much behavior changes according to the observation method is ultimately an unanswerable question—there is no way to collect completely reliable data free of this observer effect. In this respect, one source of error remains unchanged from survey to behavior data.

Second, despite researchers' desire for "direct" observations of behavior uncontaminated by poor recall, automated collection methods still introduce their own errors: sensors fail, their raw measurements are noisy, and the inference algorithms for obtaining interpretable data from these noisy observations are imperfect. That said, since the measurement mechanism is automated and the result of a constrained and hopefully reproducible process, there is the possibility of quantifying this source of error much more precisely than could be done for the highly idiosyncratic and variable error caused by subject's poor recall. (Indeed, such quantifications will be presented in Chapter 2.)

1.2 Modeling Social Behavior

Newly available social behavior data is different from traditional, survey-based social network data in two important ways. First, behavior data is very fine-grained. We no longer have just a single bit of information about whether a relationship exists. Instead, we can observe exactly how that relationship is enacted: the duration and frequency of interaction, the locations and times of interactions, the words spoken or written, and, in the case of speech data, non-linguistic aspects like pitch and volume. A second novel aspect of behavior data is that it is naturally longitudinal. Behavior will obviously play out over time, and the longer it can be recorded the more information can be gathered about the temporal evolution of the network.

Unsurprisingly, most existing social network analysis techniques have been developed around the constrained types of survey data currently available. That data usually contains a single static snapshot of the ties that exist in a network with no information about the behavior that manifests itself along those ties. The corresponding methods of analysis are thus concerned solely with the global structure of the network, and not the local behavior within that structure.

There are existing methods for modeling social behavior, but those typically rise only to the level of the dyad (Pentland, 2007) or small interacting group (Gibson, 2005). On the few occasions that entire groups are modeled jointly (McCallum et al., 2007; Eagle et al., 2009), it is only to examine the immediate social context of each person and not the social network that spans the entire group. In general, behavior modeling is concerned with only the local behavior around one person, and not the global social structure within which that behavior occurs.

Social behavior data provides an entirely new view of *both* local, individual behavior and global social structure. New methods are needed to exploit the richness of this data. Network modeling methods can be extended "downward" to include more behavior data, and behavior models can be extended "upward" to include network information.

Bringing the two together will provide opportunities to address many new research questions: How do global structural properties of the social network relate to the local behavior that comprises the network ties? For example, how does our network position affect how we behave and how others behave toward us? Do we interact differently within our close relationships? Does a person change her behavior depending on the network position of those with whom she interacts? Do clusters of behavior correspond to sub-groups within the social network, or to types of relationships between people? Can behavior be used to predict social position? Can network structure be used to predict how two people will interact?

And of course, since behavioral data is naturally longitudinal, questions about network formation and evolution can be addressed: How do changes in behavior relate to changes in the network structure? Can behavior predict which social ties will form, persist, or dissolve? Do properties of the network—its density, or tendency towards transitivity—change over time, or remain relatively constant?

1.3 Outline

The work presented in this thesis represents a small effort at answering some of these questions and addressing some of the challenges involved.

The remainder of this chapter provides an overview of the three areas common to all subsequent chapters: social networks, exponential families, and graphical models. It also describes the notation and terminology used throughout the rest of the paper.

Chapters 2 and 3 then cover the automated measurement of face-to-face conversations. Chapter 2 discusses basic methods for discovering physically colocated and conversing people from audio data that has been destructively pre-processed to protect privacy. Chapter 3 covers the University of Washington Spoken Networks project: a year-long effort that collected a corpus of such privacy-sensitive. Chapter 3 also explores some basic properties of the social network and behavior found in the data using the methods of chapter 2.

Chapters 4 and 5 concern new methods for modeling networks of social behavior and cover the application of these models to the conversation data of chapter 3. Chapter 4 discusses several extensions to one specific method for social network analysis (exponential random graph models) to enable them to exploit rich behavior data. Chapter 5 presents two methods—one simple, one not—for modeling the change observed in a person's behavior with different conversational partners and the relationship of that change to her network position.

Finally, chapter 6 presents concluding remarks and ideas for future work.

1.4 Background

This section provides introductions to general modeling techniques and notation that are common to the rest of the paper. Since later chapters cover many different measuring and modeling techniques, each with its own history in the literature, discussion of more specific related works is deferred until the relevant chapter.

1.4.1 Notation and Terminology

First come the minutiae of notation. Random variables are written in uppercase with a calligraphic font $\mathcal{Y}$ and will be bold if they are multivariate $\boldsymbol{\mathcal{Y}}$. Similarly, vectors are bold lowercase plain $\mathbf{y}$ and matrices are bold uppercase plain $\mathbf{Y}$. Scalars are medium weight italic y as are scalar components of vectors y_i and matrices y_{ij}. Bold vectors with subscripts are used to indicate a subset c of the components of the vector, either realized $\mathbf{y}_c$ or as random variables $\boldsymbol{\mathcal{Y}}_c$. The components of the vector not selected by c are denoted $\mathbf{y}_{\backslash c}$. If a subset of the components of $\mathbf{y}$ are to have their values changed to those found in another vector $\mathbf{y}'$, that will denoted as $(\mathbf{y}'_c \cup \mathbf{y}_{\backslash c})$.

Probabilities are written as $p(\boldsymbol{\mathcal{Y}} = \mathbf{y}|\boldsymbol{\theta})$ where $\boldsymbol{\theta}$ are learnable parameters. The usual shorthand of replacing $p(\boldsymbol{\mathcal{Y}} = \mathbf{y}|\boldsymbol{\theta})$ with $p(\mathbf{y}|\boldsymbol{\theta})$ will be employed when there is no ambiguity.

Generally, a distribution will be specified as a function of both data, $\mathbf{y}$, and parameters $\boldsymbol{\theta}$: $p(\mathbf{y}|\boldsymbol{\theta}) = g(\mathbf{y}, \boldsymbol{\theta})$. A specific form of g will be called a *model*. A model together with a specific value of $\boldsymbol{\theta}$ define a *distribution* over $\boldsymbol{\mathcal{Y}}$. Thus, a model alone defines a *family* of distributions indexed by $\boldsymbol{\theta}$. Constraints may be placed on the form of g (e.g. allowing only log-linear functions), and those constraints will then define a *class* of distributions [1].

1.4.2 Graphs and Social Networks

A graph $G = (V, E)$ consists of a set of nodes (or vertices) V and a set E of edges. If the graph is directed, E contains ordered pairs $(i, j) \in V \times V$ where $i \neq j$. If the graph is undirected, the pairs in E are unordered. Let $n = |V|$ denote the size of the graph. The *dyads* $D = \{\{i, j\}|i, j \in V \wedge i \neq j\}$ of the graph are all $\binom{n}{2}$ unordered pairs of unique vertices. For a directed graph, there are two potential edges per dyad, while an undirected graph has only one potential edge per dyad. Figure 1.1 shows a simple network of 6 undirected ties between 5 nodes.

A graph can be written as an adjacency matrix $\mathbf{Y}$ where component Y_{ij} corresponds to edge (i, j). For a binary graph (one in which edges either exist or do not) $Y_{ij} = 1$ if edge (i, j) appears in the graph and $Y_{ij} = 0$ if (i, j) is not in the graph. When used in contexts requiring a vector, $\mathbf{Y}$ is implicitly transformed into the vector $\mathbf{y} = (Y_{11}, Y_{12}, \ldots Y_{1n}, Y_{21}, Y_{22}, \ldots Y_{2n}, \ldots Y_{n1}, \ldots Y_{nn})^{\mathsf{T}}$. For undirected graphs, $\mathbf{Y}$ can be considered to be symmetric, though only its upper triangle $Y_{ij} : i < j$ will be used.

A subgraph of G is a set of nodes $V' \subseteq V$ and edges $E' \subseteq E$ where $(i, j) \in E' \iff (i, j) \in E$. Similarly, the adjacency matrix for a subgraph is the block of components Y_{ij} from $\mathbf{Y}$ where $i, j \in V'$. It is sometimes

[1] The class is perhaps more precisely a class of families, but "class of distributions" seems to be the more common term

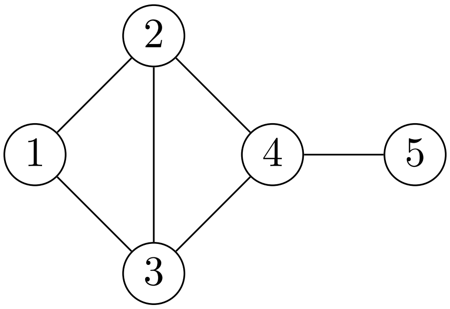


Figure 1.1: A simple network between 5 nodes.

easier to denote a subgraph by its set of possible edges than its set of nodes. For example, $\{(i, j), (j, k), (i, k)\}$ denotes the potential triangle between nodes i, j, and k.

When used to describe a social network, the nodes of a graph correspond (usually) to people and the edges correspond to some relationship between people. For a social network, edges are often called *ties* and the nodes are referred to as *actors*. To avoid confusion with graphs that do not represent people and their relationships, this paper will (attempt to!) use the terms *network*, actor, and tie when referring to social network graphs—graphs where the nodes are people and the ties are interpersonal relationships.

1.4.3 Exponential Families

The class of exponential families are those models whose densities can be written in the form

$$p(\mathcal{Y} = \mathbf{y}|\boldsymbol{\theta}) = h(\mathbf{y})\frac{1}{Z(\boldsymbol{\theta})}e^{\boldsymbol{\eta}(\boldsymbol{\theta})^{\mathsf{T}}\mathbf{f}(\mathbf{y})} \tag{1.1}$$

$\mathcal{Y}$ are the random variables being modeled with domain $\mathfrak{Y}$. Observed data is of the form $\mathcal{Y} = \mathbf{y}$ and $\mathbf{f}(\mathbf{y}) : \mathfrak{Y} \to F$, $F \subseteq \mathbb{R}^p$, is a deterministic function of statistics (or *features*) of the data. $\boldsymbol{\theta} \in \Theta$, $\Theta \subseteq \mathbb{R}^q$, are the parameters of the model that are to be learned and $\boldsymbol{\eta}(\boldsymbol{\theta}) : \Theta \to H$, $H \subseteq \mathbb{R}^p$, is a function that transforms the parameter.

$$Z(\boldsymbol{\theta}) = \int_{\mathbf{y}\in\mathfrak{Y}} h(\mathbf{y})e^{\boldsymbol{\eta}(\boldsymbol{\theta})^{\mathsf{T}}\mathbf{f}(\mathbf{y})}\,\mathrm{d}\nu(\mathbf{y}) \tag{1.2}$$

is the normalizing constant, or *partition function*, that ensures that the distribution properly sums to one. $\nu(\mathbf{y})$ is a dominating measure (i.e. $\forall\mathbf{y}\,\nu(\mathbf{y}) = 0 \to e^{\boldsymbol{\eta}(\boldsymbol{\theta})^{\mathsf{T}}\mathbf{f}(\mathbf{y})} = 0$) which can ensure that (1.2) is finite. $h(\mathbf{y})$ is an additional adjustment to ν (which also sometimes helps to ensure normalization e.g. $\binom{n}{k}$ for a binomial distribution with n trials and k successes). h can be incorporated into a modified version of ν (Kass and Vos, 1997, Ch. 2) and thus be omitted (which it will be, for the rest of this paper). Additionally, all models considered in this paper have finite, discrete $\mathfrak{Y}$ with ν as a simple counting measure. Thus it too will be omitted for clarity.

If neither $\mathbf{f}(\mathbf{y})$ nor $\boldsymbol{\eta}(\boldsymbol{\theta})$ have any linear dependencies in their components then the family is *minimal*. The *natural parameter space* $G \subseteq \mathbb{R}^p$ is defined as that containing all points $\boldsymbol{\eta}$ such that (1.2) is finite. If the family is minimal, and $q = p$ and $H = G$ then the family is *full*. If the family is full and if G is open then the family is *regular* (Brown, 1986, Ch. 1).

Most applied models in the machine learning and social networks literature are parameterized directly in natural parameter space. Additionally, many of them model a $\mathfrak{Y}$ that is discrete and finite. With those assumptions, (1.1) simplifies to the more well-used form

$$p(\mathbf{y}|\boldsymbol{\eta}) = \frac{1}{Z(\boldsymbol{\eta})} e^{\boldsymbol{\eta}^{\mathsf{T}}\mathbf{f}(\mathbf{y})} \tag{1.3}$$

$$Z(\boldsymbol{\eta}) = \sum_{\mathbf{y}\in\mathfrak{Y}} e^{\boldsymbol{\eta}^{\mathsf{T}}\mathbf{f}(\mathbf{y})} \tag{1.4}$$

The log-likelihood of (1.3) is then

$$\mathcal{L}(\boldsymbol{\eta}|\mathbf{y}) = \boldsymbol{\eta}^{\mathsf{T}}\mathbf{f}(\mathbf{y}) - \log Z(\boldsymbol{\eta}) \tag{1.5}$$

which has first and second derivatives

$$\frac{\partial}{\partial\boldsymbol{\eta}}\mathcal{L}(\boldsymbol{\eta}|\mathbf{y}) = \mathbf{f}(\mathbf{y}) - \underset{\mathbf{y}}{\mathrm{E}}\left[\mathbf{f}(\mathbf{y})\middle|\boldsymbol{\eta}\right] \tag{1.6}$$

$$\frac{\partial^2}{\partial\boldsymbol{\eta}^2}\mathcal{L}(\boldsymbol{\eta}|\mathbf{y}) = -\underset{\mathbf{y}}{\mathrm{Cov}}\left[\mathbf{f}(\mathbf{y})\middle|\boldsymbol{\eta}\right] \tag{1.7}$$

Note that since (1.4) is a finite sum its value is finite for any finite parameter value $\eta_i \in (-\infty, \infty)$. Thus $\Theta = H = G = \mathbb{R}^p$ and the family is regular. The covariance matrix in (1.7) is also the *Fisher information* $\boldsymbol{I}(\boldsymbol{\eta}) = \mathrm{E}_{\mathbf{y}}\left[\frac{\partial^2}{\partial\boldsymbol{\eta}^2}\mathcal{L}(\boldsymbol{\eta}\middle|\mathbf{y})\right]$. If the family is minimal then there is a single, unique maximum likelihood estimate, or MLE (Brown, 1986, Ch. 1):

$$\hat{\boldsymbol{\eta}} = \underset{\boldsymbol{\eta}}{\mathrm{argmax}}\,\mathcal{L}(\boldsymbol{\eta}|\mathbf{y}) \tag{1.8}$$

Since a covariance matrix is always positive semidefinite, the log-likelihood is convex and $\hat{\boldsymbol{\eta}}$ can (in theory) be found using gradient-based optimization methods.

In addition to the natural parameter space, it is useful to consider the *feature space* or *mean-value space*. The feature space is also of dimension p but is defined by F, the image of $\mathbf{f}(\mathfrak{Y})$, rather than H. Every point in natural parameter space has a one-to-one mapping to a point in the interior of the convex hull of F in feature space. Using that mapping, any regular exponential family can be parameterized either through its natural parameterization or its mean-value parameterization (Barndorff-Nielsen, 1978, Ch. 8). The derivative of the log partition function defines the mapping from natural parameter space to mean-value space, as seen in (1.6).

If the observed features are not in the interior of the convex hull of A then the naturally parameterized MLE does not exist (Barndorff-Nielsen, 1978, Ch. 9). In practice, this is a problem for points that lie on the boundary of the convex hull of A. Unfortunately, that can be the case for discrete and finite $\mathcal{Y}$ (Brown, 1986, Ch. 5). Intuitively, that means that the natural parameter MLE does not exist for any data that has an extreme value for some feature $f_i(\mathbf{y})$. For example, fitting a binomial to observed data containing only successes entails a mean-value parameter of 1 but a natural parameter of ∞.

The mean-value parameters are easy to interpret: they are the expected feature values for the distribution. The natural parameters also lend themselves to very straight-forward interpretation: η_i is the log-odds of a unit increase in $f_i(\mathbf{y})$, if all other features are held equal. As such, the absolute value of a parameter is the effect size of that feature—how important it is to the data—and the sign of the parameter shows whether increases in the feature make the data more (positive parameter value) or less (negative parameter) likely.

Standard asymptotic theory (Lehmann and Casella, 1998, Ch. 6) holds that the sampling distribution of $\hat{\boldsymbol{\eta}}$ is asymptotically normally distributed with a mean equal to the "true" parameters $\boldsymbol{\eta}^*$ and covariance equal to the inverse of the Fisher information, scaled by the sample size N:

$$\hat{\boldsymbol{\eta}} \to \mathcal{N}\left(\boldsymbol{\eta}^*, \frac{1}{N}\boldsymbol{I}^{-1}(\boldsymbol{\eta}^*)\right) \tag{1.9}$$

That fact can be used to test whether learned parameters are significantly different from zero and thus whether any of the features being used have statistically significant effects in the data.

Linear Parameter Constraints

Frequently, different components of $\mathbf{f}(\mathbf{y})$ compute the same statistic but from different subsets of $\mathbf{y}$. For example, imagine that $\mathbf{y}$ is a sequence of real numbers and we wish to model the probability that adjacent components of $\mathbf{y}$ take the same sign. The same feature is computed for all adjacent pairs y_i and y_{i+1}: $f_i(\mathbf{y}) = \mathbb{1}_{[\text{sgn}(y_i)=\text{sgn}(y_{i+1})]}$. If the distribution is assumed to be homogeneous (implying that any adjacent components are equally as likely or unlikely to have the same sign, regardless of their position in the vector), then there will be only one parameter that is shared (or tied) across all components of $\mathbf{f}(\mathbf{y})$.

Tying parameters so that they must be equal is a special case of putting a linear constraint on the parameters. In that case

$$\boldsymbol{\eta}(\boldsymbol{\theta}) = \boldsymbol{A}\boldsymbol{\theta} \tag{1.10}$$

where $\mathbf{A}$ is a $p \times q$ matrix with $a_{ij} = 1$ if the i-th feature is to get the j-th parameter and $a_{ij} = 0$ otherwise. This constrains Θ to a hyperplane of dimension q within G and the family is thus no longer full.

However, the constraint also entails that (1.1) can be re-written as

$$p(\mathbf{y}|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} e^{(\mathbf{A}\boldsymbol{\theta})^\mathsf{T}\mathbf{f}(\mathbf{y})} \tag{1.11}$$

$$= \frac{1}{Z(\boldsymbol{\theta})} e^{\boldsymbol{\theta}^\mathsf{T}\mathbf{A}^\mathsf{T}\mathbf{f}(\mathbf{y})} \tag{1.12}$$

Which shows that we can then define a new feature function $\mathbf{f}'(\mathbf{y}) = \mathbf{A}^\mathsf{T}\mathbf{f}(\mathbf{y})$ which reduces the number of features from p to q (replacing indicators with counts, in the above example) and allows the entire model to be written in the same form as (1.3). The rewritten model is once again full (if it is minimal) and if Θ is open than it has all the desirable properties of a regular family: a convex log-likelihood and unique MLE.

Clearly, $\mathbf{A}$ can contain values other than just ones and zeros to encode linear parameter constraints beyond simple equality.

Curved Exponential Families

If $\boldsymbol{\eta}(\boldsymbol{\theta})$ is a non-linear function and $q < p$, then (with some regularity conditions) H is a q-dimensional curved manifold in G. For that reason, Efron (1975) calls such distributions *curved exponential families*.

The distribution defined by a curved exponential family for a finite and discrete $\mathfrak{Y}$ is

$$p(\mathbf{y}|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} e^{\boldsymbol{\eta}(\boldsymbol{\theta})^\mathsf{T}\mathbf{f}(\mathbf{y})} \tag{1.13}$$

$$Z(\boldsymbol{\theta}) = \sum_{\mathbf{y}\in\mathfrak{Y}} e^{\boldsymbol{\eta}(\boldsymbol{\theta})^\mathsf{T}\mathbf{f}(\mathbf{y})} \tag{1.14}$$

The log-likelihood, gradient, and Hessian for a curved exponential family are

$$\mathcal{L}(\boldsymbol{\theta}|\mathbf{y}) = \boldsymbol{\eta}(\boldsymbol{\theta})^\mathsf{T}\mathbf{f}(\mathbf{y}) - \log Z(\boldsymbol{\theta}) \tag{1.15}$$

$$\frac{\partial}{\partial\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}|\mathbf{y}) = \boldsymbol{\nabla\eta}(\boldsymbol{\theta})^\mathsf{T}\left(\mathbf{f}(\mathbf{y}) - \underset{\mathbf{y}}{\mathrm{E}}\left[\mathbf{f}(\mathbf{y})\middle|\boldsymbol{\theta}\right]\right) \tag{1.16}$$

$$\frac{\partial^2}{\partial\boldsymbol{\theta}^2}\mathcal{L}(\boldsymbol{\theta}|\mathbf{y}) = \mathbf{B} - \boldsymbol{\nabla\eta}(\boldsymbol{\theta})^\mathsf{T}\underset{\mathbf{y}}{\mathrm{Cov}}\left[\mathbf{f}(\mathbf{y})\middle|\boldsymbol{\theta}\right]\boldsymbol{\nabla\eta}(\boldsymbol{\theta}) \tag{1.17}$$

where $\boldsymbol{\nabla\eta}(\boldsymbol{\theta}) \triangleq \frac{\partial}{\partial\boldsymbol{\theta}}\boldsymbol{\eta}(\boldsymbol{\theta})$ is the $p \times q$ Jacobian of $\boldsymbol{\eta}(\boldsymbol{\theta})$, and $B_{ij} = \left(\frac{\partial^2}{\partial\theta_j\partial\theta_i}\boldsymbol{\eta}(\boldsymbol{\theta})\right)^\mathsf{T}\left(\mathbf{f}(\mathbf{y}) - \mathrm{E}_\mathbf{y}\left[\mathbf{f}(\mathbf{y})\,|\,\boldsymbol{\theta}\right]\right)$

Under certain regularity conditions, the map from $\boldsymbol{\theta}$ to $\boldsymbol{\eta}(\boldsymbol{\theta})$ is an *embedding* (Kass and Vos, 1997, Ch. 2). For that reason, call $\Theta \subseteq \mathbb{R}^q$ the *embedded parameter space*.

By definition, a curved exponential family is no longer a full family. As a consequence, multiple values in feature space will map to the same value in the embedded parameter space. Specifically, all values of $\mathbf{y}$ with (1.16) equal to 0 will have the same MLE $\hat{\boldsymbol{\theta}}$. (Kass and Vos call that set of $\mathbf{y}$ values the *auxiliary space* of $\hat{\boldsymbol{\theta}}$.) That is useful, however, for models in danger of being overparameterized and thus having e.g. some $f_i = 0$ for many data sets. $\mathbf{f}$ would then lie on the boundary of the convex hull of F. So while the unconstrained MLE does not exist in natural parameter space, the constrained MLE does exist in the embedded parameter space.

The term to the right of the minus in (1.17) is the Fisher information $\boldsymbol{I}(\boldsymbol{\theta})$. Unlike (1.5), (1.15) is not, in general, convex. The asymptotic normality of the MLE in (1.9) still holds for a curved exponential family (Kass and Vos, 1997, Ch. 2).

1.4.4 Graphical Models of Exponential Families

Distributions in the form of (1.1) can clearly be represented as a product of a set of *factors*:

$$p(\mathbf{y}|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \prod_{c \in C} e^{\boldsymbol{\eta}_c(\boldsymbol{\theta})\mathbf{f}_c(\mathbf{y}_c)} \tag{1.18}$$

Frequently, a factorized feature function $\mathbf{f}_c$ involves only a subset $\mathcal{Y}_c$ of the variables in $\mathcal{Y}$ (as expressed in (1.18)). When that is the case, the factorization entails a set of conditional independence assumptions about the distribution. Those assumptions can be encoded by a *graphical model* (Lauritzen, 1996; Koller and Friedman, 2009). A graphical model is a graph with one node for each variable in the distribution, and with edges that express the absence of conditional independencies between variables. The edges may be either undirected or directed, leading to different descriptions of the family being encoded.

Undirected Graphical Models

In an undirected graphical model, two variables are conditionally independent, given all other variables, if there is no edge between them. Formally, let $\text{ne}(\mathcal{Y}_i)$ denote the neighbors of $\mathcal{Y}_i$ in the graphical model. $\text{ne}(\mathcal{Y}_i)$ is the *Markov blanket* of $\mathcal{Y}_i$ and Y_i is conditionally independent of all other variables given its Markov blanket.

An undirected graphical model can be constructed from a factorized distribution by adding a clique to the graph for each factor. The joint distribution defined by the graphical model can then be expressed as (1.18) where C is a set of cliques in the graph and $\mathcal{Y}_c$ are the variables in clique c.

Factor Graphs The combination of cliques induced by a set of factors can produce an undirected graphical model that ultimately has larger cliques than those explicitly defined via the factors. In other words, the factor cliques are not necessarily maximal with respect to the implied dependency graph. A consequence of this is that the dependency graph alone does not provide enough information to reconstruct the factorization present in the original model. To remedy that, the model can be also be represented with a bipartite *factor graph*. One set of nodes in the factor graph are the same variable nodes as in an undirected model. The other set of nodes are factor nodes, one per explicitly defined factor. There are edges between a variable node and the nodes for all factors in which the variable appears. The factor graph thus preserves the "visibility" of the explicit factors and can aid in the representation of different models whose undirected graphical representations may otherwise be identical.

Directed Graphical Models

The joint distribution defined by a directed graphical model is

$$p(\mathbf{y}|\boldsymbol{\theta}) = \prod_i p(\mathcal{Y}_i = y_i|\text{pa}(\mathcal{Y}_i) = \mathbf{y}_{\text{pa}(i)}) \tag{1.19}$$

where $\text{pa}(\mathcal{Y}_i)$ are the *parents* of $\mathcal{Y}_i$ in the graph: nodes with a directed edge to the node for $\mathcal{Y}_i$. $\mathbf{y}_{\text{pa}(i)}$ denotes the values of $\text{pa}(\mathcal{Y}_i)$ in $\mathbf{y}$. The graph must be acyclic. The parameters $\boldsymbol{\theta}$ are now the parameters of fully normalized conditional distributions defined for each child-and-parents set $\{\mathcal{Y}_i, \text{pa}(\mathcal{Y}_i)\}$.

The distribution defined by a directed graphical model and its parameters may be transformed into a distribution defined by an undirected model (and its parameters) by creating a clique $c = \{\mathcal{Y}_i \cup \text{pa}(\mathcal{Y}_i)\}$ for all $\mathcal{Y}_i$. The new clique now has edges between all $\text{pa}(\mathcal{Y}_i)$ and so the operation is referred to as "marrying the parents" or moralizing the graph. $\boldsymbol{\eta}(\boldsymbol{\theta})$ and $\mathbf{f}_c(\mathbf{y}_c)$ must then be arranged so that the factor for the clique has value $p(\mathcal{Y}_i|\text{pa}(\mathcal{Y}_i))$. For example, $\eta_{c_k}(\boldsymbol{\theta}) = \log p(\mathcal{Y}_i = x|\text{pa}(\mathcal{Y}_i) = \mathbf{z})$ and $f_{c_k}(\mathbf{y}_c) = \mathbb{1}_{[y_i = x \wedge \text{pa}(\mathcal{Y}_i) = \mathbf{z}]}$ with k indexing all possible assignments of $\mathcal{Y}_i$ and $\text{pa}(\mathcal{Y}_i)$ (assuming all variables are discrete). Note though, that this representation is not a minimal family since $\sum_x p(\mathcal{Y}_i = x|\mathbf{y}_{\text{pa}(i)})$ must equal 1. However, because of that constraint the local factor values are fully normalized conditional distributions, and the partition function is equal to 1.

1.4.5 Learning Parameters from Data

Most general exponential family models do not have an analytic solution for finding the $\hat{\boldsymbol{\theta}}$ that maximizes (1.1) so iterative methods must be used. Even then, because of the intractability of $Z(\boldsymbol{\theta})$, exact calculations of the log-likelihood, gradient, and Hessian are not feasible. A number of approximation techniques have been developed for this problem, but this paper only considers the two that are used most often for exponential family models of social networks: maximum pseudolikelihood estimation, and Markov Chain Monte Carlo approximations of the log-likelihood's gradient.

Maximum Pseudolikelihood Estimation

Besag (1975) introduced maximum pseudolikelihood estimation in the context of lattice-structured undirected graphical models applied to spatial data. It can easily be applied to most distributions of the form of (1.1), particularly when the variables are finite and discrete.

The pseudo-loglikelihood is defined as

$$\mathcal{PL}(\boldsymbol{\theta}|\mathbf{y}) = \sum_i \log p(\mathcal{Y}_i = y_i|\mathbf{y}_{\setminus i}) \tag{1.20}$$

$$= \sum_i \boldsymbol{\eta}(\boldsymbol{\theta})^{\mathsf{T}}\mathbf{f}(\mathbf{y}) - \log \sum_{y'_i \in \mathfrak{y}_i} e^{\boldsymbol{\eta}(\boldsymbol{\theta})^{\mathsf{T}}\mathbf{f}(y'_i \cup \mathbf{y}_{\setminus i})} \tag{1.21}$$

with gradient

$$\frac{\partial}{\partial \boldsymbol{\theta}}\mathcal{PL}(\boldsymbol{\theta}|\mathbf{y}) = \boldsymbol{\nabla}\boldsymbol{\eta}(\boldsymbol{\theta})^{\mathsf{T}} \left(\sum_i \mathbf{f}(\mathbf{y}) - \operatorname*{E}_{y'_i}\left[\mathbf{f}(y'_i \cup \mathbf{y}_{\setminus i}) \,|\, \mathbf{y}_{\setminus i}\right] \right) \tag{1.22}$$

$Z(\boldsymbol{\theta})$ has conveniently cancelled out and the complexity of (1.21) is $O(N)$ compared to $O(2^N)$ for (1.4). The maximum pseudolikelihood estimate, or MPLE, $\tilde{\boldsymbol{\theta}} = \operatorname{argmax}_{\boldsymbol{\theta}} \mathcal{PL}(\boldsymbol{\theta}|\mathbf{y})$, can thus be tractably found. In models where all $\mathcal{Y}_i$ are marginally independent and $p(\mathcal{Y}_i|\mathbf{y}_{\setminus i}) = p(\mathcal{Y}_i)$ the MPLE is equal to the MLE.

Markov Chain Monte Carlo Approximations

While the expectation required for (1.6) or (1.16) is intractable, it could be approximated with samples drawn from $p(\boldsymbol{\mathcal{Y}}|\boldsymbol{\theta})$. Unfortunately, sampling from the model distribution is also almost always intractable. However, we can use a Metropolis-Hastings scheme (Hastings, 1970; Robert and Casella, 2004, ch. 7) to construct a Markov chain that has $p(\boldsymbol{\mathcal{Y}}|\boldsymbol{\theta})$ as its stationary distribution.

A Metropolis-Hastings chain requires a *proposal distribution* $q(\boldsymbol{\mathcal{Y}} = \mathbf{y}'|\mathbf{y})$ that is easier to sample from than $p(\boldsymbol{\mathcal{Y}}|\boldsymbol{\theta})$. Assuming that the chain is at some state $\mathbf{y}$, it proceeds by drawing a $\mathbf{y}'$ from $q(\boldsymbol{\mathcal{Y}} = \mathbf{y}'|\mathbf{y})$. Then, with probability $A(\mathbf{y}, \mathbf{y}')$, $\mathbf{y}'$ is either accepted as the next state in the chain or rejected, in which case the current $\mathbf{y}$ is retained as the next state.

The probability of accepting $\mathbf{y}'$ is

$$A(\mathbf{y}, \mathbf{y}') = \min\left(1,\ \frac{p(\mathbf{y}'|\boldsymbol{\theta})q(\mathbf{y}|\mathbf{y}')}{p(\mathbf{y}|\boldsymbol{\theta})q(\mathbf{y}'|\mathbf{y})}\right) \tag{1.23}$$

$$= \min\left(1,\ \frac{e^{\boldsymbol{\eta}(\boldsymbol{\theta})^{\mathsf{T}}\mathbf{f}(\mathbf{y}')}q(\mathbf{y}|\mathbf{y}')}{e^{\boldsymbol{\eta}(\boldsymbol{\theta})^{\mathsf{T}}\mathbf{f}(\mathbf{y})}q(\mathbf{y}'|\mathbf{y})}\right) \tag{1.24}$$

$$\log A(\mathbf{y}, \mathbf{y}') = \min\left(0,\ \boldsymbol{\eta}(\boldsymbol{\theta})^{\mathsf{T}}\left[\mathbf{f}(\mathbf{y}') - \mathbf{f}(\mathbf{y})\right] + \log\frac{q(\mathbf{y}|\mathbf{y}')}{q(\mathbf{y}'|\mathbf{y})}\right) \tag{1.25}$$

Again, $Z(\boldsymbol{\theta})$ has cancelled out leaving a computable quantity. Furthermore, for some features the difference in (1.25) can often be computed very quickly, particularly if q only changes a few components of $\mathbf{y}$.

Using samples $\mathbf{y}^1, \ldots, \mathbf{y}^M$ drawn from this chain, the expectation required for the gradient can be approximated as

$$\operatorname*{E}_{\mathbf{y}}\left[\mathbf{f}(\mathbf{y})\right] \approx \frac{1}{M}\sum_i \mathbf{f}(\mathbf{y}^i) \tag{1.26}$$

Gibbs Sampling A special case of Metropolis-Hastings is *Gibbs sampling*. In Gibbs sampling the proposal distribution modifies only a subset of all the variables and that subset is drawn from its true conditional distribution:

$$q\left(\mathbf{y}_i' \cup \mathbf{y}_{\setminus i} \middle| \mathbf{y}\right) = p(\mathcal{Y}_i = \mathbf{y}_i' | \mathbf{y}_{\setminus i}) \tag{1.27}$$

The acceptance ratio will always be 1

$$\begin{aligned} A(\mathbf{y}, \mathbf{y}_i' \cup \mathbf{y}_{\setminus i}) &= \frac{p(\mathbf{y}_i' \cup \mathbf{y}_{\setminus i}) p(\mathbf{y}_i | \mathbf{y}_{\setminus i})}{p(\mathbf{y}_i \cup \mathbf{y}_{\setminus i}) p(\mathbf{y}_i' | \mathbf{y}_{\setminus i})} \qquad (1.28) \\ &= \frac{p(\mathbf{y}_i' | \mathbf{y}_{\setminus i}) p(\mathbf{y}_{\setminus i}) p(\mathbf{y}_i | \mathbf{y}_{\setminus i})}{p(\mathbf{y}_i | \mathbf{y}_{\setminus i}) p(\mathbf{y}_{\setminus i}) p(\mathbf{y}_i' | \mathbf{y}_{\setminus i})} \qquad (1.29) \end{aligned}$$

and the sampling process is considerably simplified. Typically only a single variable is modified ($\mathbf{y}_i$ would be y_i). Clearly, any methods developed for pseudolikelihood computation can be easily repurposed for Gibbs sampling. Gibbs sampling is also well suited to inference in directed graphical models since their local factors are already specified in exactly the form needed for (1.28).

MCMC MLE and Importance Reweighting Geyer and Thompson (1992) were the first to propose an algorithm for general exponential family maximum likelihood estimation using (1.26). They showed that while the log-likelihood value at any $\boldsymbol{\theta}$ cannot be easily approximated, the change in log-likelihood can:

$$\mathcal{L}(\boldsymbol{\theta}|\mathbf{y}) - \mathcal{L}(\boldsymbol{\theta}'|\mathbf{y}) = \left[\boldsymbol{\eta}(\boldsymbol{\theta}) - \boldsymbol{\eta}(\boldsymbol{\theta}')\right]^{\mathsf{T}} \mathbf{f}(\mathbf{y}) - \log \frac{Z(\boldsymbol{\theta}')}{Z(\boldsymbol{\theta})} \tag{1.30}$$

with

$$\begin{aligned} \frac{Z(\boldsymbol{\theta}')}{Z(\boldsymbol{\theta})} &= \sum_{\mathbf{y} \in \mathfrak{Y}} \frac{1}{Z(\boldsymbol{\theta})} e^{\boldsymbol{\eta}(\boldsymbol{\theta})^{\mathsf{T}} \mathbf{f}(\mathbf{y})} e^{\left[\boldsymbol{\eta}(\boldsymbol{\theta}') - \boldsymbol{\eta}(\boldsymbol{\theta})\right]^{\mathsf{T}} \mathbf{f}(\mathbf{y})} && (1.31) \\ &= \operatorname*{E}_{\mathbf{y}} \left[e^{\left[\boldsymbol{\eta}(\boldsymbol{\theta}') - \boldsymbol{\eta}(\boldsymbol{\theta})\right]^{\mathsf{T}} \mathbf{f}(\mathbf{y})} \middle| \boldsymbol{\theta} \right] && (1.32) \\ &\approx \frac{1}{M} \sum_i e^{\left[\boldsymbol{\eta}(\boldsymbol{\theta}') - \boldsymbol{\eta}(\boldsymbol{\theta})\right]^{\mathsf{T}} \mathbf{f}(\mathbf{y}^i)} \qquad \text{with } \mathbf{y}^1, \ldots, \mathbf{y}^M \sim p(\mathcal{Y}|\boldsymbol{\theta}) && (1.33) \end{aligned}$$

Since (1.30) will attain its maximum at the same point as (1.15) it can be used, together with a similar approximation of (1.16) and (1.17), to approximately optimize (1.15) with any iterative, gradient-based approach.

However, gathering samples from a Markov chain during every step of an iterative optimization procedure is very computationally expensive. Geyer and Thompson suggest using importance sampling (Robert and Casella, 2004, Ch. 3) to reweight already collected samples, allowing them to be reused for many gradient steps:

$$\begin{aligned}
\underset{\mathbf{y}}{\mathrm{E}}\left[\mathbf{f}(\mathbf{y})|\boldsymbol{\theta}'\right] &= \underset{\mathbf{y}}{\mathrm{E}}\left[\frac{p(\mathbf{y}|\boldsymbol{\theta}')}{p(\mathbf{y}|\boldsymbol{\theta})}\mathbf{f}(\mathbf{y})\middle|\boldsymbol{\theta}\right] && (1.34)\\
&= \sum_{y\in\mathfrak{Y}} \frac{Z(\boldsymbol{\theta})}{Z(\boldsymbol{\theta}')} e^{\left[\boldsymbol{\eta}(\boldsymbol{\theta}')-\boldsymbol{\eta}(\boldsymbol{\theta})\right]^{\mathsf{T}}\mathbf{f}(\mathbf{y})}\mathbf{f}(\mathbf{y}) && (1.35)\\
&\approx \frac{\sum_{i=1}^{M} e^{\left[\boldsymbol{\eta}(\boldsymbol{\theta}')-\boldsymbol{\eta}(\boldsymbol{\theta})\right]^{\mathsf{T}}\mathbf{f}(\mathbf{y}^i)}\mathbf{f}(\mathbf{y}^i)}{\sum_{j=1}^{M} e^{[\boldsymbol{\eta}(\boldsymbol{\theta}')-\boldsymbol{\eta}(\boldsymbol{\theta})]^{\mathsf{T}}\mathbf{f}(\mathbf{y}^j)}} && (1.36)\\
&= \sum_{i=1}^{M}\left(\frac{e^{\left[\boldsymbol{\eta}(\boldsymbol{\theta}')-\boldsymbol{\eta}(\boldsymbol{\theta})\right]^{\mathsf{T}}\mathbf{f}(\mathbf{y}^i)}}{\sum_{j=1}^{M} e^{[\boldsymbol{\eta}(\boldsymbol{\theta}')-\boldsymbol{\eta}(\boldsymbol{\theta})]^{\mathsf{T}}\mathbf{f}(\mathbf{y}^j)}}\right)\mathbf{f}(\mathbf{y}^i) \qquad \text{with } \mathbf{y}^1\ldots\mathbf{y}^M \sim p(\boldsymbol{\mathcal{Y}}|\boldsymbol{\theta}) && (1.37)
\end{aligned}$$

By reweighting samples according to (1.37), the gradient (and Hessian) at any point $\boldsymbol{\theta}'$ can be approximated with samples drawn at another point $\boldsymbol{\theta}$. The approximation deteriorates as $\boldsymbol{\theta}'$ and $\boldsymbol{\theta}$ get farther apart, so implementations often take a limited number of steps, or penalize steps too far from $\boldsymbol{\theta}$. After converging, another sample is taken and the process repeats. The resulting parameters are called the Markov chain Monte Carlo maximum likelihood estimate, or MCMC MLE.

Chapter 2

Privacy-Sensitive Conversation Modeling

When collecting situated conversation data it is necessary to protect the privacy of not just people who willingly consent to wear a recording device, but also of those who may happen to come within range of one the microphones. For that, we require some destructive processing of the audio that yields a feature set that does not allow us to reconstruct intelligible speech or infer the identities of anyone not wearing a device. A further constraint on the feature set is that all features must be computed in real-time within the limited computational resources of a wearable device—no raw audio should ever be stored, even temporarily.

At the same time, the features must still contain enough information to allow conversations to be found and meaningful inferences made about those conversations. Fortunately, the non-linguistic aspects of a conversation—who speaks when and for how long, how loud, and at what pitch—still allow for many useful analyses. Interruptions and speaking time reveal information about status and dominance (Hawkins, 1991). Speaking rate reveals information about a speaker's level of mental activity (Hurlburt et al., 2002). Energy (loudness) can reveal a person or group's interest in the conversation (Gatica-Perez et al., 2005). Pitch alone has a long history as a fundamental feature for inferring emotion (Dellaert et al., 1996), and energy and duration of voiced and unvoiced regions are also informative emotional features (Schuller et al., 2004).

This chapter presents a set of useful privacy-sensitive features that can be extracted from an audio stream in real-time (Section 2.1), along with methods for using those features to automatically determine who is in conversation with whom (Section 2.2) and *how* people are speaking in the conversation (Section 2.3).

Parts of this chapter were previously published in (Wyatt, Choudhury, Bilmes and Kautz, 2007), (Wyatt, Choudhury and Kautz, 2007), and (Wyatt, Choudhury and Bilmes, 2007).

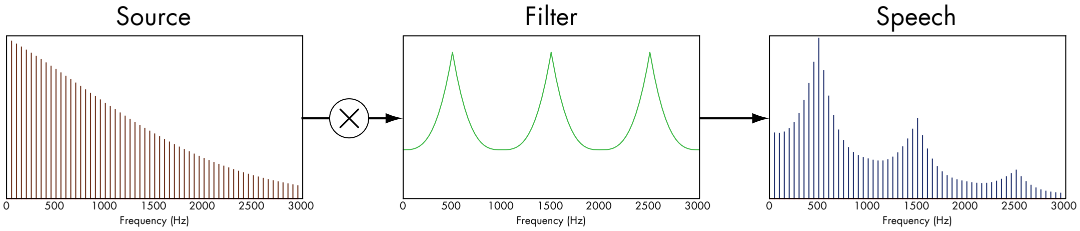


Figure 2.1: Conceptual schematic of the source-filter model.

Related Work To the best of our knowledge, prior to this research, there were only two existing methods for finding conversations in separately recorded streams of audio. The method proposed by Corman and Scott (1994) computes normalized cross-correlation between raw audio signals and concludes that two people are in a conversation if their correlation coefficients are above a threshold estimated from labeled data. Obviously, using raw audio does not protect privacy, but a privacy-sensitive variant of their method is considered below. Similarly, the method proposed by Basu (2002) computes the mutual information between binary signals that represent voiced/unvoiced speech and places two people in a conversation if their mutual information is above a pre-specified threshold. This work extends Basu's method in three important ways: (i) to handle multiperson conversation detection (not just dyadic), (ii) to operate at a finer time granularity while still producing a "smooth" inference over time, and (iii) to learn its threshold in an unsupervised manner.

2.1 Privacy-Sensitive Features

Following Basu, the approach we take to extracting non-linguistic speech information is founded on the ability to detect voiced human speech. A basic model for the production of human speech is the *source-filter model* (Quatieri, 2001) shown in Figure 2.1. As its name suggests, the source-filter model posits two independent components at work in the production of speech: (1) a source sound that is generated in the glottis and then passed through (2) the filter, provided by the vocal tract, that shapes the spectrum of the source.

The source can be *voiced* or unvoiced. If it is voiced, the vocal cords are vibrating at what is called the *fundamental frequency*, or F0, which is the pitch at which the person is speaking. A true sequence of speech will alternate rapidly between voiced and unvoiced segments. Prosodic features of speech—intonation, stress, duration—are described by how the fundamental frequency and energy (volume) change during speech.

The source sound is shaped into words by changing the shape of the vocal tract. It is the frequency response of the vocal tract, particularly the resonant peaks known as *formants*, that contains information about the phonemes that are the constituent parts of spoken words. Any processing of the audio that removes information about the formants will ensure that intelligible speech can not be synthesized from the information that remains.

Thus, to find conversations and retain information about how people are speaking, we save information about the source while discarding (almost) all information about the filter.

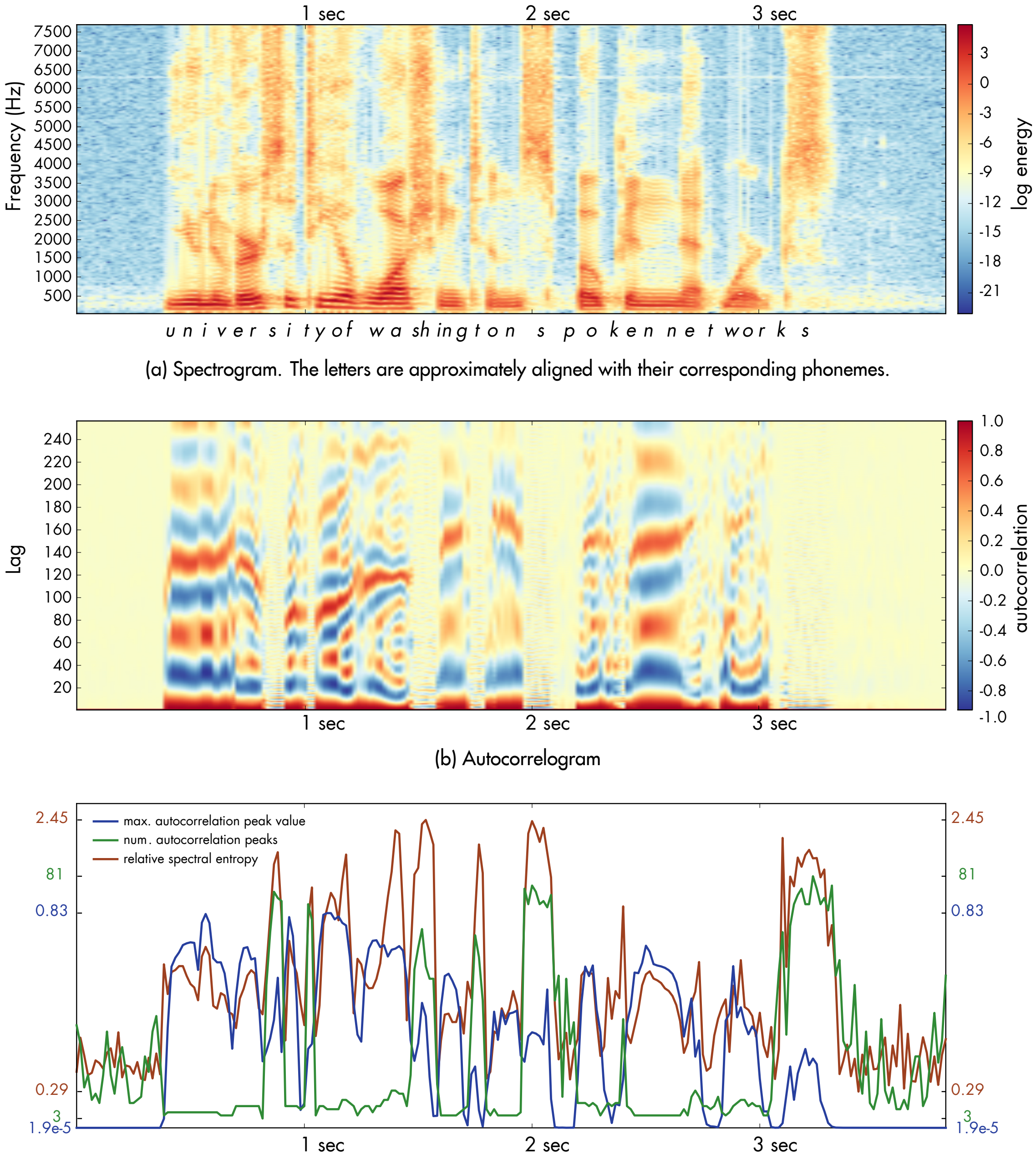
The first step in that process is finding voiced speech. Figure 2.2(a) shows the spectrogram for a male voice saying the phrase "University of Washington Spoken Networks." In a spectrogram, time runs along the x-axis and frequencies increase along the y-axis; color indicates energy at a given frequency (Solzhenitsyn, 1968). In this example all of the phonemes are voiced except those for "s," "t," "sh," "p," and "k." The strong harmonics are indicators of voiced speech and we take advantage of that harmonicity to find segments of voiced speech.

Three features that have been shown to be useful for robustly detecting voiced speech under varying noise conditions are: (1) non-initial maximum autocorrelation peak, (2) the total number of autocorrelation peaks and (3) relative spectral entropy (Basu, 2002). To provide an intuition for the the first two features, Figure 2.2(b) shows the autocorrelogram for the example phrase. As in the spectrogram, time runs along the x-axis. The y-axis shows increasing lags at which the autocorrelation is computed, and colors show the value of the autocorrelation. The voiced segments show fewer, stronger peaks.

All 3 features are shown in Figure 2.2(c). During voiced segments, the number of autocorrelation peaks drops, while the maximum autocorrelation value and relative spectral entropy rise.

The harmonicity in the spectrogram shows that voiced speech has a low spectral entropy, compared to non-voiced regions. However, in many environments there can be noise centered strongly at a specific frequency. Figure 2.2(a) shows two possible examples of such noise: a low frequency hum (from 300 to 500 Hz) that may be an air conditioner, and a sharp high frequency noise (around 6400 Hz) that is probably a computer fan or hard drive. Such narrow spectrum noise will lower the general environmental spectral entropy. The relative spectral entropy is the relative entropy (also known as Kullback-Leibler divergence, see Equation (2.1)) between an instantaneous normalized spectrum and the mean normalized spectrum for a much longer window of time. Relative spectral entropy captures the quick change in entropy caused by short segments of voiced speech while smoothing away any environmental reductions in entropy. Additionally, narrow spectrum noise can also create strong autocorrelation peaks. Fortunately, in settings where conversations can comfortable occur, such noise is usually low energy (compared to voiced speech) and its autocorrelation can be disrupted by adding low energy white noise to the signal.

The precise procedure for computing features is as follows: The 15360 Hz raw audio signal is split into frames of 512 samples (one 30th of a second) with overlaps of 256 samples (one 60th of a second). Those frames have their means subtracted and are multiplied with a Hamming window. A discrete Fourier transform is applied to each frame resulting in a 256 point spectrum. The absolute value of the spectrum is taken and



(a) Spectrogram. The letters are approximately aligned with their corresponding phonemes.

(b) Autocorrelogram

(c) Features used for detecting voiced speech. Values are scaled to fit, so y axis labels indicate minimum and maximum values for each feature.

Figure 2.2: Audio of a male voice saying "University of Washington Spoken Networks."

squared to yield a power spectrum. We save the sum of all values in the power spectrum as the *energy* of the frame. To disrupt low-energy, narrow spectrum noise, we uniformly whiten the power spectrum with additional energy equal to 1% of the maximum energy possible per frame. The inverse Fourier transform of the whitened power spectrum is then taken to find the autocorrelation of the frame at all lags (Gray and Davisson, 2004). The number of autocorrelation peaks (defined as positive regions between zero-crossings) is counted and the value and lag of the highest peak is saved (which naturally excludes the initial maximum at lag 0). A running mean of the normalized spectrum is kept for the last 500 frames ($\approx$ 8.33 seconds) and the relative entropy is computed between the current normalized spectrum and that running mean.

Altogether, we save 6 acoustic features: (i) value and (ii) lag of the non-initial maximum autocorrelation peak, (iii) the total number of autocorrelation peaks, (iv) instantaneous and (v) relative spectral entropy, and (vi) energy.

On the specific device we used (described in Section 3.2.1), all computations are carried out in the frequency domain using fixed point arithmetic. The logarithm required to compute entropy is not practical given the device's limited processing power. However, the device's comparatively large amount of RAM allows us to instead use a look-up table pre-populated with logarithms for all 16 bit values.

The energy is used later to determine who is speaking. The lag of the maximum autocorrelation peak is not needed for detecting voiced speech, but it is useful for determining a speaker's F0 (Rabiner, 1977). The peak will often correspond not to the exact F0 but instead to one of its harmonics. Formants are expressed through the attenuation of many of the harmonics present while letting only those near the resonant peaks of the vocal tract pass through. This means that at least one—and often more harmonics—will correspond to single formant. To reproduce speech intelligibly, information on at least three formants is required (Donovan, 1996). Since we save at most one harmonic, we believe that all of our features are privacy-sensitive and cannot be used to reconstruct intelligible speech.

2.2 Extracting Conversation Data

To gather data about face-to-face conversations, presumably multiple people will wear recording devices that each save separate streams of the privacy-sensitive features described above. After recordings have been made, all of the recordings must be combined and conversations must be found within the combined data. Finding conversations proceeds in four steps, each of which is described in a following section. First, we must find voiced speech in each person's recording (Section 2.2.1). Second, people must be partitioned into colocated groups where all the members of a group are considered "together" with each other and not together with any person in any other group (Section 2.2.2). Third, we must infer who is speaking when within each colocated group (Section 2.2.3). Finally, once colocated groups and speakers have been identified, we can conclude that people who are colocated and speaking are in conversation together and extract further features of their conversation (Section 2.3). Figure 2.3 provides an overview of the entire process.

Evaluation Data All of the techniques presented in the following sections were evaluated using a small set of labeled data collected using the same wearable devices as the large Spoken Networks corpus. To record this smaller data set, 5 people wore devices for just over 50 minutes while moving around a building and entering and leaving different conversations with one another. The participants were told where to go and whom to speak with, but were not told what to talk about. They are all friends and had no trouble filling the time with casual conversation. The two primary locations were a quiet meeting room and a loud and noisy public space (where most of the background noise is other speech), but conversations also occurred while the participants walked together and rode elevators between locations. In order to label the data, raw audio was saved for this small set. To test the performance of our methods in the presence of unmiked speakers, we selectively removed streams from the data set and performed inference using only the remaining streams. Results reported for fewer than five microphones are averaged over all permutations of that number of microphones with standard errors also reported.

2.2.1 Finding Voiced Speech

Our method relies on first inferring whether a recorded stream contains voiced speech. We use a hidden Markov model (HMM) with one time step per 60 Hz frame of audio features. The HMM's observation variable is a 3-dimensional vector containing the 3 features previously described as useful for voicing detection: the value of the non-initial autocorrelation peak, the number of autocorrelation peaks, and the relative spectral entropy. Let $\mathbf{x}_a$ denote the vector of observations for person a with $\mathbf{x}_a^t$ being the three observed variables at time t. Similarly, let $\mathcal{V}_a$ be the vector of hidden states for person a.

The observation probability $p(\mathbf{x}_a^t|\mathcal{V}_a^t)$ is modeled with a full covariance 3 dimensional Gaussian, and the state transition probabilities are modeled with the usual transition matrix. The parameters of the voicing HMM are learned from data that does not contain any of the speakers in our evaluation data (or in our larger corpus). This voicing HMM has been shown to be speaker-independent and robust across different environmental conditions (Basu, 2003).

For each recorded stream, we use the forward-backward algorithm (Rabiner, 1989) to infer $p(\mathcal{V}_a^t|\mathbf{x}_a)$: the posterior probability of voiced speech in each frame, given the entire recorded stream. Figure 2.4 shows the spectrogram and autocorrelogram from Figure 2.2 with the inferred voicing posterior for the example recording overlaid.

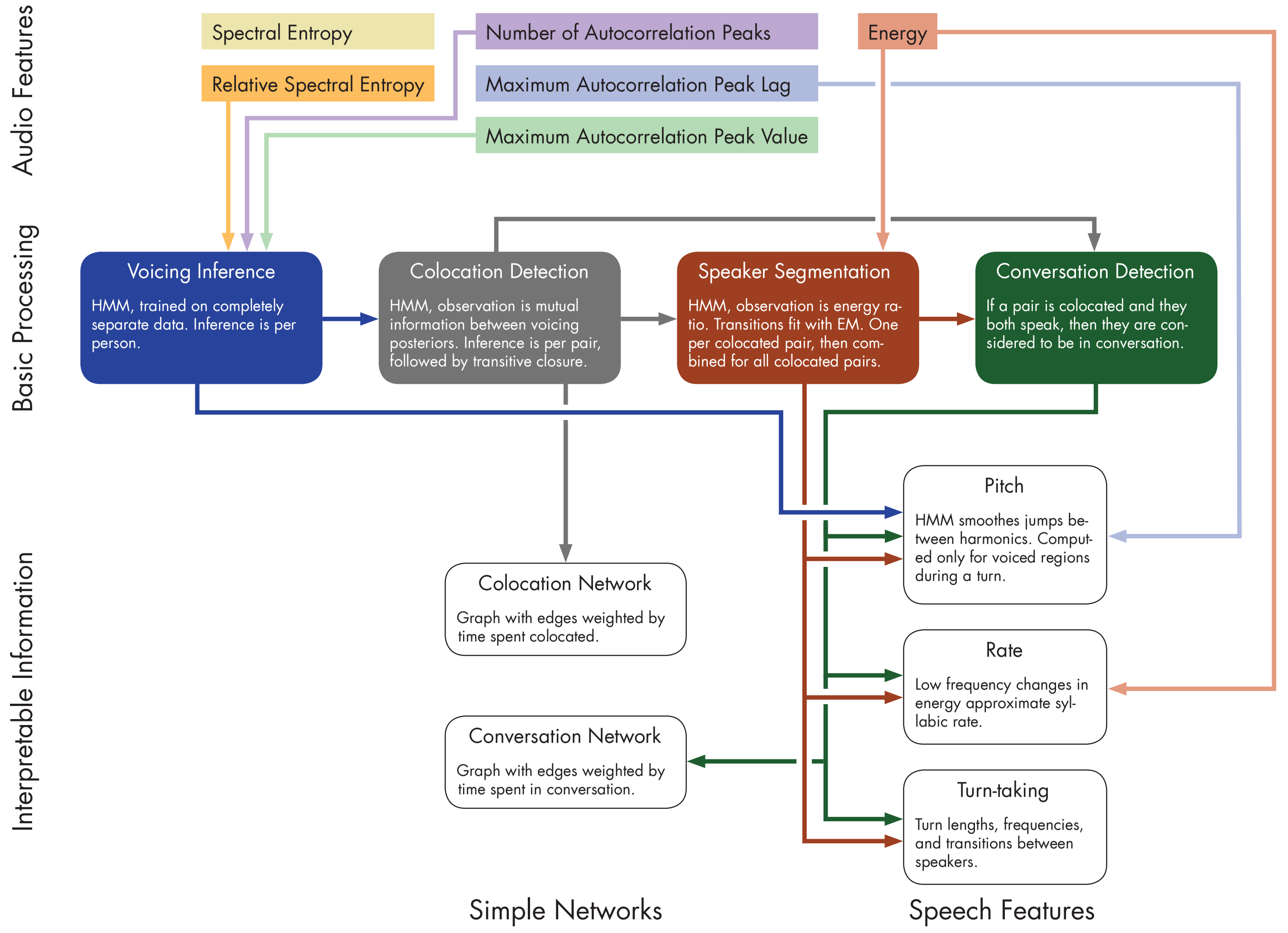
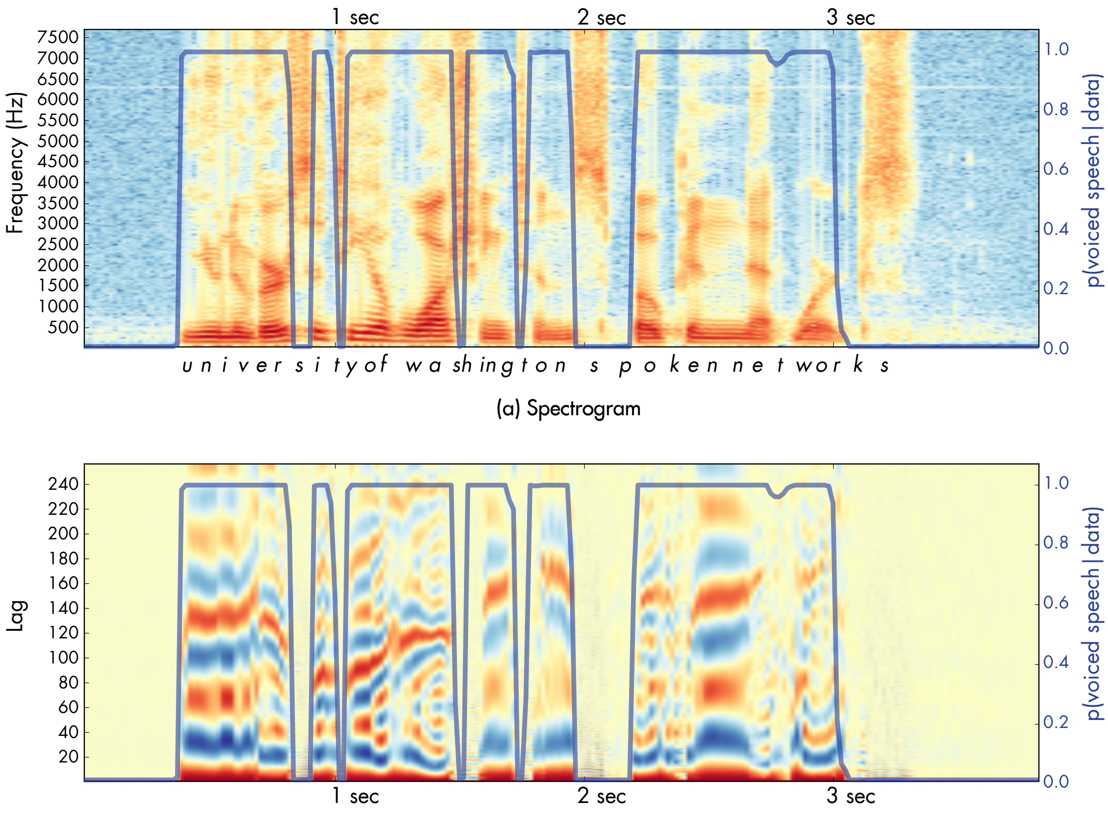

Audio Features
Basic Processing
Interpretable Information
Spectral Entropy
Relative Spectral Entropy
Number of Autocorrelation Peaks
Maximum Autocorrelation Peak Lag
Maximum Autocorrelation Peak Value
Energy
Voicing Inference
HMM, trained on completely separate data. Inference is per person.
Colocation Detection
HMM, observation is mutual information between voicing posteriors. Inference is per pair, followed by transitive closure.
Speaker Segmentation
HMM, observation is energy ratio. Transitions fit with EM. One per colocated pair, then combined for all colocated pairs.
Conversation Detection
If a pair is colocated and they both speak, then they are considered to be in conversation.
Colocation Network
Graph with edges weighted by time spent colocated.
Conversation Network
Graph with edges weighted by time spent in conversation.
Pitch
HMM smoothes jumps between harmonics. Computed only for voiced regions during a turn.
Rate
Low frequency changes in energy approximate syllabic rate.
Turn-taking
Turn lengths, frequencies, and transitions between speakers.
Simple Networks
Speech Features

Figure 2.3: A schematic illustrating the process required to go from recorded audio features to meaningful conversation data.



(a) Spectrogram

(b) Autocorrelogram

Figure 2.4: Inferred voicing posterior (blue line, right y axis) overlayed on examples from Figure 2.2

2.2.2 Finding Colocated People

We treat finding colocated groups within the multiple streams of data as a clustering problem. Successful conversation detection requires clustering portions of streams together if the wearers who recorded the streams were in a conversation during those portions. Once the voicing posteriors are computed, the voicing frames are aggregated into *colocation windows* of size $W = 1200$ voicing frames (20 seconds), with no overlap between windows. To determine whether two people are colocated, we examine the mutual information between simultaneous colocation windows from each of their streams. The mutual information between persons a and b during colocation window w is

$$I(\mathcal{V}_a^w, \mathcal{V}_b^w) = \sum_{(v,v') \in \{0,1\}^2} p(\mathcal{V}_a^w = v, \mathcal{V}_b^w = v') \log \frac{p(\mathcal{V}_a^w = v, \mathcal{V}_b^w = v')}{p(\mathcal{V}_a^w = v) p(\mathcal{V}_b^w = v')} \tag{2.1}$$

where $p(\mathcal{V}_a^w = 1)$ is the probability that any of the 1200 frames from person a is voiced, and $p(\mathcal{V}_a^w, \mathcal{V}_b^w)$ is the joint distribution over the 4 possible combinations of voiced states for a simultaneous frame for both a and b. Since the voicing values are not directly observed, we estimate these aggregate voicing probabilities as

$$p(\mathcal{V}_a^w = v, \mathcal{V}_b^w = v') = \frac{1}{W} \sum_{t=\tau}^{\tau+W} p(\mathcal{V}_a^t = v) p(\mathcal{V}_b^t = v') \tag{2.2}$$

$$p(\mathcal{V}_a^w = v) = \frac{1}{W} \sum_{t=\tau}^{\tau+W} p(\mathcal{V}_a^t = v) \tag{2.3}$$

where τ is the first time index in window w.

That is, we estimate the aggregate voicing distributions using their expected sufficient statistics according to the posterior distribution $p(\mathcal{V}_a^t | \mathbf{x}_a)$ computed by the voicing HMM. This allows uncertainty in the voicing inference to carry through to the conversation inference. The earlier method of (Basu, 2002) estimated the same probabilities using the *maximum a posteriori* (MAP) sufficient statistics (calculated from the Viterbi decode of the voicing HMM). We gain slightly in accuracy (Tables 2.3 and 2.4) by using this "soft" mutual information computed from expected sufficient statistics instead of a "hard" one computed from an MAP estimate.

While there are many methods for computing a similarity between two signals, mutual information between voicing inferences seems uniquely suited to finding conversations between people wearing microphones. At the expected physical distances for a face-to-face conversation, all microphones worn by participants in the conversation will pick up the speech of any speaker in the conversation. It is extremely unlikely that two microphones that are not close enough to be in a conversation will observe the same speech signal. Other metrics (e.g. correlation between energy, considered below) do not have this property.

The voicing mutual information of (2.1) is computed for all windows and all pairs. The empirical distribution of the logs of the resulting values, shown for one week in Figure 2.5, makes the division between colocated and separate pairs clear. There is a sharp peak of high mutual information values corresponding to colocated

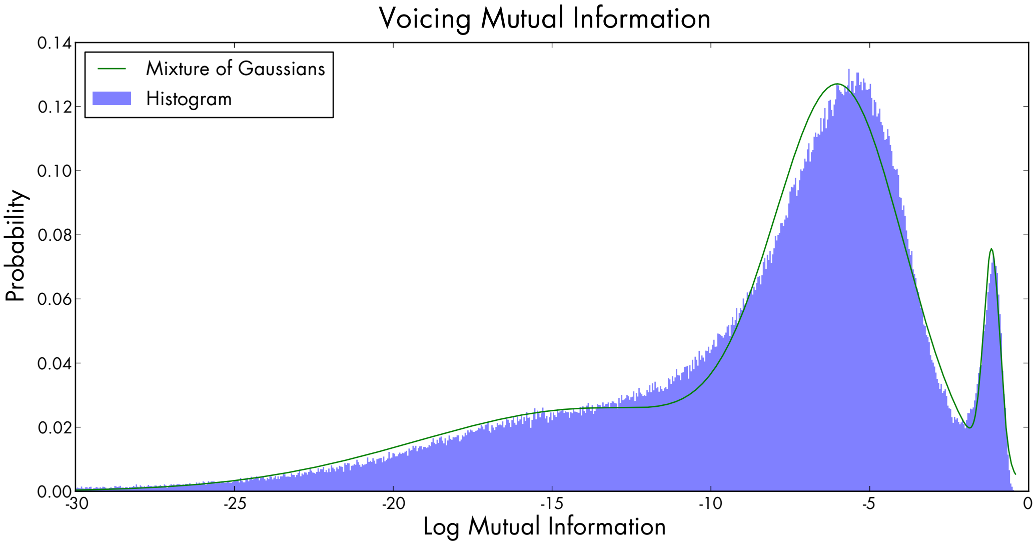


Figure 2.5: Histogram of voicing mutual information values for one week of data with fitted mixture model.

pairs, and two broader, overlapping peaks of lower values for separated pairs. That distinctness makes it easy to learn, in a completely unsupervised manner, different conditional distributions over log mutual information for colocated and non-colocated pairs. For that, a mixture of 3 Gaussians is first fit to all of the observed values (also shown in in Figure 2.5). The component with the highest mean is taken to be the conditional distribution of the log mutual information for a colocated pair. A mixture containing the other two components (with their mixture probabilities renormalized) is taken to be the conditional distribution for a non-colocated pair.

Since the colocation windows do not overlap, temporal smoothness in the colocation inference is enforced by using another HMM to infer colocation for a pair. The hidden state of the colocation HMM is a binary variable indicating whether the pair is colocated, and its observation variable is the log of the mutual information between their voicing posteriors. The observation probabilities are set to be those of the mixtures of Gaussians and the transition probabilities are fixed so that the expected duration in either state is one minute. In an earlier technique (Wyatt, Choudhury and Bilmes, 2007), we did not use an HMM for colocation but instead averaged together mutual information values from neighboring time steps using a normalized triangular window. One minute was found to be the optimal window length, hence the expected duration for the HMM. The HMM-based method does not perform any differently on labeled data than the simple window-smoothed method, but on the Spoken Networks corpus it produces much more plausible colocation inferences.

To ultimately partition people into colocated groups, the MAP sequence of colocation states for each pair is computed using the Viterbi algorithm. The transitive closure of the separate pairwise inferences is then calculated within each colocation window to ensure a consistent grouping.

Evaluation

As presented so far, there is not a single, well defined ground truth for the concept of colocation. Are two people colocated if they are in the same room? What if the room is a large hall and they are on opposite sides? The evaluation data includes labels for location at the room level as well as who is in conversation with whom. Each of those could provide ground truth for the colocation inference. Table 2.1 shows our technique's performance when compared to "in the same room with" ground truth. Table 2.2 show performance when compared to "in conversation with" ground truth.

There are 5 performance metrics presented in the tables, all derived from counts of true and false positives and negatives. To compute these metrics the set of all possible groupings of 2 or more people is considered for each colocation window. If a grouping occurs in the labeling and in the inference, then it is a true positive. If the grouping occurs in the inference but not in the labeling, it is a false positive. A true negative is a grouping that is in neither the labeled data nor the inference and a false negative is a grouping that is in the labeled data but not in the inference. Additionally, we define the *contained false positives* to be the false positives that are nevertheless valid subgroups of a true grouping—that is, inferred groups that are missing one or more true group members but contain no extra, erroneous members. The derived metrics are then defined as

$$\text{accuracy} = (\text{tp} + \text{fp})/\text{tp} + \text{fp} + \text{tn} + \text{fn} \tag{2.4}$$

$$\text{precision} = \text{tp}/\text{tp} + \text{fp} \qquad \text{(positive predictive value)} \tag{2.5}$$

$$\text{recall} = \text{tp}/\text{tp} + \text{fn} \qquad \text{(sensitivity, true positive rate)} \tag{2.6}$$

$$\text{specificity} = \text{tn}/\text{fp} + \text{tn} \qquad \text{(1 - false positive rate)} \tag{2.7}$$

$$\text{partial precision} = (\text{tp} + \text{contained fp})/\text{tp} + \text{fp} \tag{2.8}$$

To test the performance of our method in the presence of unmiked speakers, we selectively removed streams from the data set and performed inference using only the remaining streams. For $k < 5$ microphones results are computed for all $\binom{5}{k}$ combinations of excluded microphones, and the means and standard errors across these "folds" are reported. The overall result at the bottom of each table is the mean over all folds for all numbers of excluded microphones.

The conversation comparison is slightly more favorable, suggesting that the definition of colocation implicit in our voicing-based method is "close enough to converse." That is exactly what is needed to automatically collect data about face-to-face conversations.

There are two periods when the inferences disagree with one labeling or the other. First, there is one period

Table 2.1: Colocation inference compared to true room-level colocation.

	Accuracy		Precision		Recall		Specificity		Partial Precision	
Mics	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE
5	0.980	-	0.809	-	0.864	-	0.987	-	1.000	-
4	0.959	0.002	0.812	0.019	0.833	0.007	0.975	0.002	1.000	0.000
3	0.920	0.005	0.849	0.009	0.804	0.011	0.955	0.003	0.999	0.001
2	0.868	0.021	0.994	0.004	0.770	0.037	0.997	0.002	0.994	0.004
Overall	0.910	0.011	0.896	0.016	0.799	0.016	0.976	0.004	0.997	0.002

Table 2.2: Colocation inference compared to true conversation grouping.

	Accuracy		Precision		Recall		Specificity		Partial Precision	
Mics	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE
5	0.987	-	0.928	-	0.876	-	0.995	-	0.953	-
4	0.977	0.001	0.933	0.003	0.879	0.009	0.991	0.001	0.952	0.003
3	0.960	0.003	0.933	0.005	0.893	0.006	0.980	0.001	0.945	0.006
2	0.943	0.007	0.928	0.022	0.938	0.011	0.947	0.012	0.928	0.022
Overall	0.958	0.004	0.931	0.009	0.907	0.007	0.970	0.006	0.940	0.009

Table 2.3: Other colocation techniques compared to true room-level colocation.

	Accuracy		Precision		Recall		Specificity		Partial Precision	
Method	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE
HMM										
Soft MI	0.910	0.011	0.896	0.016	0.799	0.016	0.976	0.004	0.997	0.002
Hard MI	0.900	0.012	0.893	0.018	0.769	0.018	0.977	0.004	0.999	0.001
Energy	0.944	0.008	0.937	0.010	0.872	0.015	0.981	0.004	0.991	0.002
Threshold										
Soft MI	0.909	0.012	0.898	0.015	0.795	0.016	0.977	0.004	0.991	0.003
Hard MI	0.896	0.012	0.888	0.017	0.758	0.017	0.975	0.004	0.993	0.002
Energy	0.940	0.007	0.929	0.010	0.863	0.014	0.978	0.004	0.986	0.003

where the 5 people are in two groups (of 3 and 2) sitting at adjacent tables in the large public space. Their room-level location label is the same ("the large public space"), but the colocation inference separates them according to table. It could be argued that the labeling is too coarse in that situation. Conversely, there is another period where the five are again in two groups but at opposite ends of a conference table in a quiet meeting room. The colocation inferences places them all in one group—matching the room-level labeling but not the conversation labeling.

Comparing to other methods The two existing methods for acoustic colocation detection (Corman and Scott, 1994; Basu, 2002) differ from ours in two ways: (i) the choice of a similarity metric, and (ii) the method of using that metric to classify pairs as either colocated or separated. Neither previous approach proposes using any method to temporally smooth the colocation classification (as the HMM does for our method). Instead, both suggest classifying windows independently of all others using a threshold learned in a supervised way from labeled data. Unfortunately, neither proposes a specific learning algorithm or loss function. As such, it is difficult to make a direct comparison between our method and the others. We can, however, use their different similarity metrics with both the simple threshold learned through our mixture of Gaussians approach as well as with our HMM.

As mentioned above, Basu's similarity metric is the "hard" mutual information between voicing inferences computed from an MAP inference of voiced states. Corman and Scott's similarity metric is cross-correlation between raw audio signals. We can approximate that in a privacy-sensitive way by using the energy computed for each frame of features in place of the raw audio signal.

Table 2.4: Other colocation techniques compared to true conversation grouping.

	Accuracy		Precision		Recall		Specificity		Partial Precision	
Method	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE
HMM										
Soft MI	0.958	0.004	0.931	0.009	0.907	0.007	0.970	0.006	0.940	0.009
Hard MI	0.953	0.005	0.936	0.007	0.881	0.012	0.975	0.005	0.949	0.008
Energy	0.932	0.008	0.861	0.013	0.877	0.016	0.929	0.019	0.866	0.013
Threshold										
Soft MI	0.951	0.004	0.920	0.009	0.890	0.009	0.967	0.006	0.929	0.009
Hard MI	0.943	0.005	0.920	0.008	0.859	0.011	0.969	0.005	0.937	0.008
Energy	0.929	0.008	0.855	0.013	0.870	0.016	0.926	0.019	0.860	0.013

Table 2.3 shows the results for these alternate similarity metrics when compared to “in the same room with” ground truth. The soft MI row in the HMM section repeats the overall line from Table 2.1. The other rows report the same overall evaluations for different similarity metrics and classification methods. The HMM generally performs slightly better than the simple threshold, and soft mutual information generally performs slightly better than hard, but neither of those improvements is significant. More interestingly, the energy cross-correlation metric outperforms both voicing mutual information metrics. However, Table 2.4 shows the results when compared to “in conversation with” ground truth. As before, the soft MI row in the HMM section repeats the overall line from Table 2.2 and the other rows also report overall evaluations. Our method (the HMM using soft mutual information) outperforms all others, significantly so for some metrics.

This suggests that voicing mutual information is a better metric for finding people who are actually in conversation, while energy cross-correlation is better for finding people who are simply physically colocated. A possible explanation for that is that when people are colocated but in separate conversations, they are not taking turns with one another and will talk at overlapping times. The lower level voicing inference will potentially only make inferences about the louder signal—that of the wearer—and the two signals will not be similar. When people are colocated and in conversation, they take turns, allowing each persons’ speech to be clearly recorded on each microphone and the voicing inferences to be similar. So it is possible that the voicing inference is filtering out some “noise” that corresponds to speech that is not part of the microphone wearer’s conversation.

2.2.3 Segmenting Speaker Turns

Once colocated groups have been found, we want to infer, in each grouping, who was speaking when. This is a task known as speaker diarization and there are a number of existing methods for it (Ajmera et al., 2004; Reynolds and Torres-Carrasquillo, 2005; Anguera, 2006). However, all of the existing methods use features (primarily mel-frequency cepstral coefficients) from which the verbal content of the signal can be easily be inferred, violating our privacy requirements. Our method relies on the output of our voicing classifier combined with the saved energy feature. Like our approach to colocation detection, our speaker segmentation method begins with separate inferences for each pair of people which are later combined into a global inference.

Pairwise Speaker Segmentation

First, for a given person a, the 60 Hz voicing frames are aggregated into longer *speaker frames*. We use a speaker frame size of 0.26 s (16 voicing frames) with an overlap of 0.13 s (8 voicing frames). The longer speaker frames reduce the sensitivity of the speaker segmentation algorithm to small errors in the voicing inference. The specific frame size was chosen because the NIST standard for evaluating speaker segmentation (NIST, 2009) allows for 0.25 s of forgiveness around speaker turn transitions, so we are operating at the maximum conventional granularity.

Two quantities are computed for each speaker frame s for person a: (i) g_a^s, the mean energy of its constituent voicing frames, and (ii) v_a^s, the log of the sum of the constituent voicing posteriors.

For these speaker frames, we instantiate a new HMM whose hidden state $\mathcal{S}$ has four values:

1. n: no one is speaking
2. a: person A is speaking
3. b: person B is speaking
4. u: someone other than A or B is speaking

The observations for this speaker HMM are the log ratios of the speaker frame energies: $r^s = \log g_a^s - \log g_b^s$. The speaker HMM observation probabilities, $p(r^s|\mathcal{S}^s)$, are modeled as a one-dimensional Gaussian distribution. The mean of the Gaussian for states n and u is set to 0. The mean for states a and b is learned from 3 minutes of data collected in a location and from a set of speakers that are different from those in our evaluation data. A single mean $\hat{g}$ is estimated for all pairs of speakers, and states a and b have their means set to $\hat{g}$ and $-\hat{g}$. The variances of the Gaussians for all four states (identical for a and b) are also estimated from this training data.

Generally, the log ratio r^s is greater than zero when $\mathcal{S} = a$ is speaking, less than zero when $\mathcal{S} = b$ is speaking, and $r_s \approx 0$ when $\mathcal{S} = n$ or $\mathcal{S} = u$. To disambiguate between states n and u, the probability of that any person is speaking during speaker frame s is computed as $p(w^s|v_a^s) = \left(1 + e^{\alpha - \beta k_a^s}\right)^{-1}$ where $k_a^s = \sum_{t \in s} v_a^t$ is the sum of voicing probabilities in speaker frame s for person a. In other words, $p(w^s|v_a^s)$ is computed with a logistic regression. The parameters α and β of that logistic regression are estimated from the same training data used to learn the HMM's observation probabilities.

The speech probability $p(w^s|v_a^s)$ is incorporated into the speaker segmentation HMM as soft, or virtual, evidence (Bilmes, 2004). Virtual evidence introduces a pseudo-observation vector $\mathcal{X}$ whose value is always observed to be 1, i.e. $\forall s\, \mathcal{X}^s = 1$. The observation probability for that pseudo-observation is then defined to be

$$p(\mathcal{X}^s = 1|\mathcal{S}^s = a) \triangleq p(w^s|v_a^s) \tag{2.9}$$

$$p(\mathcal{X}^s = 1|\mathcal{S}^s = b) \triangleq p(w^s|v_b^s) \tag{2.10}$$

$$p(\mathcal{X}^s = 1|\mathcal{S}^s = n) \triangleq \frac{1}{2}(p(w^s|v_a^s) + p(w^s|v_b^s)) \tag{2.11}$$

$$p(\mathcal{X}^s = 1|\mathcal{S}^s = u) \triangleq 1 - p(\mathcal{X}^s = 1|\mathcal{S}^s = n) \tag{2.12}$$

Note that the information about the voicing posterior is not incorporated through any variable's value, but instead through the inhomogeneous parameterization of $p(\mathcal{X}^s|\mathcal{S}^s)$, which varies with s.

For each conversation, the transition probabilities are set to intuitive initial values which are refined using expectation-maximization (EM). We tried using the entire dataset of all conversations to learn the transition probabilities, but that degraded performance. Additionally, learning the observation probabilities, $p(r^s|\mathcal{S}^s)$, using EM also reduced overall accuracy. This suggests that speaker transitions vary for different pairs of people in different conversations, and that energy ratios are difficult to separate in an unsupervised manner. Once the EM procedure converges, we infer the posterior distribution for each speaker frame using the forward-backward algorithm.

Combining Pairwise Segmentations Once posterior distributions over speaker states have been inferred for all pairs, those posteriors are combined into a single, global distribution for the entire group of colocated people. This is done by expanding each pairwise distribution into a larger distribution that has more than four states. Specifically, the expanded distribution has one state for each speaker who has been grouped together with the pair in the colocation step; one state for no speaker; and one state for any other unmiked speakers. If there are m speakers in a conversation the probability that was assigned to state u (for a given pair a and b) is divided evenly among the remaining $m - 2$ speakers' states and the unmiked speaker state. The probability values for the other states, a, b, and n, remain unchanged.

The expanded distributions from each pair are then combined to form the global distribution. We evaluated two simple methods of combining the distributions: summing $p(\mathcal{S}^s = y) = \frac{1}{Z}\sum_{a,b} p_{ab}(\mathcal{S}^s = y)$ and multiplying $p(\mathcal{S}^s = y) = \frac{1}{Z}\prod_{a,b} p_{ab}(\mathcal{S}^s = y)$, where $p_{ab}(\mathcal{S}^s = y)$ is the posterior probability computed by pair (a, b) and Z is a re-normalizing term. The summing approach achieved better empirical results and was the method used to construct the final global distribution.

From this global speaker distribution it is then easy to construct an MAP speaker segmentation vector $\mathbf{s}$ with $s_i = \text{argmax}_y\, p(\mathcal{S}^i = y)$. Note that for a conversation with m participants the values of $\mathbf{s}$ will range from 1 to $m + 2$, where the two "extra" values denote silence (no one speaking) and some unmiked other speaking.

Evaluation

To evaluate speaker segmentation, for each speaker frame we choose the most likely state from the combined speaker distributions and compare it to the ground truth in our evaluation data set. We perform this evaluation on two versions of our evaluation data: a raw version, and smoothed version. The raw evaluation considers all frames in the data. The smoothed evaluation, in accordance with the NIST standard mentioned above (NIST, 2009), merges any pause shorter than 0.3 s in a single speaker's turn and ignores 0.25 s of data around a change in speaker.

Since the segmentation problem has more than two states, simple metrics (like (2.4) to (2.7)) do not readily apply. However, a full confusion matrix for each conversation is also uninformative since it is not very interesting to see how often a specific person a is confused with any other specific person. We can examine pseudo-confusion matrices that show 3 ground truth states: no one, miked speaker, un-miked other; and 4 meaningfully collapsed inferred states: no one, the correct miked speaker, an incorrect miked speaker, and an unmiked other.

From these pseudo-confusion matrices three summary evaluation metrics are computed:

1. Accuracy: the fraction of frames in which the inferred state matches the ground truth state
2. Precision: the fraction of frames inferred to be spoken for which the correct speaker is inferred
3. Recall: the fraction of truly spoken frames for which the correct speaker is inferred

Table 2.5 shows the pseudo-confusion matrix for the raw evaluation, and Table 2.6 shows the corresponding summary metrics. Overall, the results are promising. The correct state is inferred most of the time. Importantly, miked speakers are rarely confused with one another. The most common mistake is when an unmiked other is incorrectly inferred to be one of the miked participants.

Table 2.7 shows the pseudo-confusion matrix for the smoothed evaluation, with the corresponding summary metrics in Table 2.8. Both accuracy and precision improve significantly when the ambiguous boundaries at the starts and ends of speaker turns are excluded. The confusion between un-miked others and miked speakers has also been reduced.

Table 2.5: Raw speaker pseudo-confusion matrix.

		Inferred Class							
		None		Miked, Correct		Miked, Incorrect		Un-miked Other	
		Mean	SE	Mean	SE	Mean	SE	Mean	SE
True Class	None	**0.067**	0.003	-	-	0.072	0.003	0.017	0.002
	Miked Speaker	0.013	0.001	**0.569**	0.017	0.011	0.001	0.028	0.002
	Un-miked Other	0.018	0.002	-	-	0.091	0.007	**0.115**	0.011

Table 2.6: Raw speaker segmentation performance.

	Accuracy		Precision		Recall	
Mics	Mean	SE	Mean	SE	Mean	SE
5	0.817	-	0.825	-	0.967	-
4	0.781	0.006	0.788	0.006	0.961	0.003
3	0.750	0.010	0.756	0.010	0.956	0.004
2	0.730	0.015	0.736	0.016	0.955	0.007
Overall	0.751	0.008	0.757	0.009	0.957	0.003

Table 2.7: Smoothed speaker pseudo-confusion matrix.

		Inferred Class							
		None		Miked, Correct		Miked, Incorrect		Un-miked Other	
		Mean	SE	Mean	SE	Mean	SE	Mean	SE
True Class	None	**0.047**	0.003	-	-	0.051	0.003	0.011	0.002
	Miked Speaker	0.006	0.001	**0.645**	0.020	0.007	0.001	0.018	0.002
	Un-miked Other	0.013	0.002	-	-	0.088	0.008	**0.114**	0.013

Table 2.8: Smoothed speaker segmentation performance.

	Accuracy		Precision		Recall	
Mics	Mean	SE	Mean	SE	Mean	SE
5	0.876	-	0.883	-	0.983	-
4	0.838	0.006	0.846	0.005	0.978	0.003
3	0.806	0.012	0.813	0.012	0.973	0.003
2	0.782	0.017	0.789	0.017	0.972	0.006
Overall	0.806	0.009	0.813	0.010	0.974	0.003

2.3 Conversation Data

The steps described so far provide ways of determining who is physically colocated with whom and who is speaking when, but they do not provide a method for determining who is in conversation with whom. Such a method is difficult to define because the ground truth for the relation "in a conversation with" is more ambiguous than physical location or speaking state.

For example, imagine two officemates a and b who work mostly silently for two hours while occasionally talking. a makes a comment, b responds, and a short exchange ensues before they fall back into silence. When does the conversation start and when does it end? If a makes another comment but b does not explicitly respond, is that a conversation? If a third person c enters the room and speaks to b but only a responds, who was in conversation with whom?

To define conversations for our subsequent analyses we make the following three assumptions: (i) to converse, two people must be physically colocated; (ii) all people considered to be in a conversation together must speak at least once; and (iii) "enough" intervening silence ends a person's participation in a conversation.

Making those assumptions concrete, we say that a person is *active* during a 20 s colocation frame if he speaks for at least half a second during that frame. We also say that he is active for 20 s before the first frame in which he first speaks (to account for people beginning to join an ongoing conversation) and that he is active for 40 s after the last frame in which he speaks (an ad hoc threshold for "enough silence"). If two people are colocated and active, then they are considered to be in conversation with each other.

Note that this is a pairwise relation. a and b can be in conversation for a long period of time if they continue speaking, but c may only occasionally be put into a conversation with them if she speaks infrequently. This may seem to exclude more silent people, but the short threshold required to be active should capture even the slightest back channel communication required for a conversation to proceed smoothly. Additionally, the previous enforcing of transitivity for the colocation relation will ensure that the conversation relation is properly transitive.

These heuristics happen to match our evaluation data perfectly, so an evaluation comparing the resulting inferred conversations to the "in conversation with" ground truth label yields exactly the same results as Table 2.2.

2.3.1 Low-Level Speech Features

As mentioned above, many useful inferences about non-linguistic aspects of a person's speech can be inferred from the privacy-sensitive set of features we use. Once we know when a person is speaking, we can compute several additional measures that capture *how* she is speaking. Later chapters will consider three specific measures: patterns of turn-taking, pitch (F0), and rate.

Turn-Taking Features

Obviously, once the speaker segmentation vector $\mathbf{s}$ for a conversation has been inferred we have information about how people were taking turns during the conversation. For a conversation with m participants, we can compute the complete $m+2 \times m+2$ conversation turn transition matrix $\mathbf{T}$ where, assuming $\mathbf{s}$ is of length T,

$$T_{ij} = \sum_{t=2}^{T} \mathbb{1}_{[s_{t-1}=i \wedge s_t=j]} \tag{2.13}$$

Two summary statistics, derived from $\mathbf{T}$, will be used: turn frequency and turn duration. Turn frequency is the number of turns taken by a person divided by the length of the conversation. The number of turns taken by person i can be computed from $\mathbf{T}$ as $n_i = (\sum_j T_{ij}) - T_{ii}$, making sure to increment n_i by 1 if $s_1 = i$. Turn duration is the mean length of a person's turns. That mean can be computed as $d_i = 1 + \frac{T_{ii}}{n_i}$ (as long as $n_i > 0$, which our minimum turn length heuristic ensures is the case).

Pitch

As explained above, all voiced speech has some fundamental frequency F0 and the lag of the non-initial maximum autocorrelation peak can be used to determining a speaker's F0 (Rabiner, 1977). The peak will not always correspond to the exact F0, however. Sometimes a harmonic of F0 will be momentarily stronger. In that situation, the lag of the maximum peak will jump by approximately an integer factor, corresponding to e.g. a doubling or tripling of the frequency.

To smooth out such jumps and estimate the true F0 we employ (yet) another HMM (developed by Alex Stupakov). The hidden state in the pitch HMM corresponds to the true F0 Hertz value, and the observation variables are the pitches derived from the autocorrelation lags. To retain the convenience of discrete states, all values are discretized to integers and bounded by intuitive values (75 Hz to 400 Hz for the hidden pitch, 0 Hz to 2000 Hz for the observable pitch). The transition probability is defined to be (the multinomial discretization of) a Gaussian with mean equal to the previous pitch and variance heuristically set to 500. The observation probability is constructed as follows. First, a mass of 1 is placed at the frequency corresponding to the true hidden pitch. Then, for each harmonic, a mass decreasing as e^{-3f} is placed at the f-th harmonic. These point masses are smoothed by convolving them with a Gaussian window with standard deviation equal to one-eight of an octave and a width of one octave. Finally, the entire vector is normalized to ensure that it is a valid distribution.

To estimate a person's pitch, the segments of speech inferred to be both spoken by her and voiced are fed separately to the pitch HMM. The Viterbi decode of the HMM is then used as the estimate of her pitch for that segment.

Rate

To measure a person's rate of speech, we wish to compute the number of syllables spoken per second. Since we have discarded all information about the linguistic content of the speech, we must approximate that quantity using only our privacy-sensitive features.

The enrate estimator (Morgan et al., 1997) has been shown to reliably approximate syllabic rate using only information about energy. Our implementation of enrate works in the following steps. First, a low-pass filter with a 15 Hz cut-off is applied to the energy to (which has already been effectively low-pass filtered with a cut-off of 60 Hz). Second, the filtered energy is broken into one second long windows, with 3/4 seconds of overlap. These windows are multiplied with a Hamming window, and a discrete Fourier transform is taken and squared to yield a power spectrum. Finally, the spectral mean of the power spectrum is found.

Ultimately, enrate is computing the expected frequency below 15 Hz, which will ideally capture the frequency of "bursts" of energy corresponding to syllables.

Chapter 3

The Spoken Networks Corpus

Using the conversation detection methods from the previous chapter we collected a corpus of real-world face-to-face conversations among a population of 24 subjects. This chapter first contrasts our effort with earlier data collection projects (Section 3.1), it then explains the procedure used to gather the data (Section 3.2), provides summary statistics about the data itself (Section 3.3), and shows novel measures of social behavior that can be easily extracted form the data (Section 3.4).

3.1 Related Work

The data that we have collected is novel in its combination of two broad aspects, each of which has its own antecedents. First, it contains the situated speech data for an entire subject population. In that aspect, it is related to earlier efforts at both spontaneous speech data collection and real-world social interaction measurement. Second, it covers an entire year of social interactions. That aspect relates it to previous work on collecting temporal social network data.

3.1.1 Spontaneous Speech Data

Existing efforts at collecting real-world speech data have considered settings—meetings, phone conversations, interviews (Ang, 2002; McCowan et al., 2003; Dielmann and Renals, 2004; NIST, 2009; Stupakov et al., 2009)—where the content of the speech is unpredictable, but the decision to have a conversation is made in

Parts of this chapter were previously published in (Wyatt, Choudhury and Kautz, 2007) and (Wyatt, Choudhury, Bilmes and Kitts, 2008).

advance. In these scenarios the dialogue is spontaneous, but the existence of the conversation is not. As such, the data sets do not capture information about their subjects' social networks.

Beyond that, most of the existing research on speech and emotion has either used acted speech data (Douglas-Cowie et al., 2003)—which is known to poorly reflect natural emotion (Batliner et al., 2000)—or small data sets limited to a handful of observations of each subject that cannot be used to compare one person's speech across different situations or over time (e.g. Greasley et al., 1995; Douglas-Cowie et al., 2000; Ang, 2002). Most are also recorded in relatively unnatural settings (television shows, interviews) that are not representative of ordinary human communication. Situated speech data will provide better measurement of actual, lived emotion in speech. We have found only one other attempt at collecting data in settings as spontaneous as ours Campbell (2002), but it only recorded single participants in isolation (i.e. only one side of a conversation).

3.1.2 Social Behavior and Temporal Network Data

Several studies have used cell phone data to consider real-world social interactions. Onnela, Saramäki, Hyvönen, Szabó, de Menezes, Kaski, Barabási and Kertész (2007) construct an undirected network of reciprocated cell phone calls with ties weighted according to time spent in conversation. They find that stronger ties occur within tightly connected groups and weaker ties cross groups. Additionally, removing weak ties eventually results in a sudden breakdown in reachability (the size of the largest network component relative to the number of people), while removing strong ties only gradually diminishes the reachability (Onnela, Saramäki, Hyvönen, Szabó, Lazer, Kaski, Kertész and Barabási, 2007). Considering cell phone call networks temporally, Palla et al. (2006) find that the ratio of ties out of a community to ties within is weakly predictive of how long a person will remain in the community.

Another temporal study comes from Kossinets and Watts (2006), who consider email sent between all students, faculty, and staff at a university over one academic year. They compare the empirical probabilities of new tie formation to counts of shared partners, network distance, and number of shared courses. They find that a new tie is more likely to form between a pair with more shared partners or courses, and the relationship between tie formation and those quantities is more pronounced for pairs that do not share any classes.

Of course, new data collection methods are not limited to only virtual communication. Borovoy (2002) developed an wearable badge capable of detecting physically proximate people. The badge used infrared sensors and thus could only detect people facing each other with a clear line of sight. Connolly et al. (2008) use data collected from motion sensors (Wren et al., 2007) to infer social events like walking together, attending the same meeting, or coincidentally meeting in a break room.

Eagle and Pentland (2006) present a system for inferring physical proximity from the short-range Bluetooth radios in people's cell phones. They can also infer coarse absolute location using cell tower IDs. Using this system they collected data for 94 graduate students from two different departments at one university. Confirming the early results on informant accuracy, they found that people's recall of whom they were colocated with is generally poor but does reflect long-term patterns of proximity (Eagle et al., 2009).

But perhaps the most interesting real-world social behavior data collection (and the immediate ancestor of this work) is that made possible with the *sociometer*: a wearable platform combining infrared, motion, and—most importantly—audio sensors (Choudhury and Pentland, 2003). Choudhury (2004) recruited 23 members of the MIT Media Lab—including graduate students, faculty and staff—to wear the sociometer for two weeks. She was able to automatically extract conversations from the data with accuracies ranging from 64% to 88%. (That study saved raw audio, so the conversation detection could be compared to a subset of the data that was labeled.)

Such automatically collected conversation data has many advantages over other real-world data collection methods. It is not restricted to line of sight like infrared and it will not infer colocation through walls like Bluetooth. It captures actual interactions, not just physical proximity that may not correspond to any interaction. And it allows for a much finer-grained observation of the behavior during an interaction, not just the fact of whether or not an interaction occurred.

3.2 Data Collection Method

The data collection effort considered in this work descends from, and retains the rich information of, the original sociometer study. Where our effort differs is in the choice of subject population and the length of observation time.

The population that we recruited consists of 24 (of 27) incoming graduate students, all of whom were in the same department at a large research university. The students were almost all new to the university, city, and each other, providing an opportunity to observe the formation of their social network from very close to "time zero." Additionally, these students are all (ostensibly!) peers with no formal relationships defined between them. That is in contrast to Choudhury (2004)'s population which included professors as well as students of varying seniorities.

Our subjects recorded data by wearing a personal digital assistant (PDA) with an attached sensing device (described in more detail below). Subjects recorded data during whatever period each considered her "working hours." They recorded daily for one week each month over the 9 month course of an academic year. The first week had only 3 working days and the last only 4, for a total of 42 collection days. Aside from the days and hours, no other restrictions were placed on data collection. The subjects recorded data everywhere they went, inside and out: class, lunch, study groups, meetings, spontaneous social gatherings, etc. This 9 month period is much longer than the two weeks covered by the original sociometer study.

Data was saved to a 2 GB Secure Digital (SD) flash memory card on the PDA. Subjects were asked to upload their collected data at the end of each collection day, but because their memory cards could hold an entire week of data most waited until the end of the week. The subjects were paid for each day of data that they submitted. They were also allowed to use the PDA during non-collection weeks and were given the PDA at the end of the study.

3.2.1 Hardware and Software for Data Collection

All of the conversation data discussed in this work was collected using the same platform: an HP iPAQ hx4700 PDA with an attached multi-sensor board (MSB, Figure 3.1) containing 8 different sensors.

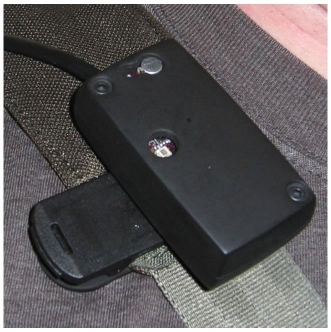
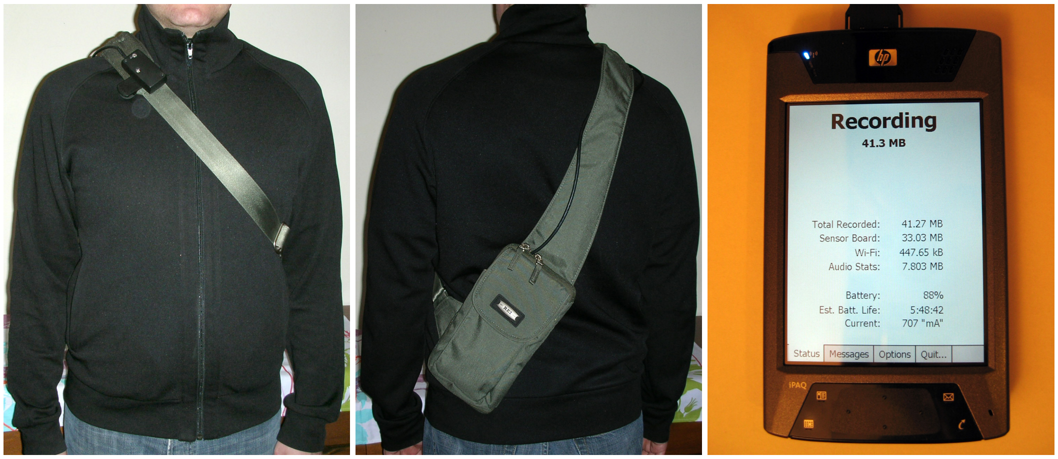


Figure 3.1: The MSB. Microphone is at top.

The PDA was carried in a small over-the-shoulder bag and the MSB was connected to the PDA via a USB cable that ran discreetly down the bag's strap (Figures 3.2a and 3.2b). The MSB was clipped to the bag's strap at the front of the wearer's shoulder, similar in placement to a lapel microphone. Recording could be started and stopped with the press of a single hardware button on the side of the PDA and the screen provided simple feedback to show whether the device was recording, how much data had been recorded, how much battery power remained, and an estimate of recording time left with the available battery power (Figure 3.2c). The PDA has an Intel XScale PXA270 624 MHz processor, with no floating-point unit, and 64 MB of RAM. As mentioned above, all data was saved to an SD card, with files rotated every half hour. The file rotation was implemented to prevent any accidental corruption from spoiling an entire data collection session, but in practice corrupted files were found to be very rare.

Of all the sensors on the MSB, the most important sensor for conversation detection is clearly the microphone. The MSB's microphone is an inexpensive electret condenser microphone that records 16 bit audio at a rate of 15,360 Hz.

For the 24 subject population, raw audio was never saved—not even temporarily—on the device. The privacy-sensitive features described in Section 2.1 were computed in real-time on the PDA and only those features were saved. For the 5 subject group that generated the evaluation data described in Section 2.2, raw audio was saved in addition to the privacy-sensitive features. That group contains no subjects from the larger study population and all members of the evaluation group consented to have raw audio recorded during their 50 minutes of observed interactions.



(a) Front: MSB is on right shoulder (b) Back: PDA is in bag. (c) PDA and data collection program.

Figure 3.2: The data collection kit worn by each subject.

Though not addressed in this work, the MSB also contains 7 other sensors that sample at varying rates: triaxial accelerometer (550 Hz), visible light (550 Hz), digital compass (30 Hz), temperature and barometric pressure (15 Hz), infrared light (5 Hz), and humidity (2 Hz). These sensors can be used to infer the wearer's physical activity (e.g. walking, sitting, standing, etc.) and whether she is indoors or outside (Lester et al., 2005). In addition to the data gathered via the MSB, the PDA records (at 0.5 Hz) the MAC addresses and signal strengths of the 32 strongest visible WiFi access points. It was hoped that the WiFi data could be used to determine the wearer's absolute physical location (Ferris et al., 2006), but repeated attempts to infer locations from the recorded data were unsuccessful. Unlike audio, the raw data from the additional sensors and the WiFi readings are saved in their entirety with no initial feature processing.

3.2.2 Survey Data

In addition to collecting sensor data, the subjects also answered a series of surveys.

An initial survey administered on the first day of data collection asked questions about the subject's previous interactions with anyone in the department as well as (a) which sub-areas of the discipline the subject was interested in pursuing, and (b) which faculty members the subject was interested in collaborating with.

At the end of every collection week (excluding the first) a survey was administered that asked 5 core questions.

1. Which other participants the subject interacted with over the previous month in 4 categories: homework

collaboration, research collaboration, social visits (outside of school), and phone calls

2. Which 5 non-participant students within the same department the subject interacted with and how
3. Which sub-areas of the discipline the subject was interested in pursuing
4. Which faculty members the subject was interested in collaborating with
5. Which faculty members the subject had collaborated with.

Once each term, the end-of-week survey also asked which classes the subject was taking, how she was funded, and whom she considered her advisor. The very first collection week did not have an end-of-week survey because the week was only 3 days long and the subjects had already answered the same questions on the first day of data collection.

Three years after the first week of data collection a follow-up survey was administered. That survey asked the same within-cohort interaction question as the end-of-week surveys (question 1 above) as well as 4 new collaboration questions:

1. What sub-areas have been the subject's course of study
2. Who the subject's advisor is
3. Who (in both their cohort and among the faculty) the subject has collaborated with on research, and whether that collaboration led to a publication
4. How many publications the subject has

The follow-up survey also asked basic demographic questions about the subject's age, gender, ethnicity, religion, and languages spoken. (Those questions were delayed until the follow-up survey because of initial IRB concerns that were subsequently addressed.)

3.2.3 Problems Encountered

We encountered four significant technical problems during data collection. First, batteries died faster than anticipated. We discovered that the PDA's operating system was attempting to connect to known WiFi networks in weak signal conditions that we had not previously tested. We alleviated this problem by reconfiguring the OS to never attempt to connect to any network while the data collection application was running. Second, all of the PDA's software and settings are stored in volatile RAM and are completely lost if the battery fully discharges. Subjects found it easy to recharge their PDAs at the end of each collection day, but would often forget to charge them between collection weeks. This led to many Monday mornings of lost recording time while PDAs were reconfigured. Third, the PDAs' clocks are shockingly unreliable. We found them to drift up to 5 minutes between collection weeks, thus needing frequent resynchronization with a time server—which required periodically re-enabling the WiFi connection. Finally, the fourth significant problem was that the cable that connected the MSB to the PDA's USB card was not durable enough for many weeks of continuous use. Over time, the cable would become loose, and the PDA would intermittently lose its connection to the MSB. The first and third problems eventually required a significant re-write of parts of the recording software while data collection was underway. This led to a larger than planned gap between the third and fourth recording weeks.

Each of these problems ultimately arises from our stretching the PDA well beyond its intended use. It was meant to be turned on only sporadically for short tasks, not to run continuously as its user goes about her day. The PDA was also intended to be attached to a computer regularly, providing it with the opportunity to charge its battery and synchronize its clock. While PDAs are handy portable platforms for short data collection efforts, they were not suitable to long term collection efforts such as ours. Fortunately for subsequent efforts by other researchers, newer platforms—particularly smart phones—are much more suited to running long-lived, independent data collection tasks.

3.3 Collected Data

Our subjects gathered a total of 4,401.51 hours—183.40 days—of data. The amount of data collected per participant varied greatly, from a maximum of 321.53 hours to a minimum of 88.41 hours, with a mean of 183.02 hours. On average, each subject recorded 4.27 hours per day, with a minimum of zero and a maximum of 10.71.

Figure 3.3 shows beanplots (Kampstra, 2008) of the average number of hours collected per day for each collection week. (Beanplots are an alternative to box plots that allow for comparison across weeks while also showing more information about the specific distribution of data within each week.) The first three weeks show an increase in the amount of data collected as the subjects became more comfortable with the device and its use, and battery life was improved. We believe that collected amounts decrease in weeks 4 through 6 as the participants become fatigued and the study becomes less novel. Before weeks 7 and 9 we sent additional messages of encouragement to the group, and those may be responsible for the subsequent upturns in collection amounts.

Since colocated people and their conversations can only be found when participants are simultaneously recording, the number of overlapping recordings is a measure of perhaps more importance than the raw amount of data collected. Figure 3.4 shows histograms of the number of people simultaneously recording any 20 second window in the data (a window is only in the data if at least one person recorded it). While there is no moment when all subjects are recording (the maximum number of simultaneous recordings is 21), there is enough overlap in the data for it to contain many interactions. The average number of simultaneous recordings per window is 8.10 for the entire corpus, and 88.53% of all recorded windows are covered by at least

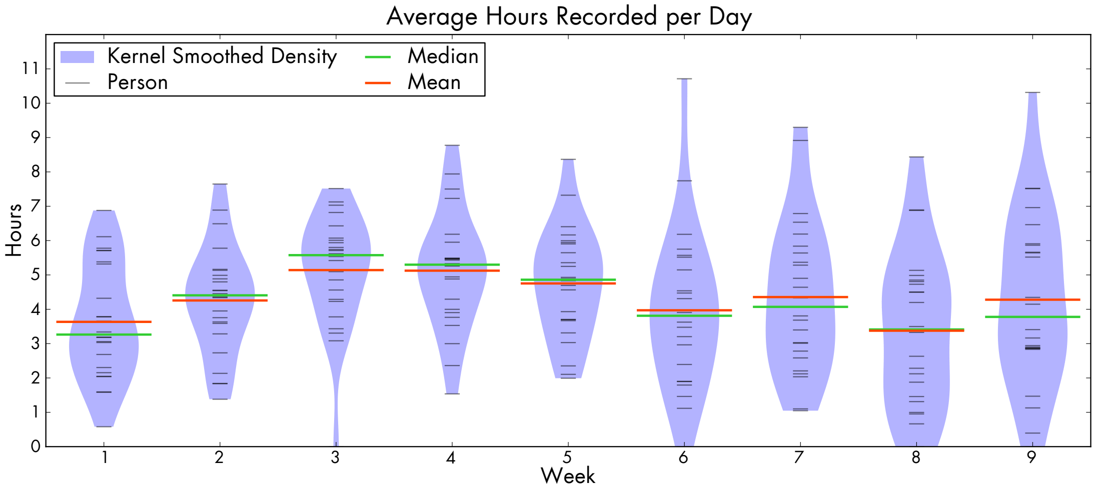


Figure 3.3: Average hours recorded per day for each subject in each week. Black lines are data points: the average for one person for that week. Blue "beans" are kernel density estimates. Green lines are medians and red lines are means.

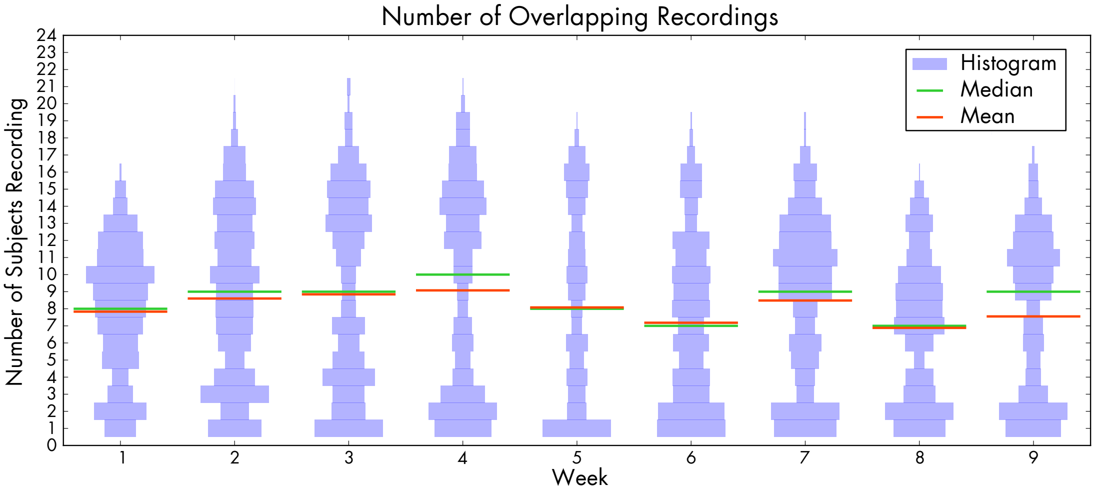


Figure 3.4: Number of people simultaneously recording each 20 second window with at least one person recording. Stacked blue boxes are histograms with one bin for each possible number of simultaneous recordings. The width of the box reflects the number of windows simultaneously recorded by the corresponding number of subjects. Green lines are medians and red lines are means.

two recordings. Additionally, there is overlapping data for all pairs of subjects. In fact, the pair with the least amount of overlapping recorded time still has 16.13 hours of simultaneous recording (the maximum is 215.18 hours).

3.4 Basic Behavioral Inferences

After the data has been collected it is processed through the three steps described in Chapter 2: colocation detection, speaker segmentation, and conversation extraction. Recall from Section 2.2.2 that colocation inference based on energy is more accurate when compared to physical location, but colocation inference based on voicing mutual information is more accurate compared to conversation grouping. Since each of these could have its benefits for sociological analysis, both methods were used to create separate colocation inferences for each week. The heuristics in Section 2.3 are used to group subjects into actual, interacting conversations, and pairs are only considered for conversation grouping if they are first determined to be physically colocated using the voicing-based colocation method.

3.4.1 Inspecting Daily Patterns

At the simplest level, the times of day that subjects turn on their recording devices provides information about their daily schedules. Figure 3.5 shows the number of subjects recording over the course of each day during week 4. Unsurprisingly, most subjects begin recording between 9 and 11 in the morning and gradually stop between 5 and 7 in the evening. The long slopes at both ends of the day show that different students keep different hours but most are around and recording during the middle of the day.

There is a noticeable increase in the number of people who begin recording around 10:30 am on Tuesday and Thursday. During this academic term, most subjects attended a class that met from 10:30 am to 12:00 pm on Tuesdays and Thursdays. The sharper increase in recording at that time is probably explained by subjects simultaneously arriving to attend class.

The colocation inferences in Figure 3.6 show the class much more clearly. Figure 3.6 shows the inferences for colocation using both energy (orange) and voicing mutual information (blue), as well as the conversation grouping (red). At each point in time, the number of pairs inferred to be together or in conversation is normalized by the number of pairs simultaneously recording at that moment. Thus each line is interpretable as the proportion of currently recording pairs grouped together according to each method. Because of that, when few people are recording (refer back to Figure 3.5) smaller groups will appear larger in the plot than perhaps they should. This is particularly true at the end of the day.

During the class on Tuesday and Thursday morning, the two colocation methods largely agree with one

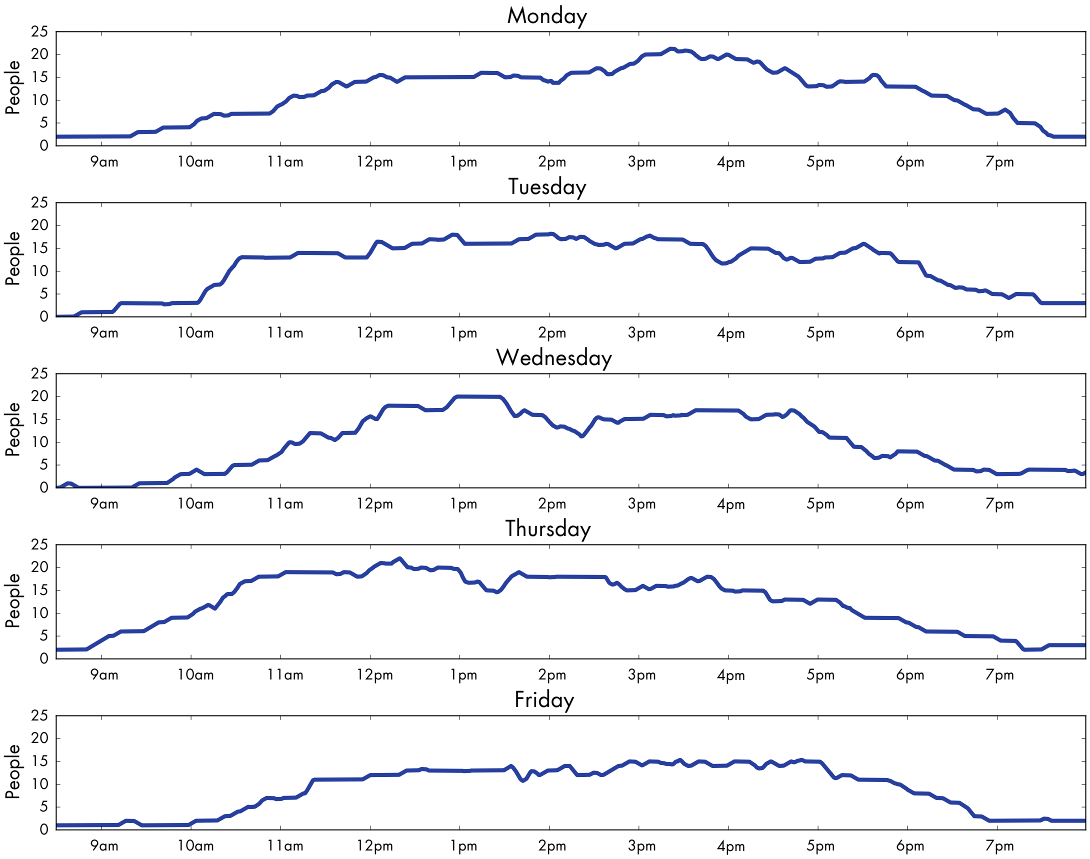


Figure 3.5: Number of people simultaneously recording over the course of each day during week 4.

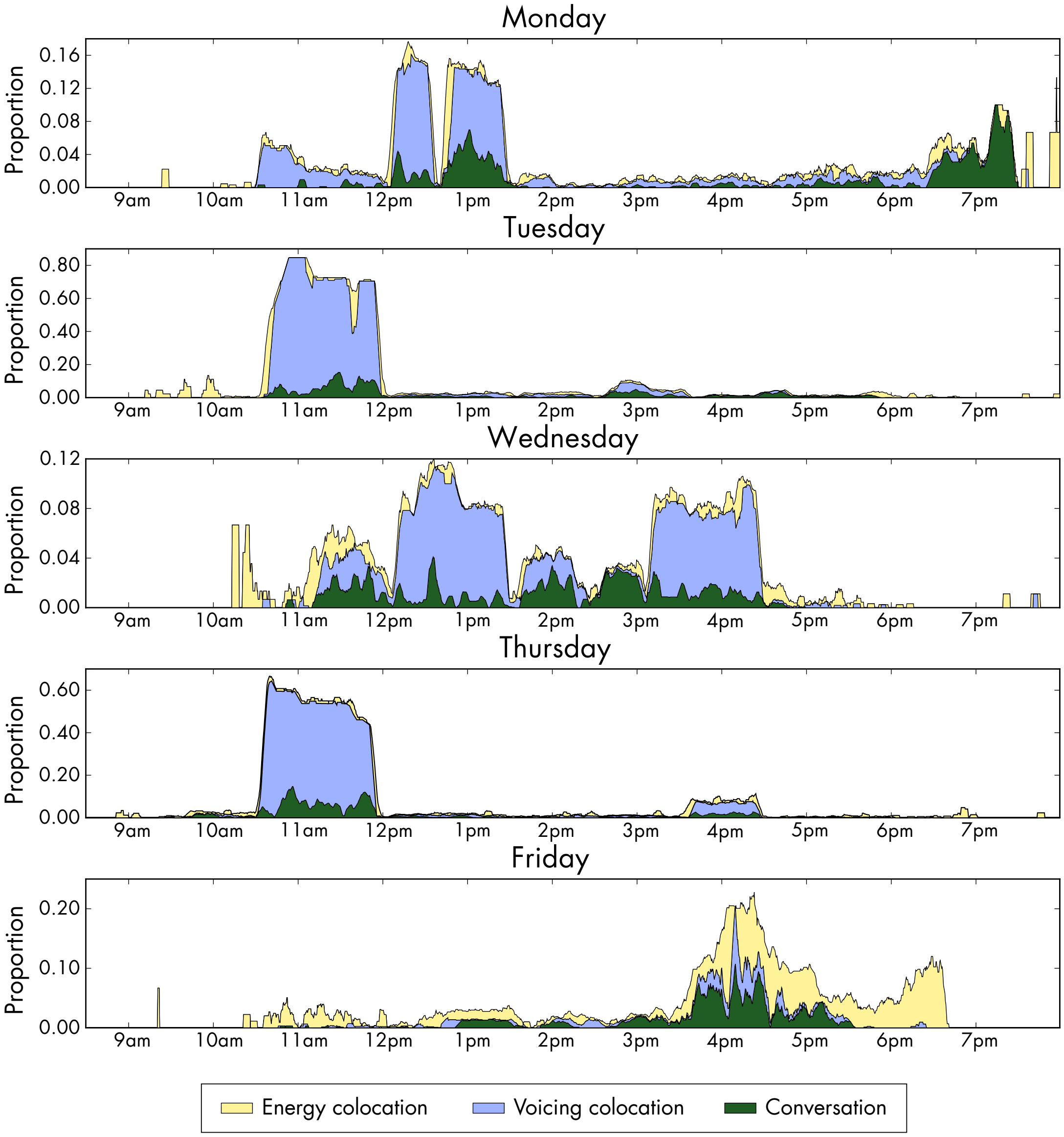


Figure 3.6: Proportion of recording pairs that are physically colocated according to energy correlation (yellow), voicing mutual information (blue), and in conversation (green).

another since the quiet of the class and the common signal of the instructor's voice will match in both energy and voicing inference. There are also classes on Monday and Wednesday from 12:00 pm until 1:30 pm, and on Thursday from 3:00 pm to 4:30 pm. All of these appear similarly in the colocation and conversation inferences.

In contrast to the classes, there is a department-wide social gathering on Friday afternoon. The energy-based colocation puts many pairs together (probably correctly), but the voicing-based colocation does not. That agrees with the earlier observation that during periods where the background noise is other conversations, the voicing colocation groups people into smaller, conversation sized groups while the energy colocation groups by broader physical location.

3.5 Basic Network Analyses

Constructing networks from survey data is usually simple: they are often just the union of self-reported ties for each actor in the network. Deriving networks from social behavior data is not so straightforward. Many short interaction events need to somehow be aggregated into a single network. This process of aggregation generally involves two broad steps: (i) aggregating observations across time into *temporal windows*–periods during which all observations are assumed to correspond to a single, static network, and (ii) deriving some measure of an edge from the data about the interactions within a window.

Aggregation across time is necessary because short, nearly instantaneous observations probably will not contain enough network structure to be interesting. For example, in the Spoken Networks data it is theoretically possible to observe networks at a granularity of 20 seconds, but it is unlikely such small snapshots would contain more than a handful of ties. Conversely, windows that are too long risk "blurring together" separate stages of the network's evolution and thus producing observed structures that do not correspond to any real network. A balance between the two must be found.

Once windows are defined, they may contain multiple or long-lived interaction events between pairs of people, e.g. many email messages or conversations. If simple networks are desired, some method is needed for deriving a single edge value from the rich data within a window. Since most network analysis techniques have been developed for binary networks, many studies of social behavior data have resorted to defining simple thresholds that separate binary ties from non-ties (e.g. Kossinets and Watts, 2006; Palla et al., 2006; Leskovec et al., 2008). An obvious alternative is to use weighted edges, but that requires using less common network analysis methodology.

For the simple analyses presented in this section we define our temporal window size to be one week. One week is both the longest contiguous window possible in our data (since there are weeks long gaps between recording weeks), and the shortest window that covers all days of the week. As Figure 3.6 shows, there are unique patterns of social behavior on different days of the week and a shorter window might mistake missing ties for the missing opportunity provided by some shared context (e.g. a once a week class).

For each week we construct two networks: the colocation network and the conversation network. In the colocation network edges between pairs indicate time spent in the same physical location. In the conversation network edges reflect time spent in conversation. To avoid selecting an arbitrary threshold, we consider weighted networks. However, since we can only observe an interaction between two people if both are simultaneously recording, it is sensible to normalize the observed interaction times by the amount of data available. Specifically, let o^t_{ij} be the amount of overlapping time in i's and j's recordings during week t. Let l^t_{ij} be the time the pair is inferred to be physically colocated (using the energy-correlation colocation detection, not the voicing-based method), and c^t_{ij} the time they are inferred to be in conversation. We define two networks: (i) the colocation network $\mathbf{L}^t$ where $L^t_{ij} = l^t_{ij}/o^t_{ij}$: the proportion of time that i and j spend colocated; and (ii) the conversation network $\mathbf{C}^t$ with $C^t_{ij} = c^t_{ij}/o^t_{ij}$, the proportion of time that i and j spend in conversation. Defining edge weights to be proportions has the added benefit of ensuring that they are all between zero and one. Many metrics developed for binary networks can then be applied without much modification, since a binary network is a special case of such a normalized weighted network where all ties (and non-ties) take on only the most extreme values.

Figure 3.7 shows the conversation networks constructed for each week. Obviously, a visual comparison of the networks can only provide so much insight. The rest of this section considers four simple network properties that can be more easily compared: network density, degree distributions, two measures of transitivity, and path lengths. We examine both how these properties change over time, and how they contrast between colocation and conversation networks.

3.5.1 Density

The density of a network is its mean edge value:

$$d(\mathbf{Y}) = \frac{1}{\binom{N}{2}} \sum_{i,j} Y_{ij} \tag{3.1}$$

For weighted networks, this has all the ambiguities inherent in summarizing a data set with its mean. For example, a weighted network with a few very strong edges may have the same density as one with many weak edges, despite the fact that they are very different networks from other perspectives. Density shows how much interaction exists in the network, but it does not reflect how that interaction is distributed. It is more illuminating, then, to consider the full distribution of edge values together with its mean.

Figure 3.8 shows those distributions as beanplots for the conversation and colocation networks across all weeks. The red line on each bean is the value of (3.1) for the week. Since most edges have very small values, it is helpful to show them on a logarithmic scale in order to see all of the variation in the data. Of course, zero

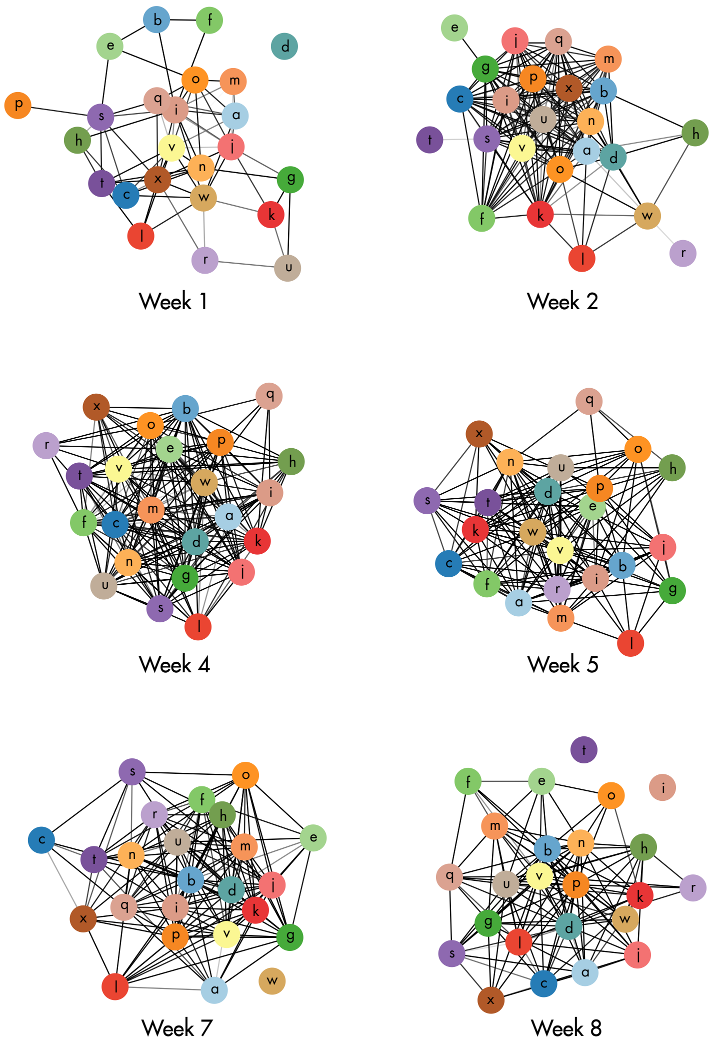
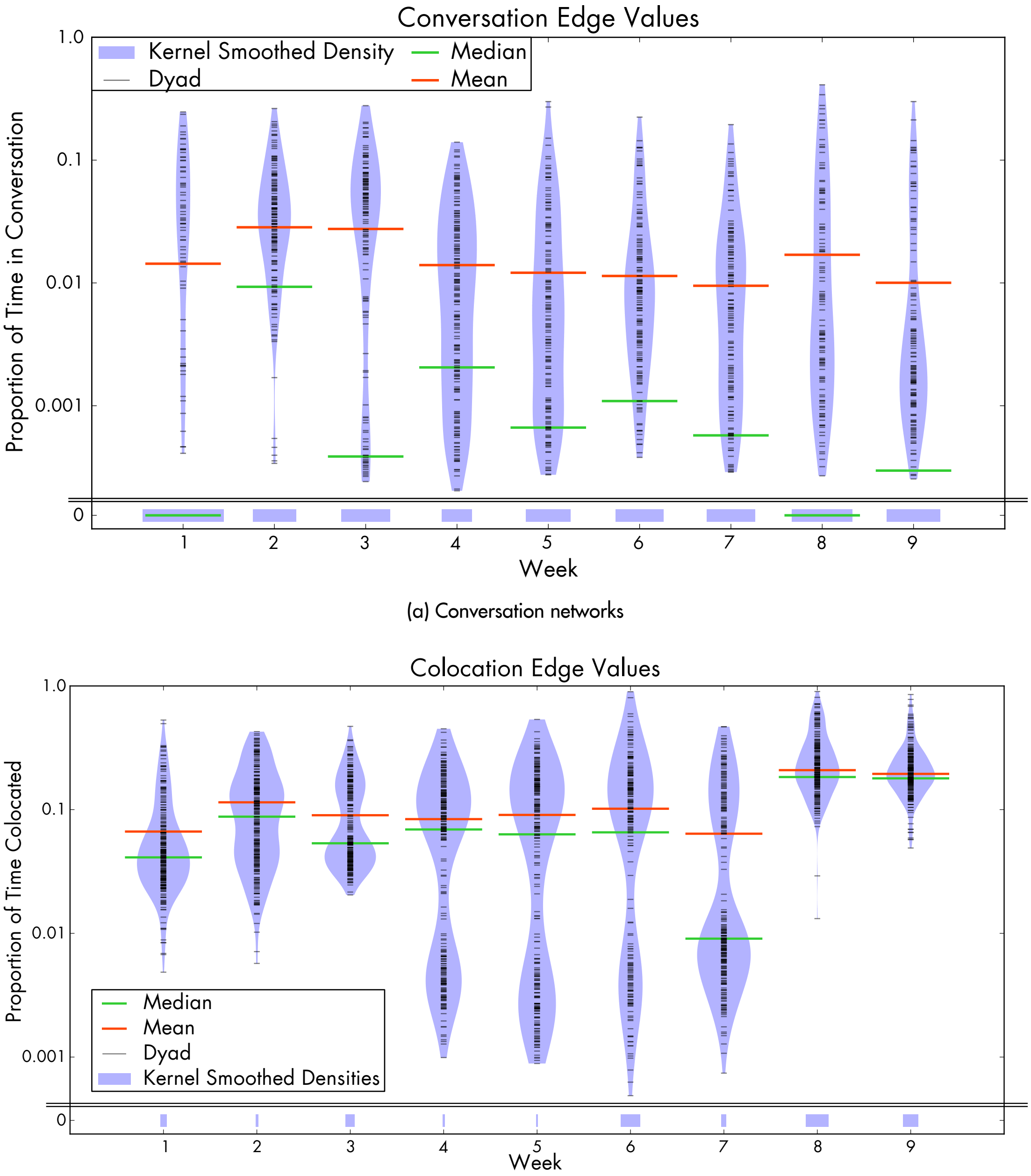

b f e d o m p q i a s h v j n t x c w g k l r u
Week 1
e j q m g p x b c i u n t s v a d h o f k w l r
Week 2
r g e j p n b m v f o h u t k i a x q d w s c l
Week 3
q x b o r p e h t v w i m a f c k d n j g u s l
Week 4
q x o n u h p d s t e k w v j b c f i g r a m l
Week 5
j d t o e h s x v f u p m n r a b i w c l k g q
Week 6
s o f h r c e u m t n b d j q i k x v g p l a w
Week 7
t i f e o m b n h q u v p r k g w d l j s a c x
Week 8
e m f n k d j v q u t o b h x i p a c l w r g s
Week 9

Figure 3.7: Conversation networks for each week. Edge shades correspond to proportion of time spent in conversation.



(a) Conversation networks

(b) Colocation networks

Figure 3.8: Edge value distributions. The data has been split into zero and non-zero valued edges. The width of the blue box at the bottom corresponds to the number of zero-valued edges for that week. The blue beans are kernel smoothed densities of log-transformed non-zero edge values. The width of the zero boxes and the beans can be compared: a wide zero box shows that there are many zero-valued edges and results in a thinner bean for the non-zero edges.

values cannot be shown on a log scale. Figure 3.8 thus shows a separate box or bin whose width corresponds to the number of zero-valued edges in the network. The blue beans are kernel-smoothed densities for the log transformed data. Thus the width of the bean at some point y on the y axis corresponds to $p(\mathcal{Y}_{ij} = y|\mathcal{Y}_{ij} > 0)$. The width of the box corresponds to $p(\mathcal{Y}_{ij} = 0)$. The width of a box and that of the corresponding bean can be compared: a wide box means there are many zero valued edges, and the bean will be thinner. Note that the means (red) and medians (green) are computed from *all* values, both zero and non-zero.

This distinction is necessary for the log scale display, but it also corresponds to a very natural intuition about weighted networks. There is a difference of kind, one beyond the simple difference in value, between zero valued edges and non-zero edges. Adding a new edge, even one with a minuscule value, can have drastic effects on the path lengths, reachability, and connectivity of the network. The box/bean split in Figure 3.8 can quickly provide a picture of the ratio of zero-valued edges to non-zero-valued edges. The median lines (green) also provide information about the ratio: for weeks 1 and 8 the median proportion of time spent in conversation is zero and thus more than half of all pairs are not connected by any edge in the conversation network.

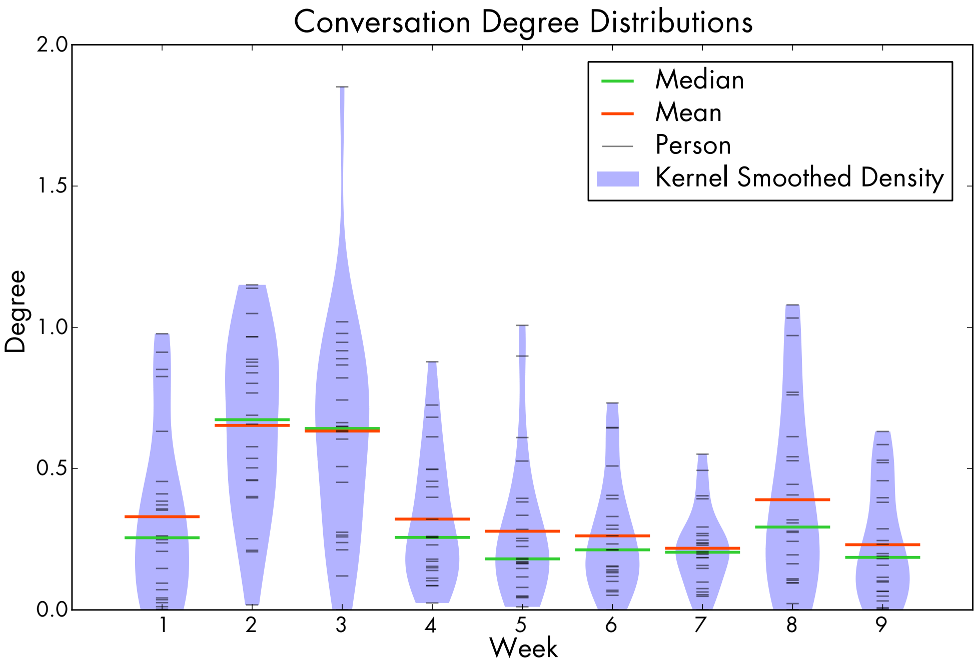
When comparing across weeks, the conversation edge values in Figure 3.8a show very different distributions. The early weeks seem almost bimodal, while the later weeks have elongated densities with gradual, almost linear decreases. Since the plot is on a log scale, this linearity corresponds to a roughly exponential decrease in probability for higher valued edges, a fact also reflected in the distance between the means and medians. There are certainly differences between weeks, but no pattern is immediately obvious.

A more useful comparison is that between the conversation and colocation networks. Figure 3.8b shows the same edge value distributions as in Figure 3.8a, only derived from the colocation networks. The differences between the networks is discussed further in Section 3.5.5 below.

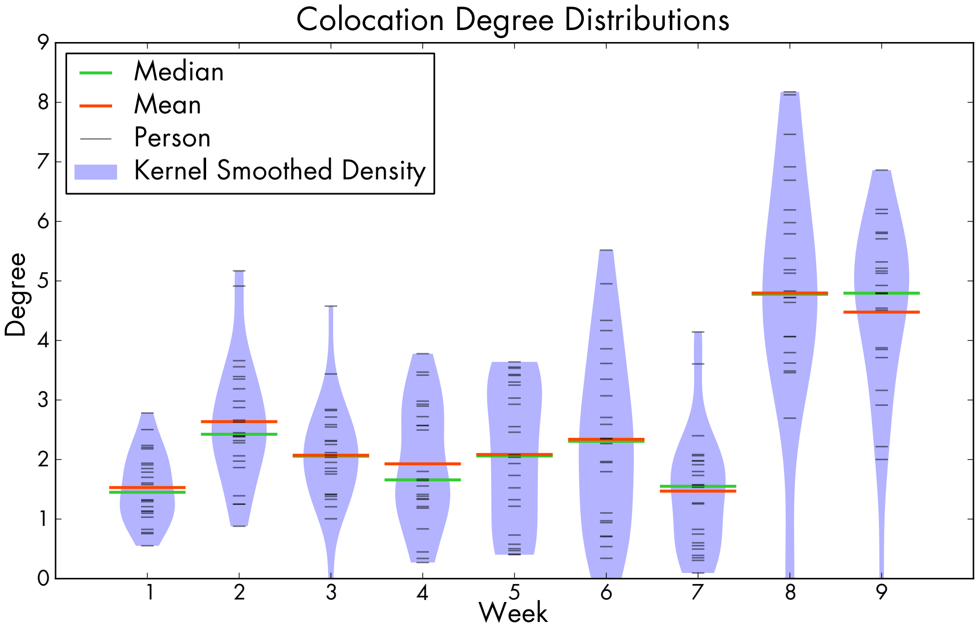
3.5.2 Degree

The degree of a node is the sum of the values of the edges incident to it: $d_i(\mathbf{Y}) = \sum_j Y_{ij}$. Different people may have different levels of interaction, and patterns in those differences can be seen in the network's degree distribution.

Figure 3.9 shows beanplots of the degrees of each person for the conversation and colocation networks for all weeks. As with the edge value distributions, the values for the colocation degrees are much higher than those for conversation degrees and the two kinds networks seem to be very different with regard to degree. This difference is also discussed below in Section 3.5.5.



(a) Conversation networks.



(b) Colocation networks.

Figure 3.9: Degree distributions.

3.5.3 Transitivity

An important property of social networks is their tendency to be transitive: people who are tied tend to both have ties to the same people. More colloquially, people tend to have mutual friends. Transitivity expresses itself through an increased number of triangles in the network, and thus metrics for quantifying transitivity are usually based around counts of triangles. In this section we will consider two such metrics: the clustering coefficient and the global triangle count.

The clustering coefficient for a person is defined as the fraction of pairs to whom she is tied who also have ties to each other (Watts and Strogatz, 1998). Equivalently, it is the number of triangles that involve her divided by the total number of possible triangles that could exist given her observed set of ties. Since the metric relies on the discrete existence or non-existence of ties, it does not generalize to weighted networks as easily as density and degree do. There are, however, several proposed variants of the clustering coefficient that can be used with weighted networks. The one we use is the weighted clustering coefficient defined by Saramäki et al. (2007):

$$C_i(\mathbf{Y}) = \frac{1}{k_i(k_i-1)} \sum_{j,k} (\hat{Y}_{ij}\hat{Y}_{ik}\hat{Y}_{jk})^{1/3} \tag{3.2}$$

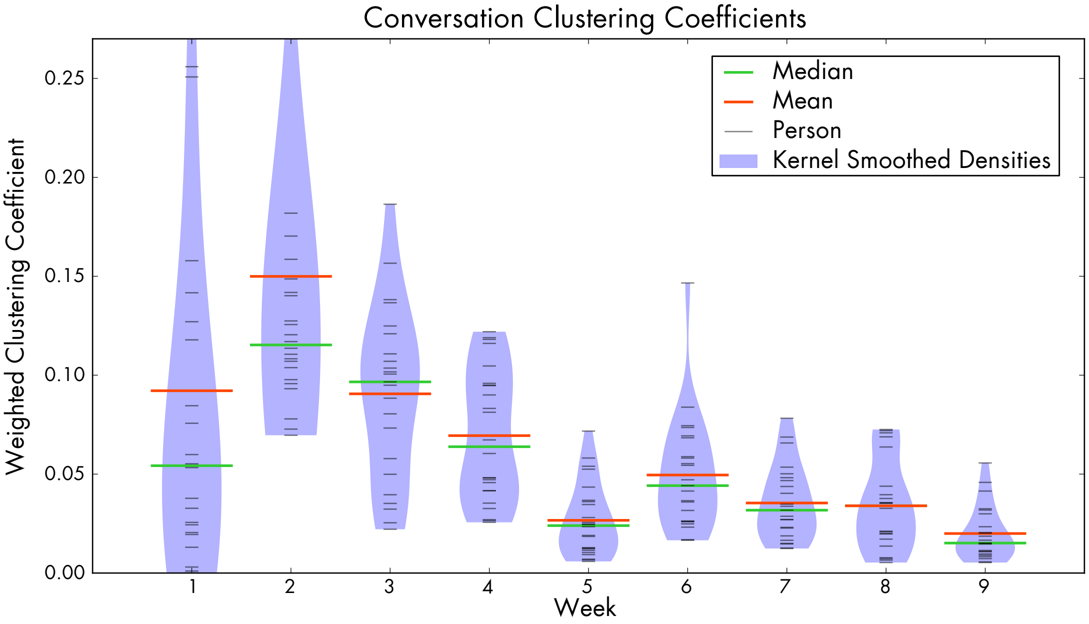
$\mathbf{Y}$ is a weighted adjacency matrix and $\hat{\mathbf{Y}} = \mathbf{Y}/\max(\mathbf{Y})$ is the normalized adjacency matrix where the maximum edge value is one. The weighted clustering coefficient defines the "intensity" of a triangle to be the geometric mean of the edges involved and thus is equivalent to the traditional clustering coefficient if edges take only zero or one values. $k_i = \sum_j \mathbb{1}_{[Y_{ij}>0]}$ is the "structural" degree of person i, and thus (3.2) captures the amount of triangle intensity that exists, divided by the total possible intensity (e.g. if i belonged to a clique where all edges have value one).

Another more global measure of transitivity is the simple count of all triangles in the network (Davis, 1970; Holland and Leinhardt, 1975). As with the clustering coefficient, the triangle count does not generalize as easily to weighted networks as degree and density Again this does not generalize to weighted networks as easily as degree or density, but, following Saramäki et al., we can define a weighted triangle value as

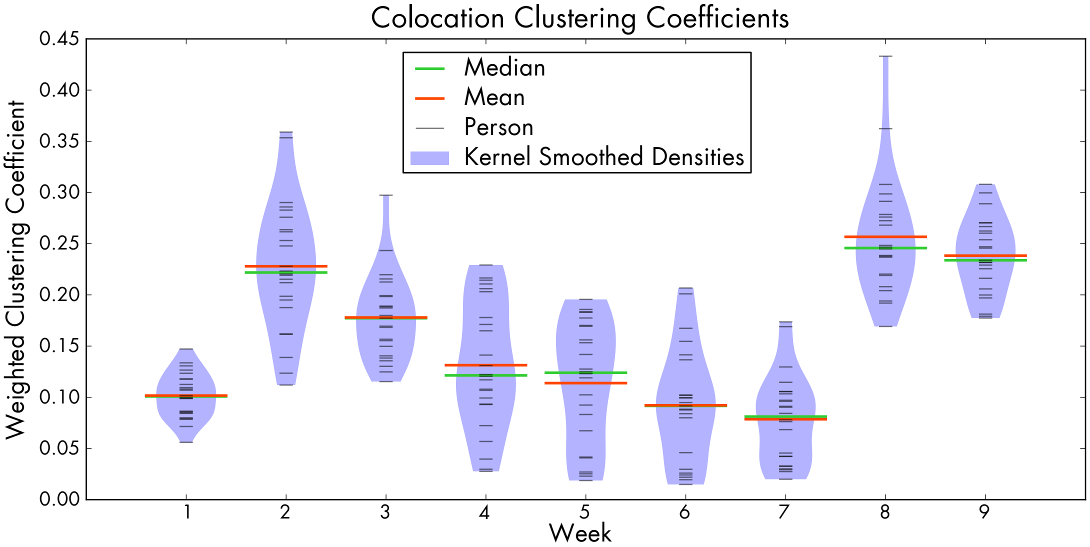
$$T_{ijk} = (Y_{ij}Y_{ik}Y_{jk})^{1/3} \tag{3.3}$$

As with (3.2), this value is equivalent to the ordinary triangle indicator if $\mathbf{Y}$ contains only binary values. As with edge values, looking at the distribution of the weighted triangle values will provide more information about transitivity in the network than the mean (or sum) alone would.

Figure 3.10 shows beanplots of the weighted clustering coefficients, and Figures 3.11 shows beanplots of the log scaled weighted triangle values. In Figure 3.11 the ratio of zero to non-zero values is shown as it was in Figure 3.8.

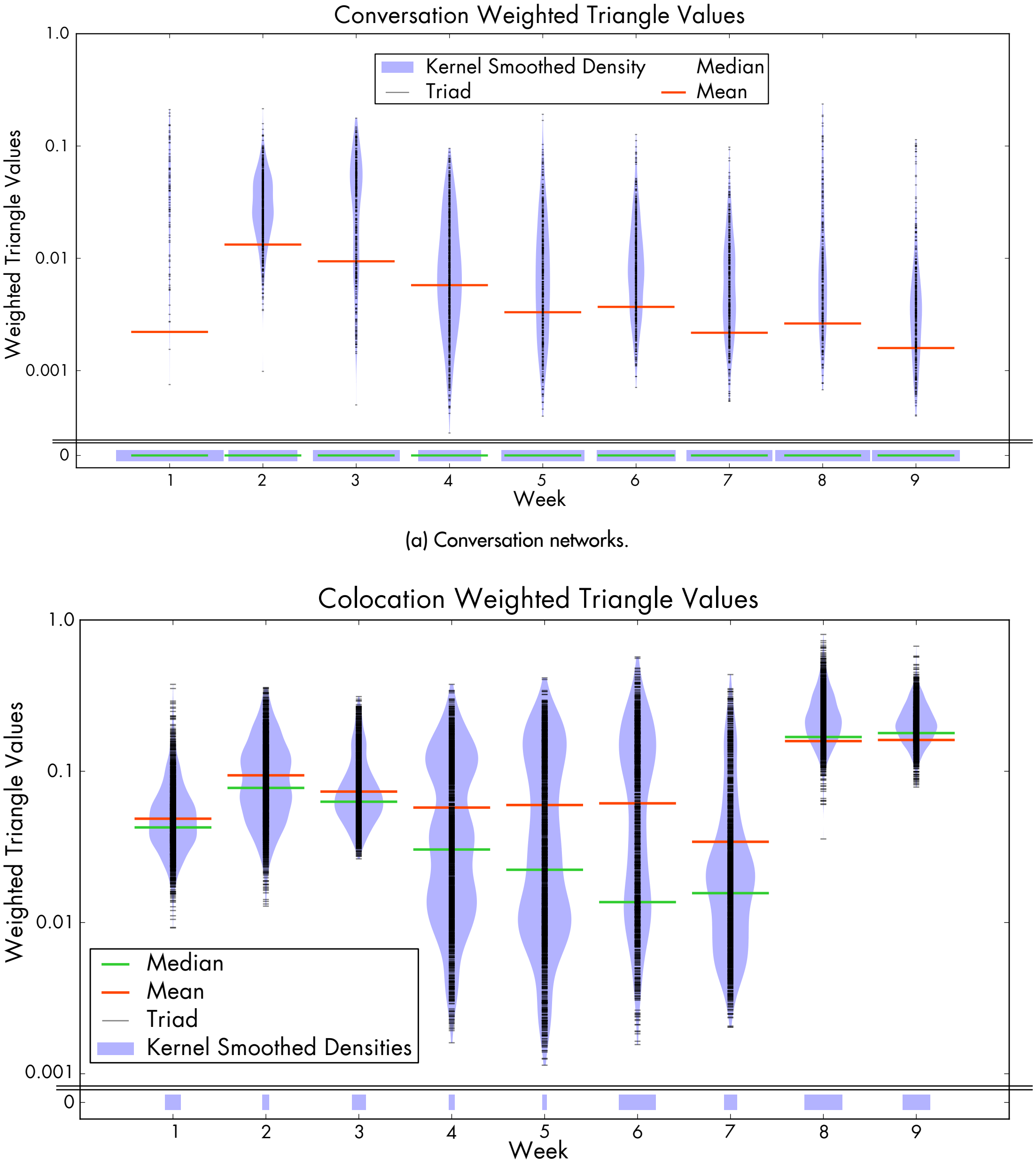


(a) Conversation networks.



(b) Colocation networks.

Figure 3.10: Weighted clustering coefficient distributions.



(a) Conversation networks.

(b) Colocation networks.

Figure 3.11: Weighted triangle value distributions.

For both metrics, there are extreme differences between the conversation and colocation networks. The clustering coefficient value are much higher in the colocation networks, and their changes over are completely different from those in the conversation networks. The median triangle count for the conversation networks is always zero: of $\binom{N}{3}$ potential triangles, the majority do not exist. For the colocation networks the median is never zero.

3.5.4 Path Lengths

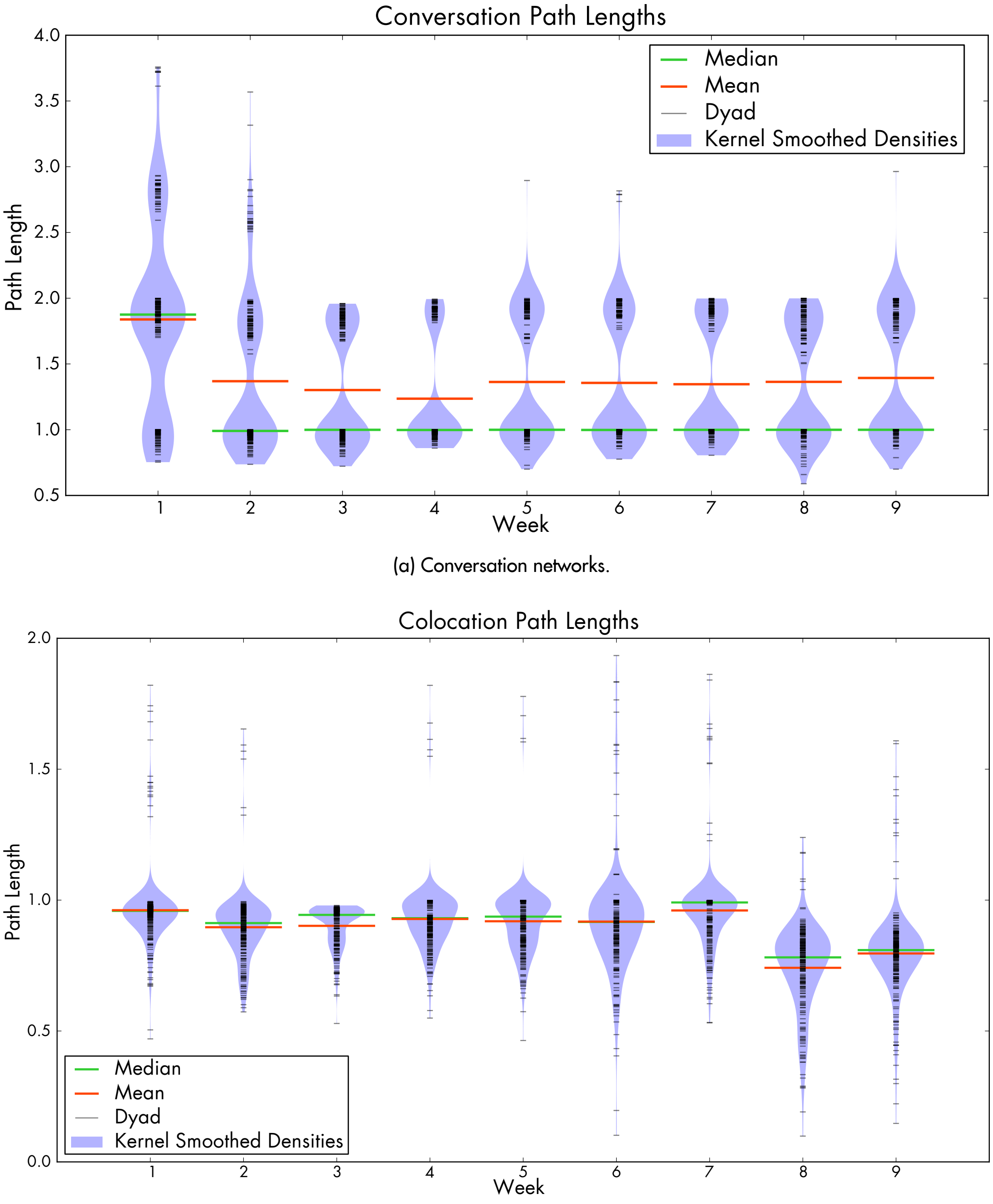
A final property of the networks to consider is the distribution of path lengths. To compute path lengths we define the length of edge (i, j) to be $1 - Y_{ij}$ if $Y_{ij} > 0$. In other words, the more time a pair spends interacting, the shorter the edge is. (If $Y_{ij} = 0$, then there is no edge between i and j and the length is undefined). The shortest path is found for all pairs and the distribution of path lengths are shown in Figure 3.12.

The conversation path lengths display a pronounced bimodality that corresponds to how many edges are involved in the path: values around 1 involve a single edge, values around 2 involve two edges, etc. This is unsurprising given the fact that most conversation edge values are small, as seen in Figure 3.8a, and thus most edge lengths are approximately one. The maximum point at each time step is the diameter of the network. We can see that paths are generally short, usually involving at most one intermediary. Indeed, in all but the first week, a majority of the shortest paths involve only a single edge. This is unsurprising given the strong connectivity of the network seen in Figure 3.7.

Paths in the colocation networks are also short, but much more so than paths in the conversation networks. However, the clustering of lengths around one does not reflect the same semi-discrete path lengths as in the conversation networks. There is much more variation in edge values in the colocation network (Figure 3.8b) so paths that traverse two edges can be as short as those that traverse only one. Nevertheless, regardless of how many edges are involved, most paths are still short.

3.5.5 Discussion

All of the metrics above reveal that the colocation and conversation networks are very different. The colocation networks are denser, and show correspondingly higher transitivity and shorter path lengths. Those differences are probably explainable through the simple phenomenon of shared classes. When many subjects attend the same class they are all colocated for a long period of time. This provides the opportunity for a single interaction event—the shared class—to create a large clique with heavily weighted edges in the network. Such large, strong cliques will naturally increase their members' degrees and clustering coefficients as well as the weighted triangle count of the entire network. Indeed, those three metrics are much higher for the colocation networks than the conversation networks.



(a) Conversation networks.

(b) Colocation networks.

Figure 3.12: Path length distributions.

Additionally, since all members of the cohort have offices in the same building, they have many opportunities to be physically proximate. That is reflected in the very different distributions of edge values for the conversation and colocation networks. The conversation networks have far more zero-valued edges, and lower non-zero values. The colocation networks have comparatively few zero-valued edges, suggesting that almost any subject is physically near most other subjects at least briefly during the week. Of course, subjects that share a class will have decidedly non-brief periods of time spent colocated. That difference may explain the bimodal colocation degree distributions of weeks 4 through 7 (Figure 3.8b), where there seems to be a distinction between pairs who spend much time together and pairs who only come together in passing.

When examining the changes in degree distributions as time progresses, the conversation distributions seem to become more stable, while the colocation distributions continue changing. That is perhaps because some durable social network begins to form. The influence of that durable network on time spent in conversation may gradually become greater than the influence of external factors, such as time spent together in class. Time in class would certainly have a larger effect on the colocation distribution. Unfortunately, the simple summary statistics presented here are not capable of distinguishing the relative importance of different factors on the network's evolution (a problem that will be addressed more in Chapter 4).

Whether the colocation networks or the conversation networks are to be preferred depends on the ultimate research question considered. If the spread of the flu, for example, is to be considered then the colocation networks may be more relevant. However, if the spread of information through face-to-face communication is to be considered then obviously the conversation networks are more relevant than the simple colocation networks.

Regardless of which is more important, the measures above reveal that the two networks are very different for this population, and that distinction that should inform future studies of real-world social networks, especially those that are based on only measurements of colocation and not actual interactions.

Chapter 4

Exponential Random Graph Models for Social Behavior Data

The descriptive analyses presented in the last chapter are able to show important differences between conversation and colocation networks, but they yield only the most basic conclusions about the how the networks change over time. There are hints of changes over time, but they are difficult to verify.

For example, both the clustering coefficients and triangle values (Figures 3.10a and 3.11a) seem to suggest that transitivity decreases over time, but there are also corresponding changes in density (Figure 3.8a) and degree (Figure 3.9a) that may explain any change in transitivity. Complicating that is the fact that many of the changes are complex. From week 7 to week 8 the mean edge value increases but the median drops to zero. How does that increase in variation between edge values interact with any potential change in transitivity?

Another example mentioned above is the fact that the colocation degree distributions move sharply toward larger values in the final weeks while the conversation degrees remain relatively stable or increase only slightly. Is there some durable, latent social network that is exerting more influence on time in conversation than the possibly exogenous factors influencing time in conversation? If so, what are the relative importances of those two potential influences?

To be sure, the problem of multiple overlapping influences, such as the examples in the previous paragraphs, is not unique to networks of social behavior data. For example, it has long been observed that social networks have much greater triangle counts than would be expected in a random network Davis (1970); Holland and Leinhardt (1975). A large number of triangles could occur for many reasons, three of which are enumerated

Parts of this chapter were previously published in (Wyatt, Choudhury and Bilmes, 2008) and (Wyatt et al., 2010).

by Goodreau et al. (2009). (i) People may have ties to others who are similar to them (homophily), groups of similar people will then form clusters with a coincidentally high number of triangles. (ii) There may be a few hub nodes of very high degree; any random tie between a hub's neighbors will complete a triangle. (iii) People may genuinely become friends with the friends of their friends ("true" structural transitivity).

To untangle these overlapping effects, we need a model capable of assigning probabilities to networks and quantifying how much changes in one statistic effect the probability of a network. With their interpretable parameters and ability to easily incorporate many statistics of interest, exponential family models of the form of (1.3) are excellent candidates.

The network models that have been developed in that form are known as *exponential random graph models*, or ERGMs. This chapter provides a short history of the development of ERGMs (Section 4.1) before discussing two extensions that we have made to ERGMs. These extensions exploit the richness of social behavior data and enable the new models to: (i) recover latent networks where hidden social relationships are observable only through noisy behavior data (Section 4.2), and (ii) discover long range, high level properties of evolving social networks using time-inhomogeneous models (Section 4.3).

4.1 A Brief History of Exponential Random Graph Models

Exponential family models, and graphical models of them, began to be applied to social network data in the 1980's. This section provides a history of the development of ERGMs, the problems discovered along the way, and the most recent advances in the field.

4.1.1 Early Exponential Random Graph Models

Holland and Leinhardt (1981) proposed perhaps the first exponential family model of a social network. In its most general form, the model defines a linear exponential family of distributions over networks that takes the form of Equation (1.3) with $\mathcal{Y}$ representing the adjacency matrix for a social network. The partition function of Equation (1.4) must then be taken over all $O(2^{N^2})$ possible networks—an intractability that early modelers often tamed only through resorting to unrealistic assumptions.

The p_1 model

Holland and Leinhardt overcome that intractability by restricting the set of features to those that allow the model to factorize in a way that makes (1.4) easy to compute. These features all rely on a *dyad independence* assumption: the two random variables $\mathcal{Y}_{ij}$ and $\mathcal{Y}_{ji}$ for the dyad (i, j) depend only on each other and are marginally independent of all other variables representing all other ties in the network. Holland and Leinhardt

(1981) refer to this class of models as p_1 to emphasize that it is only the first of many network models in the form of (1.3) that could be explored.

When using their model to simply predict back the training data (Sampson's network of 18 monks; Sampson, 1969; White et al., 1976), Holland and Leinhardt find that none of the observed ties have probability greater than 0.5; that 2/3 of them have probability less than 0.3; and that the most likely network, according to the learned model, is the empty network. A potential reason for this is that the network being modeled displays considerable clustering. The nodes can be separated into 3 clusters so that only 4 ties cross cluster boundaries. A better model would account for transitivity effects and clustering.

Markov Graphs

To extend the p_1 model to incorporate dependencies between dyads Frank and Strauss (1986) introduce a class of models for random graphs governed by a specific conditional independence assumption: two possible ties are considered conditionally independent, given the rest of the network, if they do not share a node.

Such models are called a *Markov graphs* since the class incorporates a Markov independence assumption for dyads, analogous to the assumption employed in Markov chains for temporal data or lattices for spatial data. This assumption captures the intuition that the forces driving a network's formation act only within a local context. Information about (i, j) provides direct information about all of i and j's other potential ties, but only indirect information about the rest of the network. For example, (i, j) provides information about (k, l) only through e.g. (j, k). Nevertheless, that locality is very expansive and the diameter of the graphical model is only 2 (as demonstrated in the previous sentence) so there can still be a strong dependence between all variables.

As it does for other classes of models (chains, lattices), the Markov assumption also provides a computational advantage. For a Markov graph, changing y_{ij} will only effect features involving the rows and columns for i and j in $\mathbf{Y}$. Thus the entire network does not need to be visited in order to update a feature vector. That is a great advantage for algorithms like Metropolis-Hastings (Section 1.4.5) that require repeatedly changing variables and recomputing feature values.

The graphical model for a Markov graph has one node per dyad in the network and an edge between nodes if the potential ties corresponding to those nodes are incident—share a person—in the social network.[1] Figure 4.1 shows the graphical model for a 5 node network and Figure 4.2 shows the corresponding factor graph. The factor graph makes the dependencies through people (nodes in the social network) much more clear.

[1] The "graph" in the term "Markov graph" does not refer to the dependency graph of the graphical model but to the random graph—the social network—that is being modeled. This is a source of confusion between the social network and machine learning literatures.

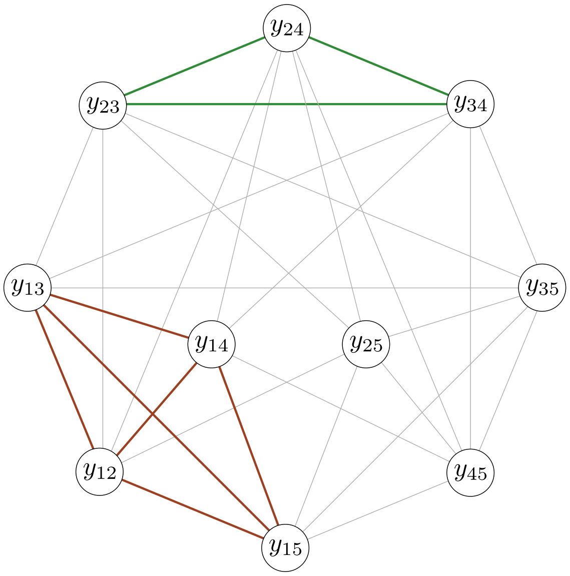


Figure 4.1: The undirected graphical model for a 5 node Markov graph. The green clique models the triangle $\{(2,3),(2,4),(3,4)\}$. The red clique models the stars around node 1.

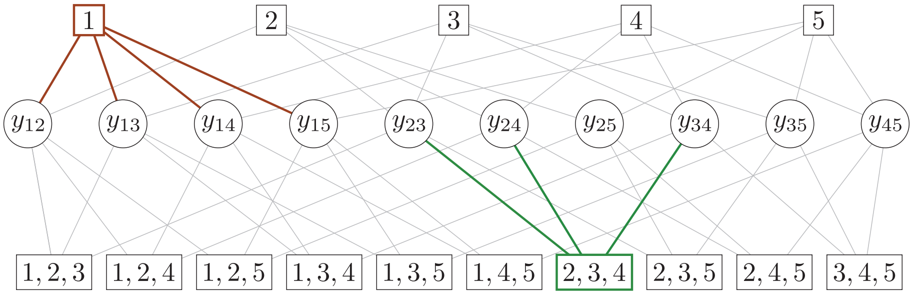


Figure 4.2: The factor graph for Figure 4.1 with factors (rectangular nodes) defined for maximal cliques. Colored factors correspond to cliques of the same color in Figure 4.1.

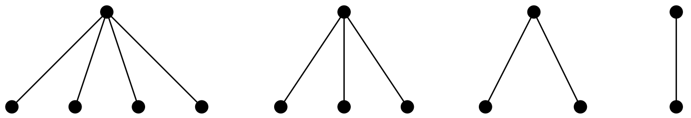


Figure 4.3: A 4-star, 3-star, 2-star and edge. Example subgraphs that are valid features in a Markov graph.

By definition, each clique in the graphical model contains only dyads that share an actor with every other dyad in the clique. These cliques can be divided into two sets: (1) those of size 3 corresponding to triangles in the social network (shown in green in Figure 4.1), and (2) those of sizes $2 < k < n-1$ corresponding to various k-stars in the social network (shown in red in Figure 4.1). A k-star is simply a person with (at least) k ties. Figure 4.3 shows examples of 4 through 1 stars in a social network.

The Markov graph framework defines a class of models by constraining the set of allowable features to be those that agree with the Markov assumption. A researcher must still choose from those allowable features the appropriate features for the network that is to be modeled. The allowable features for the star cliques are indicators for the occurrence of each k-star in the network. However, any star clique of size less than $n-1$ will be a subclique of a maximal star clique of size $n-1$. Moreover, any k-star also contains $\binom{k}{j}$ j-stars (with $j < k$), so some subgraphs will be counted multiple times if both k-star and j-star indicators are used. Each maximal $(n-1)$-star clique will contain all the tie variables for a single actor in the social network. Thus, Frank and Strauss (1986) point out, the star count features could be replaced with a histogram of actor degrees that would capture the same information while avoiding the multiple count problem (a statistic used by Moreno and Jennings in 1938, and that will reappear in Section 4.1.2).

The unique (that is, those not already possible with a star clique) allowable features for the triangle cliques are simply indicators for every potential triangle in the network.

Finally, Frank and Strauss assume homogeneity across the network so that all triangles and k-stars (of the same k) are equally likely, regardless of which actors they contain. Thus the features can be reduced to just counts of triangles and k-stars (as in Equation (1.12)).

Working with ERGMs

After Frank and Strauss, researchers continued using models that agreed with the Markov graph independence assumption. Wasserman and Pattison (1996) called such models $p*$ models since they generalize the p_1 model of Holland and Leinhardt. Because of their exponential family formulations such models are also referred to as *exponential random graph models* or ERGMs.

In addition to the features described above, most deployed models also include information about *nodal covariates*. Nodal covariates are properties of the people in the network such as age, sex, political affiliation, whether they smoke, etc. If the nodal covariates are expressed in an additional vector $\mathbf{x}$, then the log-likelihood can be broken down into the form

$$\mathcal{L}(\boldsymbol{\theta}|\mathbf{Y},\mathbf{x}) = \underbrace{\boldsymbol{\theta}_n{}^{\mathsf{T}}\mathbf{f}_n(\mathbf{Y})}_{\text{structural factors}} + \underbrace{\boldsymbol{\theta}_p{}^{\mathsf{T}}\mathbf{f}_p(\mathbf{Y},\mathbf{x})}_{\text{covariate factors}} - \log Z(\boldsymbol{\theta}) \tag{4.1}$$

where the features in the nodal covariate factors include information about both network structure and personal characteristics. Examples of such features are the number of ties whose endpoints are both smokers, or the squared difference in age between all tie endpoints.

Including nodal covariates helps distinguish "truly" structural effects from those explainable through personal characteristics—such as the homophily induced transitivity mentioned above. For the same reason, it is also important to include potentially confounding structural features: all sub-graphs of the structural features of interest. Including a parameter for each confounder ensures that the effect size of the desired feature is accurately estimated.

4.1.2 Model Degeneracy

By the early 2000's work had begun on MCMC methods for ERGMs. Unfortunately, much of that work called into question many earlier ERGM specifications.

Besag (2000) presents an MCMC method for testing models by constructing Markov chains whose stationary distributions are uniform over networks with specific feature values set to match the exact values of features in the data. Tests are then done by choosing a set of test features that are *not* used in the Markov chain and comparing the test feature values of the data to the test feature values generated by the Markov chain. Besag tests several proposed models from Wasserman and Pattison (1996) and Anderson et al. (1999) and finds that none of them produce sampled networks that come close to matching the real data.

Using MCMC to sample from ERGMs with varying parameter values Snijders (2002) found that many parameter values lead to either distributions that place most of their mass on almost (or entirely) full or empty networks. Other values lead to bimodal distributions that split mass between the almost (or entirely) full and empty modes but put very little mass on more realistic networks between those modes. Those distributions can have the same expected values as the observed data—since the mean will fall between the modes—but the observed data is nevertheless extremely unlikely according to the model. Furthermore, it is difficult for MCMC—particularly Gibbs sampling—to move between the modes of the distribution. This poor mixing will in turn result in a poor approximation of the gradient in Equation (1.6).

Handcock (2003, after Strauss, 1986) calls this phenomenon *model degeneracy*. A model is considered degenerate if it places most of its probability mass on very few networks. Recalling from Section 1.4.3 that F is the image of $\mathbf{f}(\mathfrak{Y})$ and that the MLE does not exist for points that are not in the interior of the convex hull of F (Barndorff-Nielsen, 1978, Ch. 9), a model and its parameters are considered *near degenerate* if the mean-value point they define is near the boundary of the convex hull of F. That means that the fitted model will place most of its probability mass on networks on the boundary of the convex hull, and those are generally less plausible networks.

Handcock also shows that the MCMC MLE will not exist if the convex hull of the samples to be reweighted

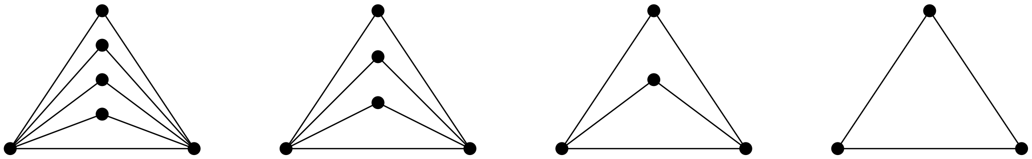


Figure 4.4: A 4-triangle, 3-triangle, 2-triangle and triangle.

in (1.37) does not contain the network being modeled. Learning, then, becomes a problem in degenerate models because they will cause MCMC to stay at a handful of networks that define a convex hull well away from the observed data. Finally, Handcock defines a model as *stable* if small changes in its natural parameterization result in small changes in its mean-value parameterization.

4.1.3 New Formulations of ERGMs

Intuitively, degeneracy can arise from the linear form of the model together with the relatively simple features explored. For example, assume that in some model a negative parameter is learned for density and a positive parameter is learned for the number of triangles. That means that, all other features kept equal, adding two triangles to the network increases its log-probability twice as much as adding one triangle. And adding three triangles increases its log-probability three times more than one triangle. The converse is true for edges: removing two is twice as log-probable as removing one, and so on. Only a small number of parameter values are able to strike a balance between the two—and even then they may result in degenerate bimodal distributions.

To alleviate that, researchers began trying higher order features such as 3-and 4-star counts as well as analogous generalizations of triangles to k-triangles (see Figure 4.4). If negative parameters are learned for larger stars and triangles, it would move mass away from fully connected networks. Note that k-triangles are not consistent with the Markov graph assumption and indeed the graphical model implied by them is fully connected.

Robins et al. (2007) observe that models with k-star features tended to learn (when they could be fit at all) k-star parameters that decrease in absolute value and alternate in sign as k increases. That phenomenon can be encoded into the model by adding an alternating k-stars feature, as well as the similar alternating k-triangles feature (Snijders et al., 2006):

$$s^a(\mathbf{Y}) = \sum_{k=2}^{n-1}(-1)^k \frac{s_k(\mathbf{Y})}{\lambda^{k-2}} \qquad \text{alternating } k\text{-stars} \tag{4.2}$$

$$t^a(\mathbf{Y}) = \sum_{k=1}^{n-2}(-1)^k \frac{t_k(\mathbf{Y})}{\gamma^{k-1}} \qquad \text{alternating } k\text{-triangles} \tag{4.3}$$

where s_k and t_k are the counts of k-stars and k-triangles, respectively. λ and γ are a fixed rates of decrease that must be either specified in advance or individually tested through an "outer loop" cross validation process.

The reliance on rates that are effectively parameters in Equation (4.2) and (4.3) means that the "features" are no longer solely a function of the data and the model is not of the form in (1.1) and thus is not (as written) an exponential family.

Curved Exponential Random Graph Models

Hunter and Handcock (2006) shows how to reformulate the above features so that the model can be written in the form of (1.13). This requires introducing two new sets of features, both of which assume an undirected network:

$$\mathbf{d}(\mathbf{Y}) = \left[\sum_i \mathbb{1}_{[Y_{i+}=1)]}, \sum_i \mathbb{1}_{[Y_{i+}=2)]}, \cdots \sum_i \mathbb{1}_{[Y_{i+}=n-1]}\right] \qquad \text{degree histogram} \tag{4.4}$$

$$\mathbf{v}(\mathbf{Y}) = [M_1(\mathbf{Y}), M_2(\mathbf{Y}), \ldots, M_{n-2}(\mathbf{Y})] \qquad \text{edgewise shared partner (ESP) histogram} \tag{4.5}$$

with

$$M_k(\mathbf{Y}) = \sum_{i<j} Y_{ij} \mathbb{1}_{[(\sum_l Y_{il}Y_{jl})=k]} \qquad \text{number of ties with } k \text{ shared partners} \tag{4.6}$$

Note that (4.4) excludes nodes with degree zero and (4.5) excludes dyads with no shared partners. Those exclusions avoid creating a linear dependency in the features and thus keep the family minimal.

To model the geometric decrease in parameter values, the natural parameters for these new histogram features are constrained to follow a predefined function. That constraint is incorporated into $\boldsymbol{\eta}(\boldsymbol{\theta})$, not $\mathbf{f}(\mathbf{y})$, so the model remains a valid curved exponential family:

$$\eta_k^d(\theta_d, \theta_w^d, \theta_r^d) = k\theta_d + \theta_w^d e^{\theta_r^d}(1-(1-e^{-\theta_r^d})^k) \qquad \text{degree parameter constraint with density} \tag{4.7}$$

$$\eta_k^t(\theta_w^t, \theta_r^t) = \theta_w^t e^{\theta_r^t}(1-(1-e^{-\theta_r^t})^k) \qquad \text{edgewise shared partner constraint} \tag{4.8}$$

The above constraints are applied to the natural parameters (with indexes adjusted appropriately to match their positions in $\mathbf{f}$) so that the feature $d_k(\mathbf{Y})$ gets natural parameter $\eta_k^d(\theta_d, \theta_w^d, \theta_r^d)$ and similarly for $t_k(\mathbf{Y})$. The θ_w embedded parameter is the usual multiplicative weight that indicates how important the feature is and whether increasing its value increases or decreases the probability of the data. The θ_r parameter is the rate at which the natural parameter values diminish as k increases. θ_d is the same density parameter that has been

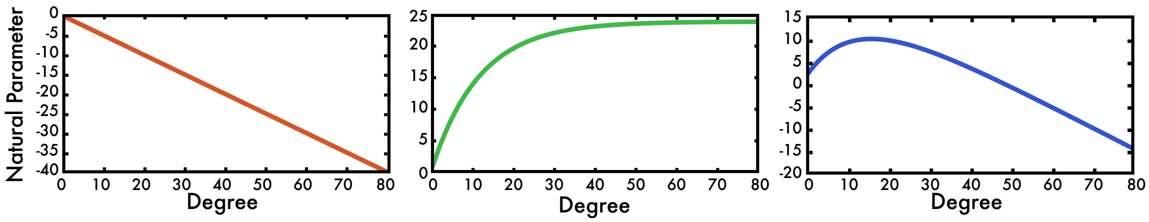


Figure 4.5: A stylized depiction of the parameter constraint (4.7) applied to the degree histogram. The left plot shows the "cost" of a negative density parameter. The middle shows the diminishing returns curve $e^{\theta_r}(1-(1-e^{-\theta_r})^k)$. Their sum is on the right.

used since the p_1 model, only here it is incorporated into the natural parameters for the degree histogram since that histogram implicitly includes the network density as $\frac{1}{2}\sum_k ks_k(\mathbf{Y})$.

The form of this parameter constraint has an appealing intuitive interpretation. The function $e^{\theta_r}(1-(1-e^{-\theta_r})^k)$ defines a curve that gradually slows in growth as k increases (Figure 4.5, middle). That reflects the notion that increases in degree or number of shared partners are beneficial, but that there is a diminishing rate of return as more edges or shared partners are added. With this constraint, three triangles are no longer $\exp(3)$ times more probable than one triangle.

In the context of diminishing returns, the density parameter θ_d in (4.7) can be interpreted (assuming $\theta_d < 0$) as the "cost" of adding an edge (Figure 4.5, left). The combination of the cost with the diminishing returns function can identify the point at which the benefit a person receives from having a higher degree is outweighed by the cost required to (for example) maintain those edges. That point (Figure 4.5, right) can be interpreted as the "ideal" degree for the network.

The combinations of histogram features and parameter constraints above are referred to together in models as the geometrically weighted degree (GWD) and the geometrically weighted edgewise shared partners (GWESP). Similarly to the nesting of 2-stars within triangles, these new features also require accounting for the nesting of dyadwise shared partners within edgewise shared partners:

$$\mathbf{q}(\mathbf{Y}) = [P_1(\mathbf{Y}), P_2(\mathbf{Y}), \ldots, P_{n-2}(\mathbf{Y})] \qquad \text{the dyadwise shared partner histogram} \qquad (4.9)$$

with

$$P_v(\mathbf{Y}) = \sum_{i<j} \mathbb{1}_{[(\sum_k Y_{ik}Y_{jk})=v]} \qquad \text{the number of dyads with } v \text{ shared partners} \qquad (4.10)$$

The only difference between (4.10) and (4.6) is that (4.10) does not consider whether or not a tie exists between i and j and thus $U_k(\mathbf{Y}) \geq T_k(\mathbf{Y})$. The natural parameters for (4.9) are constrained exactly as those for (4.5), but with their own embedded weight and rate parameters. Together, those are called the the geometrically weighted dyadwise shared partners (GWDSP). Including GWDSP accounts for any baseline tendency for pairs to have shared partners and allows GWESP to be interpreted as capturing the effect of transitivity in the network.

The curved exponential formulation allows many features to be used that would not be possible in an unconstrained model. Using the degree histogram in an unconstrained ERGM would almost certainly lead to an unlearnable model. It is unlikely that any real data will contain nodes of each possible degree, which is needed to ensure its features do not lie on the boundary of the convex hull of F. The same is true for shared partner counts. By constraining the parameters, that concern disappears. However, recall from Section 1.4.3 that the log-likelihood for models defined with a curved exponential formulation is not, in general, convex. So the convenience of the non-linear parameter constraint comes at the cost of sacrificing convexity.

4.2 Latent ERGMs

ERGMs, like almost all other existing social network analysis techniques, have been used almost solely on survey data. In addition to being usually static and binary, survey data—especially when it concerns recalled behaviors—is subject to some unknown cognitive processing that may cause the observed answers to differ from reality. Krackhardt suggests that one interpretation of the poor recall findings of e.g. Bernard et al. (1982) is that they "simply constitute evidence that one should not bother collecting behavioral data, since they do such a poor job of capturing the cognitions which live in peoples' heads" (Krackhardt, 1987). In other words, if the object of study is the subjects' perceived structure of social relationships, then surveys are far more preferable measurement instruments than recordings of behavior. Krackhardt clarifies that, of course, different research efforts may be more or less "interested in discovering the behavioral patterns, the cognitive patterns, or the relationship between them" (Krackhardt, 1987).

This distinction between cognitive social structure and behavioral social structure suggests a new approach for modeling networks of social behavior. Rather than modeling the behavior directly, we can model some abstract social structure that is itself unobservable but that does help explain any observed behavior. Note that survey answers, too, are only indirect observations of the latent structure. Reasoning about this latent structure when given only social behavior data requires, in the words of Marsden (1990), "some means of abstracting from these empirical acts to relationships or ties."

That process of abstracting from noisy observation to latent ties is implicit in other social behavior studies that define simple thresholds or heuristics to discard observations that are believed *a priori* to be noise. The remaining observations are then considered a direct observation of the "true" network (e.g. Palla et al., 2006; Kossinets and Watts, 2006). Such methods are unsatisfying because the definition of noise is ad hoc and it does not propagate any uncertainty about the latent structure into later analyses.

The flexible form of (1.3) makes it straightforward to extend ERGMs to jointly model both noisy observations and latent structure. The latent network is still modeled with a set of random variables $\mathcal{Y}$ corresponding to its adjacency matrix. To those, we add a set of observation variables $\mathcal{X}$ that capture any behavior data under consideration. $\mathcal{Y}$ are then simply treated as latent, or hidden, variables and marginalized out.

If there the latent structure, the observed structure, and the relationship between them have distinct properties that are best considered individually, that can easily be done by separating those properties into different sets of factors. In this work, we use only models that consider latent structure and the relationship between the latent structure and observed data. For those models, the marginal distribution of the observations can be written as

$$\begin{aligned} p(\mathbf{X}) &= \sum_{\mathbf{Y}\in\mathfrak{Y}} P(\mathbf{X},\mathbf{Y}) && (4.11)\\ &= \sum_{\mathbf{Y}\in\mathfrak{Y}} \frac{1}{Z(\boldsymbol{\theta})} e^{\boldsymbol{\eta}(\boldsymbol{\theta})^{\mathsf{T}}\mathbf{f}(\mathbf{X},\mathbf{Y})} && (4.12)\\ &= \sum_{\mathbf{Y}\in\mathfrak{Y}} \frac{1}{Z(\boldsymbol{\theta})} \exp\left\{ \underbrace{\boldsymbol{\eta}_s(\boldsymbol{\theta})^{\mathsf{T}}\mathbf{f}_s(\mathbf{Y})}_{\text{structural factors}} + \underbrace{\boldsymbol{\eta}_o(\boldsymbol{\theta})^{\mathsf{T}}\mathbf{f}_o(\mathbf{Y},\mathbf{X})}_{\substack{\text{factors relating}\\ \text{observations to structure}}} \right\} && (4.13)\end{aligned}$$

The structural features $\mathbf{f}_s$ can be any of those used in traditional ERGMs. The new observation features $\mathbf{f}_o$ will need to be defined according to the problem of interest.

Related Work Butts (2003) presents a method for explicitly modeling the error of survey respondents while also recovering the "true" latent network that generated the survey observations. That approach is similar in spirit to our, but different in motivation. For Butts, both the latent and observed networks are assumed to be binary. Any disagreement between them is attributable to informant inaccuracy. The relationship that we are modeling between the observed data and the latent network is not one of measurement error, but rather that of the generation of behavior within some abstract social structure. (Though extensions to model measurement error are possible, and discussed below in Section 4.4.) Another key difference between our approach and that of Butts is that he uses only a dyad-independent model for the latent network. We consider a richer model that accounts for dependence between dyads.

4.2.1 Latent Networks and Conversation Data

In our model, the specific observation data used is $\mathbf{C}$, the matrix of proportions of time spent in conversation (defined in Section 3.5). The latent network is assumed to be binary.

We define two new factors that relate the proportion of time two people spend in conversation to the probability of a tie between them existing in the latent network. These new factors, similar to the GWD and GWESP factors, involve histograms of features combined with a "diminishing returns" parameter constraint.

The new histograms are $\mathbf{c}(\mathbf{Y}, \mathbf{X})$ and $\mathbf{n}(\mathbf{Y}, \mathbf{X})$ where

$$c_v(\mathbf{Y}, \mathbf{X}) = \sum_{ij} Y_{ij} \mathbb{1}_{[zv \leq C_{ij} < z(v+1)]} \tag{4.14}$$

$$n_v(\mathbf{Y}, \mathbf{X}) = \sum_{ij} (1 - Y_{ij}) \mathbb{1}_{[zv \leq C_{ij} < z(v+1)]} \tag{4.15}$$

where z is a pre-specified bin width and the indexes v range from zero to whatever upper limit is necessary to include the maximum observed value. For our data, $z = 0.14\% \approx 3$ minutes for the pair with the largest amount of overlapping recording time. Since pairs are only counted in the $\mathbf{c}$ histogram if an edge exists between them in the latent network, we refer to $\mathbf{c}$ as the "edge on" histogram. Similarly, we refer to $\mathbf{n}$ as the "edge off" histogram. As with the degree and shared partner histograms described above, one bin is excluded from the set of model features to avoid a linear dependence and keep the family minimal.

To model the intuition that spending more time in conversation increases the probability of a tie, but only to a point, we use the same "dimishing returns" constraint on the natural parameters for $\mathbf{c}$ and $\mathbf{n}$ as that of GWD and GWESP. To model the notion that there is some "cost" associated with time spent in conversation, we also include a linearly changing weight analogous to that used to model network density in the degree histogram feature. Taken together, those two assumptions mean that the parameter constraints for $\mathbf{c}$ and $\mathbf{n}$ are identical in form to those defined in Equation (4.7). The specific functions that place natural parameters on the bins of $\mathbf{c}$ and $\mathbf{n}$, along with their embedded parameters, are thus

$$\eta_k^c(\theta_c, \theta_w^c, \theta_r^c) = k\theta_c + \theta_w^c e^{\theta_r^c}(1 - (1 - e^{-\theta_r^c})^k) \qquad \text{for } \mathbf{c} \tag{4.16}$$

$$\eta_k^n(\theta_c, \theta_w^n, \theta_r^n) = k\theta_c + \theta_w^n e^{\theta_r^n}(1 - (1 - e^{-\theta_r^n})^k) \qquad \text{for } \mathbf{n} \tag{4.17}$$

Note that the first embedded parameter, θ_c, is shared between the histograms. θ_c is the parameter meant to model the "cost" of time spent in conversation (if it is negative). Since the total amount of time spent in conversation is split between the two histograms, the same cost must be applied to both.

θ_w^c and θ_w^n are the multiplicative weights, and θ_r^c and θ_r^n are the rate parameter. Presumably, θ_w^c will be positive, denoting the fact that more time in conversation increases the probability of a latent tie existing between a pair. Similarly, θ_w^n should be negative, since more time in conversation will decrease the probability of there being no tie between a pair.

Of course, since there is no way for the model to distinguish *a priori* which histogram is "edge on" and which is "edge off", we place a Gaussian prior on $\boldsymbol{\theta}$ to encode those assumptions. We further restrict the prior to a simple diagonal covariance, which is equivalent to a single univariate Gaussian prior on each component of $\boldsymbol{\theta}$. The specific prior parameters that we use in our experiments are as follows. All multiplicative weights have a mean of zero and unit variance. All rate weights have a mean of 1 and variance of 0.5. The "edge on" conversation weight has a mean of 1, and the "edge off" weight has a mean of -1. Both have unit variance. The addition of the prior changes the marginal probability of the observations from (4.12) to

$$p(\mathbf{X}) = \prod_i \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{(\theta_i - \mu_i)}{2\sigma_i^2}} \sum_{\mathbf{Y} \in \mathfrak{Y}} \frac{1}{Z(\boldsymbol{\theta})} e^{\boldsymbol{\eta}(\boldsymbol{\theta})^\mathsf{T} \mathbf{f}(\mathbf{X},\mathbf{Y})} \tag{4.18}$$

where μ_i and σ_i^2 are the prior mean and variance matrix, respectively, of θ_i.

The structural factors that we use (in addition to the conversation factors) are the GWD, GWDSP, and GWESP.

4.2.2 Fitting the Model

With the prior, parameter estimation becomes *maximum a priori* (MAP) estimation, and is no longer technically maximum likelihood estimation. We can still easily employ gradient-based numerical optimization methods, though. The gradient of the log-likelihood for models of the form of (4.18) is

$$\frac{\partial}{\partial \boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}|\mathbf{X}) = \boldsymbol{\nabla}\boldsymbol{\eta}(\boldsymbol{\theta})^\mathsf{T} \left(\mathop{\mathrm{E}}_{\mathbf{Y}} \left[\mathbf{f}(\mathbf{X},\mathbf{Y}) \middle| \mathbf{X}, \boldsymbol{\theta} \right] - \mathop{\mathrm{E}}_{\mathbf{X},\mathbf{Y}} \left[\mathbf{f}(\mathbf{X},\mathbf{Y}) \middle| \boldsymbol{\theta} \right] \right) + \boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \tag{4.19}$$

where $\boldsymbol{\nabla}\boldsymbol{\eta}(\boldsymbol{\theta})$ is the Jacobian of $\boldsymbol{\eta}(\boldsymbol{\theta})$, as it was in (1.16), and $\boldsymbol{\Sigma}$ is the diagonal covariance matrix of the prior. The conditional expectation in (4.19) that did not appear in (1.16) is the result of introducing, and then marginalizing out, the latent variable $\mathcal{Y}$.

Both of the expectations required for (4.19) can be approximated with MCMC (as described in Section 1.4.5). For a general Metropolis-Hastings chain approximating the conditional expectation on the left the proposal distribution also conditions on the observed value of $\mathbf{X}$ and thus takes the form $q(\mathcal{Y} = \mathbf{Y}'|\mathbf{Y}, \mathbf{X})$.

We use first-order stochastic gradient ascent to find the MAP estimate. Assuming that we can observe multiple networks, we take one gradient step for each observed network. Even using the new ERGM features devised to avoid degeneracy, we found that learning could diverge if gradient steps moved too far beyond a realistic range of the parameters. To avoid this, we normalize the gradient by its norm so that no step will increase any weight by more than 1.

4.2.3 Experimental Results

We performed two sets of evaluations of our model: one on synthetic data and one on the Spoken Networks data. On the spoken networks data, we restrict our analyses to the final 6 weeks. Those are the weeks with the most data and during which the networks' properties seem the most stable. That stability (perhaps) justifies treating the separate networks as independent observations from the same distribution.

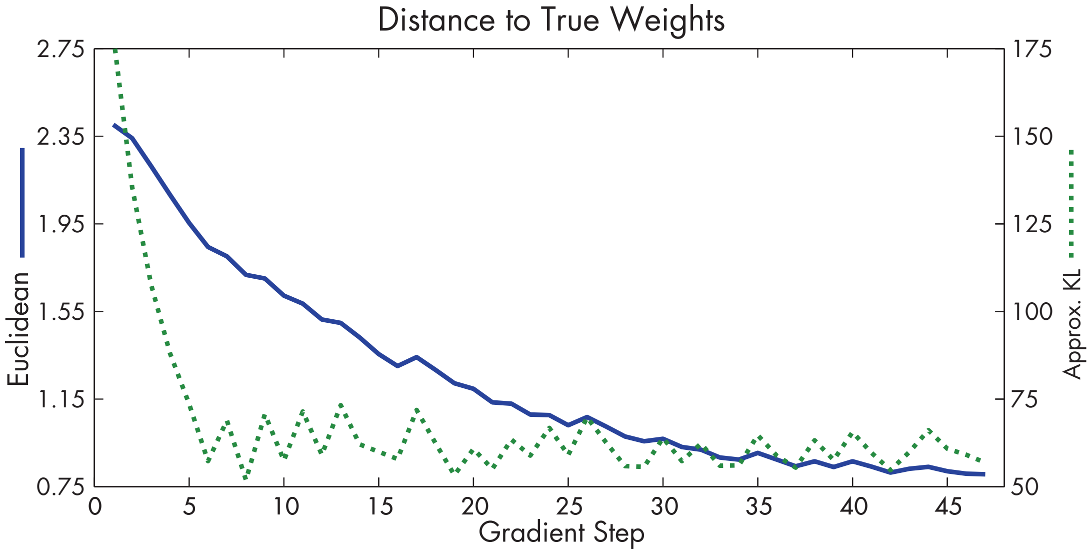


Figure 4.6: Distances from true to learned weights during learning on synthetic data.

Both evaluations take the same form. Each must learn the parameters for a 24 node network given 6 separate noisy observations for that network. Each set of observations contains information about all of the $\binom{24}{2} = 276$ pairs. We assume that all 6 observations were generated by a single, fixed distribution and thus use all six observations to learn one set of parameters for the model specified above. Note that learning one set of parameters is not the same as learning a single latent structure. Each of the 6 observations may be generated by different latent networks. We only assume that all of the latent networks have the same global properties (density, transitivity, etc.).

After the parameters have been learned, they are used to sample from the posterior distribution for the latent social network of each of the 6 examples separately. The mean of the samples for each latent edge variable is interpreted as the posterior probability of that edge. A concrete realization of the posterior network can be had by fixing a threshold and assembling the network of all edges with posterior probability greater than that threshold.

Synthetic Data

To test that our model is capable of recovering latent structure we ran it on a synthetic data set designed to simulate our real data. We used weights that had been fit to actual data to generate synthetic latent networks and times in conversation. In the synthetic data we know both the true parameters and the true latent structure, so we can evaluate our technique in terms of how well it recovers the original parameters and how well it can

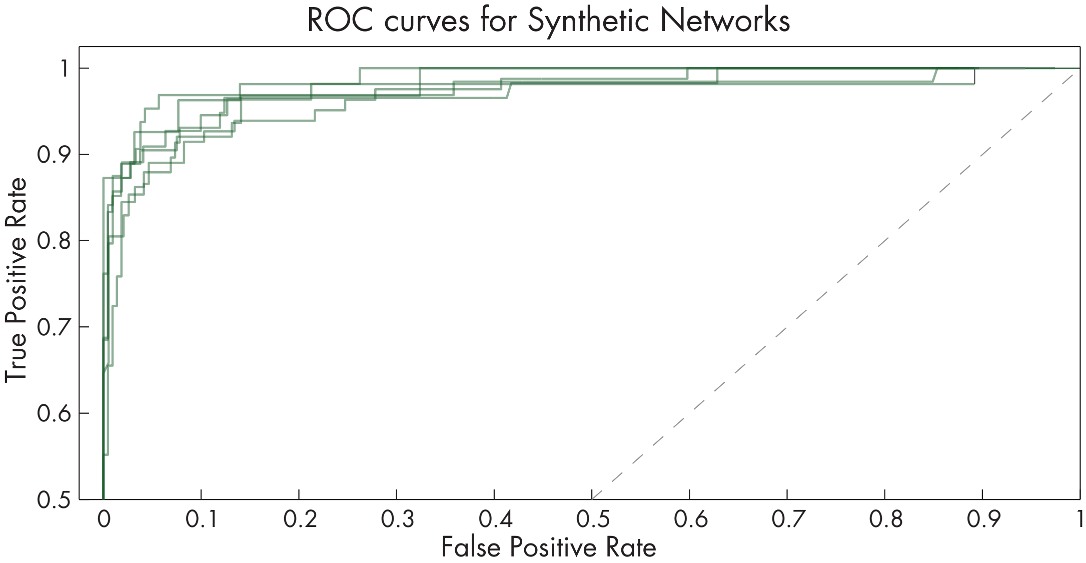
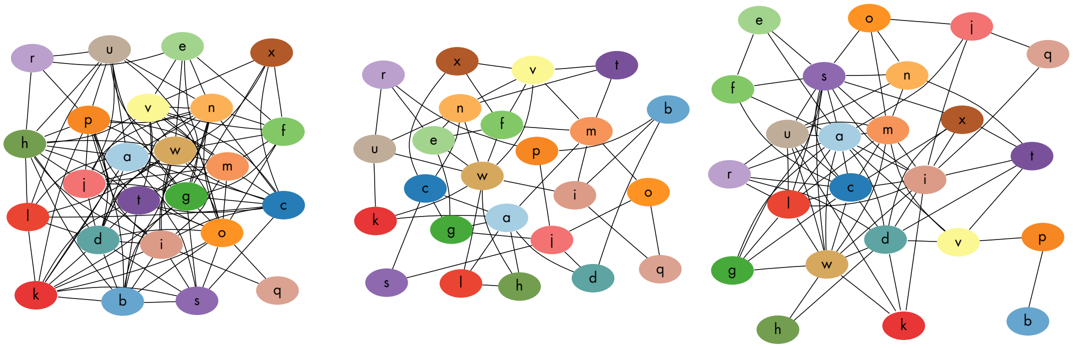


Figure 4.7: Solid lines are ROC curves for 6 synthetic data experiments. Dashed line indicates equal TPR and FPR.

Table 4.1: Mean performance on synthetic data at varying thresholds.

Threshold	Accuracy	True Pos. Rate	False Pos. Rate
0.50	0.874	0.963	0.152
0.75	0.930	0.926	0.069
0.90	0.948	0.891	0.036
0.95	0.955	0.859	0.017



(a) Conversation network.

(b) Inferred network.

(c) Survey network.

Figure 4.8: Conversation, inferred, and survey networks for week 4.

infer the latent structure.

Figure 4.6 shows, for each gradient step in the learning procedure, both the Euclidean distance between the current learned weights and the true weights and an approximation of the Kullback-Leibler divergence between the two sets of weights. The expectation required for the KL divergence is estimated using only the 6 training examples, so the KL approximation is extremely coarse.

Figure 4.7 shows the ROC curves for all 6 examples. For a threshold t, the true positive rate is the number of edges with posterior mean greater than t that are in the true latent network, divided by the total number of edges in the true latent network. The false positive rate is the number of edges with mean greater than t that are *not* in the true network, divided by the total number of non-edges (unconnected pairs) in the true network. As t is raised, the true positive rate increases sharply while the false negative rate remains small. Table 4.1 shows specific values for aggregate accuracy, true positive rate, and false positive rate at 4 different thresholds. For example, at a threshold of 0.75, we can recover the latent structure with 93% accuracy while only suffering a 7% false positive rate. In the synthetic data, the model is able to recover the latent structure quite successfully.

Spoken Networks Data

For the Spoken Networks data, evaluation of the inferred latent networks is difficult since the hidden structure that we are trying to recover is genuinely hidden—there is no ground truth to which we can compare it. However, recall that the subjects answered survey questions about their interactions. Since they only recorded sensor data during school, we can use the responses to the research and coursework questions to build a separate observation of the same latent network. We stress that these surveys should not be considered ground truth, but rather a second noisy observation of the same latent social structure. Nevertheless, one would expect

Table 4.2: Agreements between survey networks and random networks, raw conversation networks, and inferred conversation networks.

				Inferred vs. Random		Inferred vs. Raw		Raw vs. Random	
Week	Random	Raw	Inferred	diff.	p	diff.	p	diff.	p
1	0.595	0.616	0.717	0.123	.0012	0.101	.0058	0.021	.3035
2	0.544	0.629	0.696	0.151	1.2E-4	0.067	.0493	0.085	.0214
3	0.511	0.633	0.638	0.127	.0013	0.040	.4578	0.123	.0018
4	0.664	0.803	0.823	0.159	1.0E-5	0.019	.2788	0.139	1.1E-4
5	0.768	0.794	0.830	0.062	.0350	0.036	.1400	0.026	.2313
6	0.768	0.822	0.844	0.076	.0117	0.023	.2383	0.054	.0592
Overall	0.641	0.713	0.758	0.116	1.4E-13	0.045	.0016	0.071	6.0E-6

there to be some agreement between our inferred structures and those expressed in the surveys.

Indeed, that is what we found. Using a threshold of 0.5, we compared the inferred networks to the survey networks and computed the number of edges for which they agreed. We compute the same agreements for random networks with the same expected density as the survey network, and for the network formed by the raw conversation data (that is, a network with edges for any pair who ever spent time in conversation). Results for these comparisons are in Table 4.2. Samples of the raw, inferred, and survey networks for week 4 are in Figure 4.8.

For all weeks, the inferred networks have better agreement with the surveys than random networks, and the improved agreements are statistically significant (one-tailed t-test). The inferred networks also have better agreement than the raw networks in all weeks. While those improvements are not statistically significant in all weeks, they are significant in aggregate.

Interpreting the parameters As with any ERGM, the learned parameters provide information about the process that generated the network. Of specific interest for the latent ERGM model presented here are the parameters defining the relationship between observed time in conversation and the existence of a latent social tie. Once the MAP parameters $\hat{\boldsymbol{\theta}}$ are learned, we can easily compute the conditional probability of time in

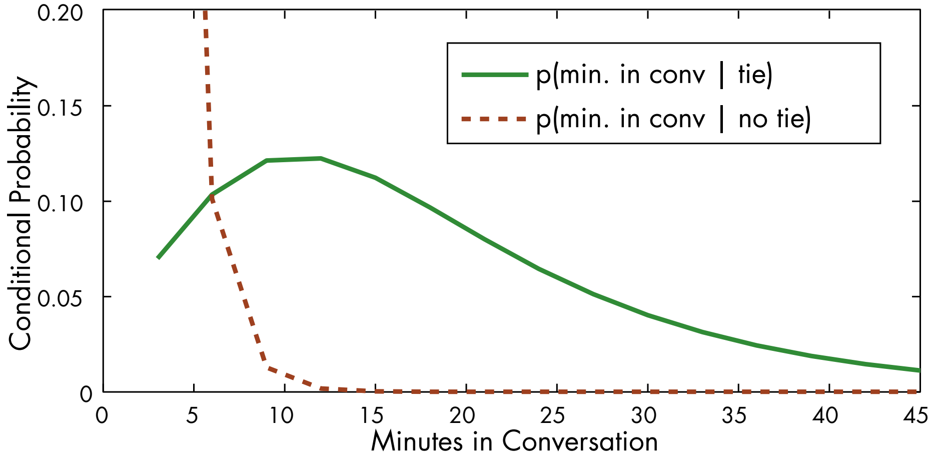


Figure 4.9: Conditional probabilities of time in conversation given existence of latent social tie.

conversation given the state of the latent tie as

$$p(\mathcal{X}_{ij} \approx k | \mathcal{Y}_{ij} = 1) = \frac{1}{Z_c} e^{\eta_k^c(\hat{\theta}_c, \hat{\theta}_w^c, \hat{\theta}_r^c)} \tag{4.20}$$

$$p(\mathcal{X}_{ij} \approx k | \mathcal{Y}_{ij} = 0) = \frac{1}{Z_n} e^{\eta_k^n(\hat{\theta}_c, \hat{\theta}_w^n, \hat{\theta}_r^n)} \tag{4.21}$$

where $\mathcal{X}_{ij} \approx k$ is shorthand for the value of C_{ij} falling in the k-the bin of the histogram: $zk \leq C_{ij} < z(k+1)$. $Z_c = 1 + \sum_k e^{\eta_k^c(\theta_c, \theta_w^c, \theta_r^c)}$ normalizes the distribution, with Z_n is defined similarly. The $1 + \ldots$ accounts for the left out bin, whose probability is $1/Z_c$ (or $1/Z_n$). This transformation of $\boldsymbol{\eta}$ to $p(\mathcal{X}|\mathcal{Y})$ is precisely an example of moving from the natural parameterization to the mean-value parameterization that is described in Section 1.4.3.

Plots of these conditional probabilities for the 6 weeks of Spoken Networks data considered above are shown in Figure 4.9. This plot shows several interesting facts about the network that can be derived from the distributions. The point where the lines cross is the point at which a latent tie becomes more likely than a non-tie. This threshold (about 6 minutes in the Spoken Networks data) could be interpreted as the point at which conversation transitions from being courteous chit-chat and instead becomes the expression of a durable social tie. The peak in the probability of time in conversation given an existing tie (about 12 minutes) could be seen as the optimum time to spend in conversation in order to maintain a tie—beyond that diminishing returns do not offset the increasing cost. Of course, these interpretations are very speculative, but they provide an example of questions that could be addressed using this modeling methodology with social behavior data.

4.3 Multi-valued Time-inhomogeneous Temporal ERGMs

The latent network model of the last section is still a static model. We fit it to multiple networks, but only under the assumption that all of the observed networks come from the same distribution. That discards any information about the evolution of the network over time. Since one of the exciting novel aspects of social behavior data is its natural longitudinal nature, ignoring the temporal aspect of the data is unsatisfying. We would like to know how influences relate to changes in the network structure, and even whether the importances of those influences may themselves change over time.

This section provides an overview of the slim existing literature on temporal extensions to ERGMs and then presents our contribution of a time-inhomogeneous temporal ERGM.

4.3.1 Existing Temporal ERGMs

Temporal ERGMs have been discussed for as long as ERGMs have existed. Holland and Leinhardt (1981) explicitly mention that their p_1 model could be extended to include time series. Wasserman and Iacobucci (1988) make that extension by allowing for multiple observations of a network at T different timesteps. Let $\mathbf{Y}$ now be the sequence of all observed sociomatrices and $\mathbf{Y}^t$ be the network observed at time t. To the static "within timestep" features of Holland and Leinhardt's p_1 model, Wasserman and Iacobucci add a set of dynamic "between timestep" features that count how many ties persist between timesteps, how many ties become mutual, and how many become asymmetric.

Wasserman and Iacobucci still assume total dyad independence both within and between timesteps, but they also assume that the number of observed timesteps will be small and that the above features will be computed for all $(t, t') : t < t'$ pairs of timesteps. While that provides for a flexible model capable of capturing long-range dependencies, it cannot scale to handle automatically collected data that is observed at very fine scales and for very long periods of time (to say nothing of the unrealistic dyad independence assumption).

If thought of as a Markov chain, the order of Wasserman and Iacobucci's model is T. Earlier, Holland and Leinhardt (1977) proposed modeling each edge with a continuous-time Markov chain. If that were done with Wasserman and Iacobucci's model it would mean computing temporal features only for pairs of times (t, t') where there is no t'' such that $t < t'' < t'$.

Robins and Pattison (2001) present the first general model for temporal ERGMs that assumes a first-order

Markov independence between timesteps. Specifically, they model a sequence of networks as

$$
\begin{aligned}
p(\mathbf{Y}^1, \ldots \mathbf{Y}^T) &= p(\mathbf{Y}^1) \prod_{t=2}^{T} p(\mathbf{Y}^t | \mathbf{Y}^{t-1}) &(4.22)\\
&= \frac{1}{Z(\boldsymbol{\eta}_s)} \exp\left\{\boldsymbol{\eta}_s{}^{\mathsf{T}} \mathbf{f}_s(\mathbf{Y}^1)\right\} \times \\
&\prod_{t=2}^{T} \frac{1}{Z(\boldsymbol{\eta}, \mathbf{Y}^{t-1})} \exp\left\{ \underbrace{\boldsymbol{\eta}_s{}^{\mathsf{T}} \mathbf{f}_s(\tilde{\mathbf{Y}}^t)}_{\text{static factors}} + \underbrace{\boldsymbol{\eta}_d{}^{\mathsf{T}} \mathbf{f}_d(\tilde{\mathbf{Y}}^t, \mathbf{Y}^{t-1})}_{\text{dynamic factors}} \right\} &(4.23)
\end{aligned}
$$

where

$$
\begin{aligned}
Z(\boldsymbol{\eta}) &= \sum_{\tilde{\mathbf{Y}}^t \in \mathfrak{Y}^t} \exp\left\{\boldsymbol{\eta}_s{}^{\mathsf{T}} \mathbf{f}_s(\tilde{\mathbf{Y}}^t)\right\} &(4.24)\\
Z(\boldsymbol{\eta}, \mathbf{Y}^{t-1}) &= \sum_{\tilde{\mathbf{Y}}^t \in \mathfrak{Y}^t} \exp\left\{\boldsymbol{\eta}_s{}^{\mathsf{T}} \mathbf{f}_s(\tilde{\mathbf{Y}}^t) + \boldsymbol{\eta}_d{}^{\mathsf{T}} \mathbf{f}_d(\tilde{\mathbf{Y}}^t, \mathbf{Y}^{t-1})\right\} &(4.25)
\end{aligned}
$$

This easily allows any static feature from the existing ERGM literature to be used alongside any newly developed dynamic feature.

In their specific implementation Robins and Pattison only consider a data set with two timesteps (temporal network data truly was scarce!) and they condition on timestep 1 so the static features only apply to timestep 2. They explain that it is simple to generalize their chain graph model to more timesteps if one makes a time-homogeneity assumption, which "although not atypical, would be a strong [assumption], implying that the systematic processes are unchanged in effect size and direction across the entire measurement period."

Hanneke et al. (2010, building on preliminary work by Guo et al., 2007) are the first to make that generalization, and to do so they rely on exactly such a time-homogeneity assumption. Like Robins and Pattison, Hanneke et al. condition on the first timestep. That conditioning becomes more important because they also restrict their model to use only dynamic features that involve at most one dyad in the later timestep:

$$
\mathbf{f}(\mathbf{Y}^t, \mathbf{Y}^{t-1}) = \sum_{i<j} \mathbf{f}_{ij}(Y^t_{ij}, \mathbf{Y}^{t-1}) \tag{4.26}
$$

In other words, only those features that allow the conditional distribution $p(\mathbf{Y}^t | \mathbf{Y}^{t-1})$ to factor over dyads in $\mathbf{Y}^t$ so that

$$
p(\mathbf{Y}^t | \mathbf{Y}^{t-1}) = \prod_{i<j} p(Y^t_{ij} | \mathbf{Y}^{t-1}) \tag{4.27}
$$

This "conditional dyad independence" assumption, combined with conditioning on the first timestep, allows for the same tractability as the static dyad independence assumption and makes exact learning possible. Specifically, (4.25) simplifies to

$$
Z(\boldsymbol{\eta}, \mathbf{Y}^{t-1}) = \prod_{i<j} \sum_{y'_{ij} \in \mathfrak{Y}^t_{ij}} \exp\left\{\boldsymbol{\eta}^{\mathsf{T}} \mathbf{f}_{ij}(y'_{ij}, \mathbf{Y}^{t-1})\right\} \tag{4.28}
$$

Note that (4.26) also allows any static dyad independent feature to be used since those are functions of just Y_{ij}^t.

In addition to making learning tractable, the conditional dyad independence assumption admits easier analysis of properties of the distribution in (4.27). Upper and lower bounds on the entropy and density of the model can be derived. For a small network, Hanneke et al. compute the entropy of (4.27) for varying parameter values and show that the entropy is high for small absolute parameter values and that the entropy varies smoothly as parameters change. These analyses suggest that the models available through the conditional dyad independence assumption are not degenerate and spread their probability mass across many networks.

4.3.2 Our Model

The assumption of time-homogeneity in the model of Hanneke et al. (2010) is very common, but Robins and Pattison warn that "this assumption, although not atypical, would be a strong one, implying that the systematic processes are unchanged in effect size and direction across the entire measurement period" (Robins and Pattison, 2001). In other words, while a time-homogeneous model can capture the surface changes in a network—how ties come and go—they cannot capture changes in the underlying properties of the network, like density or tendency to transitivity. Worse, most time-homogenous models also assume that the underlying properties are stationary through time, which may not be the case in real social networks. Indeed, when testing their model, Hanneke et al. explain that they have to discard the first few observations from their data since they seem to be outliers when compared to later observations. A time-inhomogeneous model is one way of ensuring that there is no stationary distribution for the underlying properties, and thus may be more suitable for processes in which those properties evolve over time.

Perhaps the simplest way of achieving time-inhomogeneity is to have, as Wasserman and Iacobucci (1988) did, completely distinct parameters for each timestep. Of course, that would not scale well to longer sequences. More importantly, it would be hard to detect trends in the changes of underlying properties if each timestep has its own completely unconstrained set of parameters. Ordinary stochastic variation could result in sudden changes in parameter values that mask longer term trends. Some *post hoc* regression of yet another function to the learned parameters might be able uncover long term trends, but interpreting that regression—especially any uncertainty around it—would require a complex set of assumptions. Instead, what is required is a constrained time-inhomogeneity, one that allows for different parameters at each timestep but smooths away short term fluctuations in order to reveal long range trends.

We propose exactly such a model that leverages the parameter constraints of a curved exponential family to enforce smoothness while allowing time-inhomogeneity. Our model is also designed to directly examine the network of behavior—not an unobservable latent network. Since the behavior network involves weighted edges, we must also define a new set of features capable of exploiting the information in the edge weights.

Multi-Valued ERGMs

To retain some of the computational advantages of pseudo-likelihood and Gibbs sampling, we discretize continuous edge weights into v discrete, ordinal values. To permit comparisons with binary-valued models, the values are scaled so that the smallest is 0 and the largest is 1. Simple network statistics can be redefined for this model in the same straightforward manner presented in Section 3.5: the density of a network is the sum of its edge values; a node's degree is the sum of the values of the edges incident to that node.

More complicated features that involve subgraphs require defining the intensity of a subgraph. As in Section 3.5.3, we use the geometric mean of the edge values composing the subgraph. For example, a shared partner k for nodes i and j is defined to be a partner of intensity $(y_{ik}y_{jk})^{\frac{1}{2}}$, where y_{ij} represents the multi-valued edge between nodes i and j. The count of shared partners for a pair, SP_{ij} is the sum of these intensities:

$$SP_{ij} \triangleq \sum_k (y_{ik}y_{jk})^{\frac{1}{2}} \tag{4.29}$$

To model edgewise shared partners we take the product of an edge's value with its shared partner sum:

$$ESP_{ij} \triangleq y_{ij} SP_{ij} \tag{4.30}$$

Note that if $v = 2$ and all edge values are either 0 or 1, then our features are equivalent to the traditional ERGM features.

Time-inhomogeneous ERGMs

For a time-homogeneous model with s static features and d dynamic features, a feature vector of length $s+d$ can be computed for each pair of adjacent timesteps. Time-homogeneity allows all of these vectors to be summed into a single vector (clearly also of length $s + d$) that summarizes the entire sequence. By doing that, (4.23) can be rewritten as

$$\begin{aligned} p(\mathbf{Y}^1, \ldots, \mathbf{Y}^T) = {} & \frac{1}{Z(\boldsymbol{\eta}_s)} \exp\left\{\left\langle \boldsymbol{\eta}_s\,,\, \mathbf{f}_s(\mathbf{Y}^1)\right\rangle\right\} \times \\ & \frac{1}{Z_T} \exp\left\{\left\langle \boldsymbol{\eta}_s\,,\, \sum_{t=2}^{T} \mathbf{f}_s(\mathbf{Y}^t)\right\rangle + \left\langle \boldsymbol{\eta}_d\,,\, \sum_{t=2}^{T} \mathbf{f}_d(\mathbf{Y}^t, \mathbf{Y}^{t-1})\right\rangle\right\} \end{aligned} \tag{4.31}$$

where $Z_T = \prod_{t=2}^{T} Z(\boldsymbol{\eta}, \mathbf{Y}^{t-1})$ and we have switched inner product notation so $\langle \boldsymbol{u},\, \boldsymbol{v}\rangle \triangleq \boldsymbol{u}^\mathsf{T}\boldsymbol{v}$.

For our time-inhomogeneous model we compute the same set of features for each timestep but can no longer collapse them into one sum since we also allow each timestep to have its own set of parameters:

$$\begin{aligned} p(\mathbf{Y}^1, \ldots, \mathbf{Y}^T) = {} & \frac{1}{Z(\boldsymbol{\eta}_s)} \exp\left\{\left\langle \boldsymbol{\eta}_s\,,\, \mathbf{f}_s(\mathbf{Y}^1)\right\rangle\right\} \times \\ & \prod_{t=2}^{T} \frac{1}{Z(\boldsymbol{\eta}_s^t, \mathbf{Y}^{t-1})} \exp\left\{\left\langle \boldsymbol{\eta}_s^t,\, \mathbf{f}_s(\mathbf{Y}^t)\right\rangle + \left\langle \boldsymbol{\eta}_d^t,\, \mathbf{f}_d(\mathbf{Y}^t, \mathbf{Y}^{t-1})\right\rangle\right\} \end{aligned} \tag{4.32}$$

Thus the feature vector for the entire sequence grows to length $T(s + d)$ and the feature output for time t begins at index $[(t-1)(s+d)+1]$ in $\mathbf{f}$. For example, consider a model that includes a single feature: network density. The density of each $\mathbf{y}^t$ is computed and placed at index t in the feature vector. The resulting vector is the sequence of densities as the network evolves through time. Clearly, the longer the sequence gets, the longer its feature vector gets.

However, by leveraging the functional form of $\boldsymbol{\eta}$ in a curved exponential family we can keep the number of parameters fixed. And by choosing a flexible form for $\boldsymbol{\eta}$ we can smooth away short term variations in the data to discover long range patterns of change over time.

Note that not only does having separated features per timestep allow for time-inhomogeneity, it also allows—with properly defined transition features—for irregularly spaced observations. When observing a real-world social network it is likely that observations may not appear regularly.

Features and Parameter Constraints

The models we employ use different combinations of three features: (i) the edge value histogram, (ii) network anti-stability, and (iii) GWESP.

The edge value histogram is the simple vector of counts of how many edges take each of the v discrete values. One value (the highest) is excluded to avoid having a linear dependency among the features. For a multi-valued model, this is a generalization of the usual network density feature (which is a trivial histogram with one bin for a binary network). As mentioned above, the simple sum of Equation (3.1) is ambiguous in multi-valued networks since it loses information about the distribution of edge values. The edge value histogram preserves that information. There is one multiplicative weight parameter per bin in the edge value histogram. Thus, if the edge value histogram were the only feature in a time-homogeneous model, the model would be a simple multinomial distribution over edge values. A time-inhomogeneous version is thus a time-evolving multinomial.

Network anti-stability, $a(\mathbf{Y}^{t'}, \mathbf{Y}^t)$, is the sum of squared differences in edge values between observations:

$$a(\mathbf{Y}^{t'}, \mathbf{Y}^t) \triangleq \sum_{ij} \frac{(Y_{ij}^{t'} - Y_{ij}^t)^2}{t' - t} \tag{4.33}$$

where $t' > t$ and there is no other observed timestep between t' and t (thus implying a Markov property). Note that t' need not be $t + 1$ (and frequently is not in our evaluations) so this feature is still capable of modeling irregularly spaced observations. Dividing by $t' - t$ makes (4.33) equivalent to modeling the change in an edge's value (when all other features are held constant) as a discrete time Gaussian random walk. The parameter for anti-stability is a simple multiplicative weight, and the learned value of that weight will be inversely proportional to the negative of the variance of the Gaussian. Essentially, what the model learns is the variance of a

Gaussian random walk that describes how edges change their values. In a time-inhomogeneous model, that variance is allowed to change over time.

GWESP is as it is defined in Section 4.1.3. Our time-inhomogeneous version of GWESP allows only the multiplicative weight to vary with time, which allows the model to learn the (potentially) changing importance of transitivity to the network.

The specific constraint that we place on the time-varying multiplicative weights for each of these three features is a sigmoid with offset:

$$\eta^t_{f_k}(\theta_{w_k}, \theta_{a_k}, \theta_{b_k}, \theta_{s_k}) = \theta_{w_k} \left(\frac{1}{1 + e^{-(\theta_{a_k} + \theta_{b_k} t)}} + \theta_{s_k} \right) \tag{4.34}$$

Specifically, θ_{w_k} is the ordinary multiplicative weight for feature k. That weight is scaled by the logistic with parameters θ_{a_k} and θ_{b_k}. Since the logistic will only take values between 0 and 1, the offset parameter θ_{s_k} shifts it up or down, allowing it to cross zero. Features with a positive weight make the data more likely as they increase in value and those with a negative weight make the data less likely as they increase in value. If the learned sigmoid crosses zero at some time, it means that the model has found a point at which a feature has shifted between helpful and harmful for the network.

Note that what previously would have been one parameter, θ_{w_k}, in a time-homogeneous model is now 4 parameters in our time-inhomogeneous model. That is the cost of the increased flexibility provided, but it is fixed: the number of parameters stays the same no matter how long the data sequence is.

Any number of functions could have been chosen to model time-inhomogeneity. We chose the sigmoid for 3 reasons. First, the networks we consider are observed within bounded "episodes" for their respective populations (one academic year, one senate session). We want to see if there is a shift from one underlying regime to another, e.g. from low transitivity to high. Second, the logistic has an asymptotic bound. With 4 parameters we could have used a degree 3 polynomial, but that would grow infinitely as time increased. An asymptotic function is more plausibly extended into the future. Third, while the logistic is defined for all real values of t, in our specification t will always be positive and will be effectively bounded by some maximum T. The θ_a and θ_b parameters allow the sigmoid to be shifted left and right, so it is free to only decrease or only increase. It can also stay constant if there is no time-inhomogeneity present in the data.

Of course, a single sigmoid models only a single change. For longer observation times or data where there are multiple "periods" of observation (like several academic years), more complex models will be needed.

4.3.3 Evaluation

We test this model on two real-world social network data sets. First, a simple model applied to data from the U.S. Senate illustrates the basic advantages of a time-inhomogeneous approach. Then we apply a more complex model to the Spoken Networks data.

In both data sets we quantize continuous edge values to v discrete values. All zero values are left at zero and all non-zero values are quantized to $v - 1$ discrete points using k-means. The quantized values are then normalized so that the maximum value is 1. We also experimented with equally-spaced and equally-weighted binning schemes but found that the non-uniform binning provided by k-means produced the best model fits. For the senate data, $v = 5$ and for the conversation data $v = 10$. (Initial experiments showed that the model was robust across larger values of v Wyatt et al. (2009).)

To learn the parameters we first use pseudo-likelihood to find a starting point and then use Gibbs sampling to approximate the expectation in (1.16). Despite their non-convexity, BFGS has been successfully used for learning curved ERGMs (Hunter et al., 2008) and we use it as well.

Senate Data

The senate data comes from Fowler (2006*a*,*b*) and is the same population considered by Hanneke et al.. The data captures the cosponsorship network of senators in the 108th United States Senate. When a bill (or resolution or amendment, all referred to as "bills" here) is proposed in the U.S. Senate it must be sponsored by one senator. Additional senators may sign on as co-sponsors of the bill any time before the senate votes on the bill.

We divide the senate data into sliding windows that are 28 calendar days long with 7 calendar day offsets. If the senate is not in session for more than 28 calendar days in a row we include no measurement for that period. After such a gap, the next window starts at the soonest date the senate is in session. From each window we build an undirected cosponsorship network by adding edges between two senators if one cosponsored the other's bill during that window. The strength of the edge is the number of bills cosponsored during the window, normalized by the number of days in session within the window (which adjusts for small variations due to e.g. 3 day weekends).

Figure 4.10 shows that the network's density is clearly time-varying. We fit the simplest of our models to this data: one that includes only the edge value histogram feature. Since the edges histogram feature assumes all edges are independent, the gradient for this model can be computed exactly as can its predicted networks. The green line in Figure 4.10 shows the expected density of the network as predicted by our model over time. With $v = 5$ this model has 16 parameters.

The red lines in Figure 4.10 represent prediction from a time-homogeneous Markov chain that learns a complete $v \times v$ transition matrix (and thus has 20 parameters). Each red line shows the expected density of a separate chain run forward from every observed data point. As is to be expected, the chains quickly converge to their stationary distribution. But that distribution is not near the data: the root mean square error for the Markov chain is 87, but for the time-inhomogeneous model it is only 45.

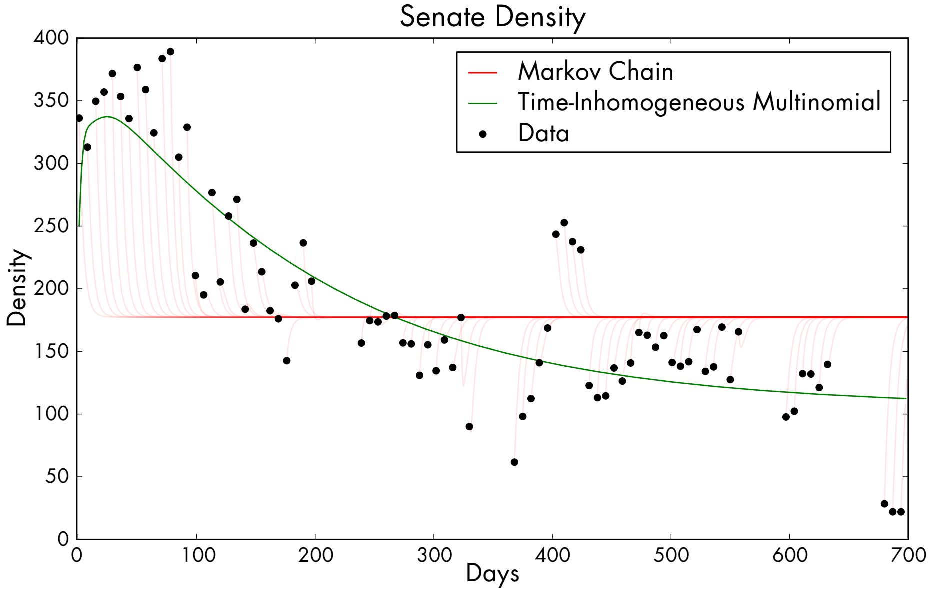


Figure 4.10: Density of senate networks with best fit values from time-homogeneous Markov chains and a time-inhomogeneous edge value model.

Spoken Networks Data

We divide the Spoken Networks data into 2 day long windows with a sliding offset of 1 day. Due to academic calendar fluctuations (and a technical issue after the 3rd week) the recording weeks do not all start at evenly spaced intervals. A network is built from each window by putting an edge between two students if they spent time in conversation during the window. The edge's weight is set to the proportion of time the pair spends in conversation, as in Section 3.5

The model we apply to this data includes all three features described above: edge value histograms, anti-stability, and GWESP. We fit both a time-inhomogeneous model that uses sigmoid constraints on the weights and a time-homogeneous model that learns the same weights for all timesteps.

Hunter et al. (2008) propose a graphical goodness-of-fit test for ERGMs that is designed to easily reveal any model degeneracy. Once a model has been fit, samples are drawn from it using MCMC. The empirical distributions of features of the samples are compared to the features of the data. This will reveal degeneracy (through, e.g., narrow or bimodal density distributions) as well as poor fits.

That "comparative samples" approach is the one we adopt for testing our models' goodness-of-fit. Specifically, after learning parameters from the data, we then simulate entire sequences of networks from learned model. For the time-inhomogeneous model we provide only the time indexes at which it should generate

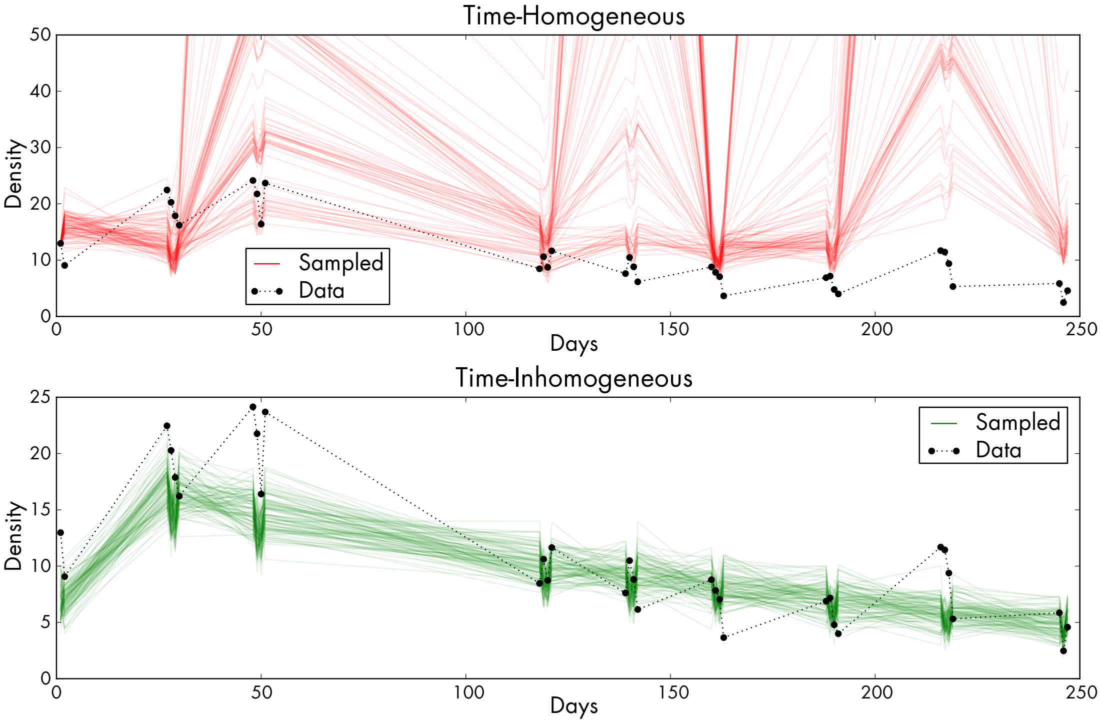


Figure 4.11: Density of sampled conversation networks compared to data.

networks. For the time-homogeneous model we provide both the time indexes for which it should generate networks, as well as the true first network observation—thus giving it potentially more information about the network series.

We use Gibbs sampling to generate sample sequences, with a burn in of 1000 sweeps over all variables and subsequent samples saved every 100 sweeps. We compare the simulated sequences to the data using more both features included in the model and features that *are not* in the model.

Figure 4.11 shows the density of the conversation networks along with the densities of networks sampled from the two models. On the top, in red, the time-homogeneous model shows a very poor fit to the data. The extreme samples that extend beyond the plot's limits show that the model is exhibiting degeneracy and assigning significant probability to completely connected graphs. In fact, if we sample from this model without conditioning it on the first observation it only returns sequences of graphs with all edges set to or near their maximum value. The time-inhomogeneous model on the bottom, in green, shows a much better fit to the data.

Figure 4.12 shows mean path lengths, computed as described in Section 3.5. Path length is a global property of the network and thus is not directly modeled by our strictly local set of features. Good reproduction of global properties is evidence of good model fit Hunter et al. (2008). Again, the time-homogeneous model (top) shows

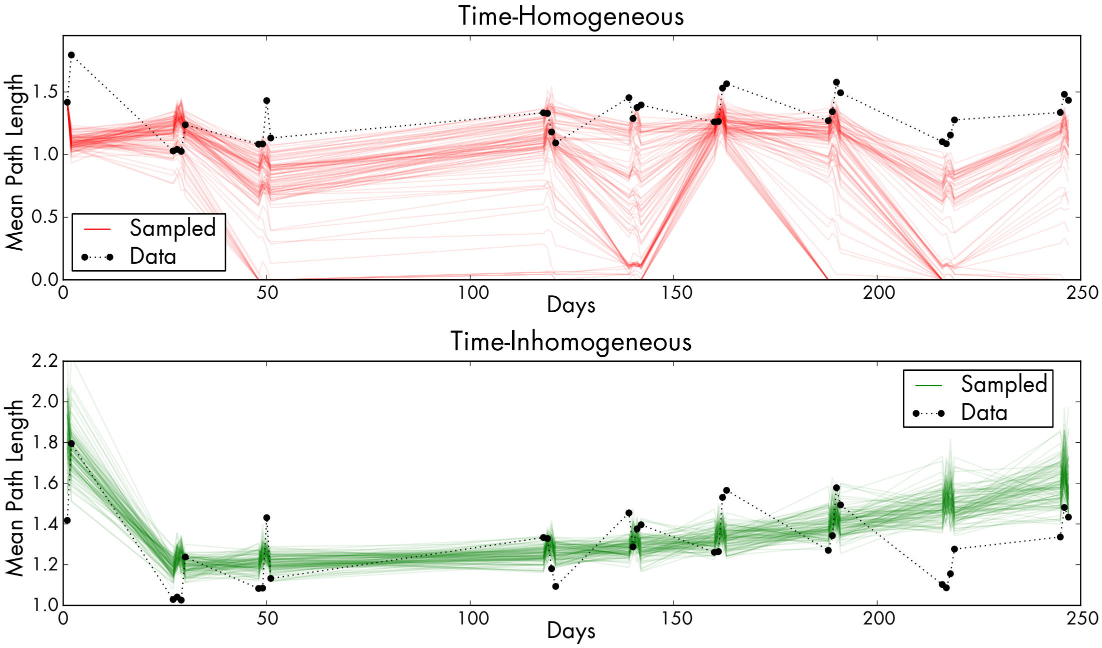


Figure 4.12: Mean path length of sampled conversation networks compared to data.

a poor fit to the data and exhibits degeneracy by generating many maximally connected networks with all paths at length zero. The time-inhomogeneous model (bottom) provides a much better fit.

What Didn't Work Before arriving at the above features we also tried two others. Simple density—the sum of all edge values—yielded networks with very low total densities and was replaced with the edge value histograms. The sum of weighted triangle values (from Equation 3.3) defined as $D(\mathbf{Y}) \triangleq \sum_{ijk}(Y_{ij}Y_{ik}Y_{jk})^{\frac{1}{3}}$ and a "poor man's GWESP" of $\log(1 + D(\mathbf{Y}))$ both lead to degeneracy.

Interpreting the Learned Parameters Since the edge histograms are simple multinomials, we can easily convert their natural parameters to mean-value parameters. Those mean-value parameters can be interpreted as the the probability that an edge takes a given value at a given time, with all other features held equal. That is, it reflects the "pure" probabilities of edge values as a function solely of time, without interference from any influence due to transitivity or stability.

Figure 4.13 shows such edge value probabilities for the conversation networks. In the beginning, strong edges (top) have larger probability than others. In the middle, weak edges become more probable, and eventually zero-valued edges (bottom) increase their dominance. This generally matches the empirical edge value distributions shown in Figure 3.8a.

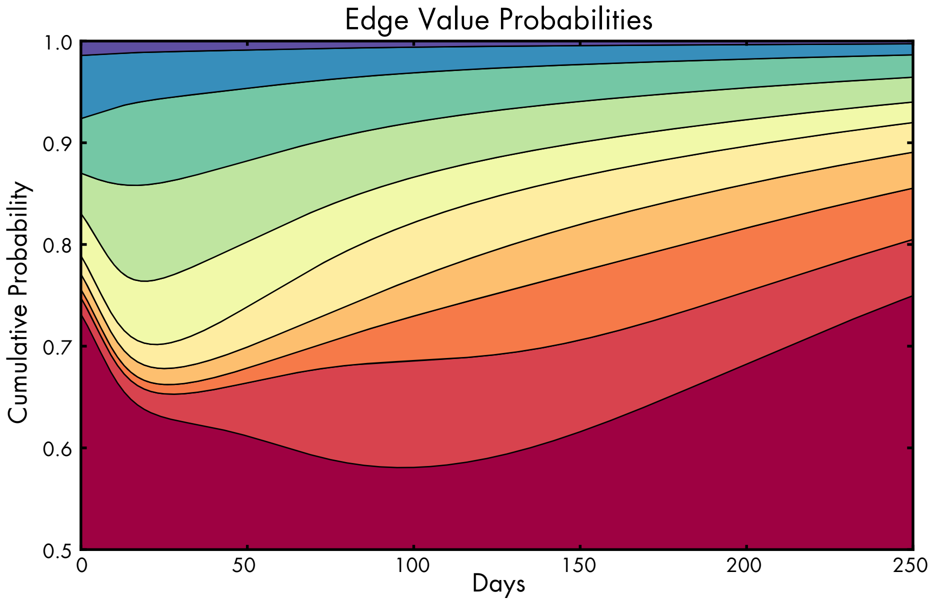


Figure 4.13: Edge value probabilities over time, with all other features kept equal. Values increase from 0 (red) at bottom to 1 (dark blue) at top. Note that y axis starts at 0.5.

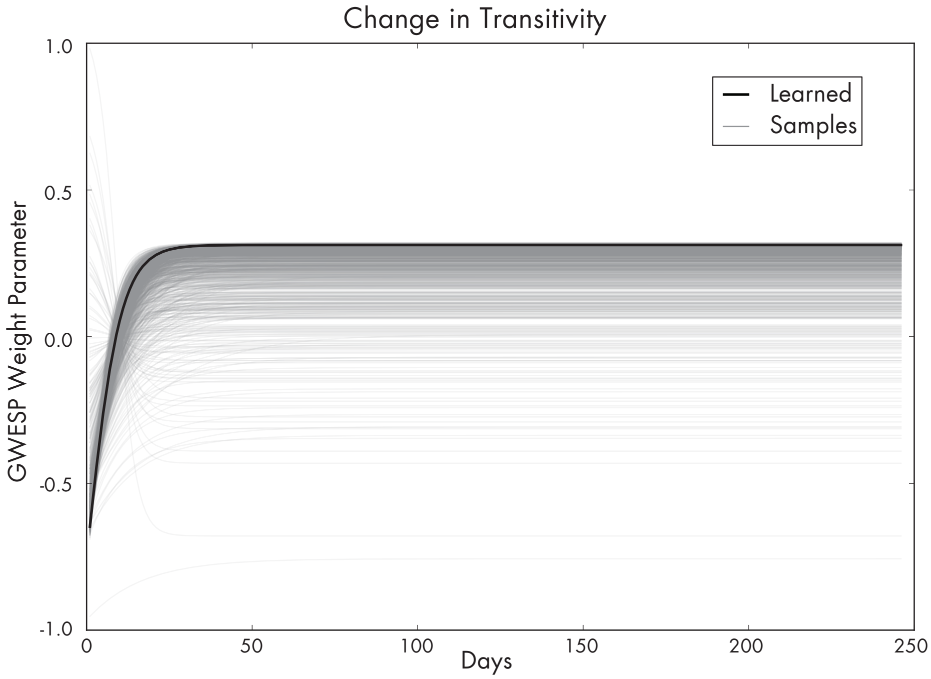


Figure 4.14: Estimated GWESP weight, with additional weights drawn from its nominal sampling distribution.

Table 4.3: Sigmoid parameter values learned for GWESP weight.

Parameter	Value	s.e
w	-2.24	1.17
a	-0.09	0.95
b	-0.19	0.05
s	-0.14	0.07

Of course, there is the usual uncertainty associated with the single set of parameters learned by the model. Fortunately, curved exponential families still allow for the Fisher information to be used to estimate standard errors around learned parameter values Hunter and Handcock (2006). Table 4.3 shows the parameter values (from (4.34)) learned for the GWESP weight in the conversation data, along with their nominal standard errors. Unfortunately, as $\boldsymbol{\eta}$ becomes more complex, standard errors around a learned $\hat{\boldsymbol{\theta}}$ become harder to reason about. We can get a coarse feel for the uncertainty, though, by sampling new parameter values from the nominal asymptotic normal sampling distribution of $\hat{\boldsymbol{\theta}}$. We can then feed those sampled parameters through $\boldsymbol{\eta}$ to see how different $\hat{\boldsymbol{\theta}}$ values might effect our interpretation of the model. Figure 4.14 shows such samples for the learned GWESP weight. The MLE (the output of (4.34) for the values in Table 4.3) is in solid black and the gray lines are weights computed from sampled values of $\hat{\boldsymbol{\theta}}$. The sampled values follow the general form of the point estimate and all suggest that in this network transitivity quickly increases in importance and then stays important. That is in contrast to the simple transitivity metrics considered in Section 3.5.3 that declined over time. That difference is exactly the reason for using an ERGM: it can separate overlapping influences (in this case, density, stability, and transitivity) and show that transitivity is actually increasing or stable through time.

4.4 Future Work

The ERGMs presented in this chapter appear to work well: they recover sensible properties of the data and they achieve comparatively good model fits. But there is always opportunity for improvement, and the most obvious area in which these models could be improved—especially when applied to the Spoken Networks corpus—is their treatment of missing data.

Recall that the size of the eligible study cohort was 27 people, though only 24 chose to participate. Those 3 missing people entail 75 completely missing entries in any undirected adjacency matrix constructed from our data. In addition to that, using just the proportion of time spent in conversation as our single observation makes the strong assumption that data is missing at random. For the Spoken Networks data, that assumption would mean that the times each person chose to record or not record are independent both of others' recording times and of the structure of their social network. That is not a realistic assumption. Subjects with social relationships did remind each other to wear their devices, and many subjects informally reported intentionally wearing their device when they knew they would be most active socially. This means that the simple proportion probably overestimates the true amount of time spent in conversation.

To compensate for this, we could model the "process of missing-ness" directly in an ERGM. Gile and Handcock (2007) presents exactly such a method for a binary network with binary missing-ness: edges are either completely observed or completely unobserved. In our data, neither the network nor the missing-ness is binary. Some dyads may be well observed (i.e. record a large amount of overlapping data), and some me be less well observed. For the less well observed dyads, there can be two sources of missing data: (i) one or both members may simply not record much data, or (ii) they may both record a fair amount of data but simply not be recording at the same times. It is in the latter that the data collection process—the decision of when to wear the recording device—contains information about the social network: if two people simply keep different hours, then the probability of an interaction between them should be low. Extensions to the methods of Gile and Handcock to handle this new kind of missing data are an interesting direction for future work.

Another related area is the treatment of low-level measurement error. As mentioned above, Butts (2003) presents a latent ERGM that explicitly models the error of survey respondents. In Butts's model (like that of Gile and Handcock) both the latent and observed networks are assumed to be binary. We made initial attempts to apply Butts's approach in our model by adding a new "layer" of latent variables between the binary, hidden social network and the observable times spent recording and in conversation. This middle layer was intended to model the true proportion of time spent in conversation, and the relationship between it and the observable data was to be an explicit model of measurement error. In early experiments with synthetic data, it proved far too difficult to fit a model with so many latent variables and the approach was abandoned.

Chapter 5

Influence in Social Behavior

It has long been known that people change their speech to *accomodate* their conversation partners (Giles et al., 1991). That accommodation has been observed in many different aspects of speech, including word choice (Brennan and Clark, 1996), turn and pause length (Cappella and Planalp, 1981), accent (Giles, 1973), rate (Gregory and Hoyt, 1982), and pitch (Gregory et al., 1997). The broad phenomenon of accommodation can be divided into two separate categories that denote the "direction" of the accommodation. *Convergence* occurs when a person makes her speech more like that of her partner; *divergence* occurs when she moves away from her partner's speaking style(Giles et al., 1977).

Interestingly, whether and to what degree a person converges on his partner's speech has been found to correlate with the partner's social status (Gregory and Webster, 1996). This suggests that simply observing how a person changes his speech may reveal his perceived "importance" of his conversational partner. In social network analysis, an abstract notion of a person's importance to the network is often quantified through measures of *centrality* (Wasserman and Faust, 1994). A natural question to consider, then, is whether there is a relationship between a person's change in speaking style and the network centrality of his partner.

In the social network literature, a change in behavior that results from an interaction with someone else is known as *influence* (Robins et al., 2001). That change is usually considered at a larger, "macro" scale (e.g. voting for a certain party, or starting to smoke) than the small, "micro" scale of changes in speech. Now that we have situated speech data, it is possible to consider influence at the scale of speech accommodation. We will use the term *influence* here—instead of accommodation—because it emphasizes the receiver (the person being accommodated) more than the sender and we will be comparing the receiver's influence to her importance. But we are referring to the same phenomenon: one person changing his speaking style based on the identity

Parts of this chapter were previously published in (Wyatt, Choudhury, Bilmes and Kitts, 2008).

of her conversational partners.

This chapter will address the relationship between influence and network centrality using two methods: a simple descriptive approach (Section 5.2.1), and a model-based approach (Section 5.2.2). Each of those has precedents in the existing literature on social behavior modeling, and those earlier models are summarized first (Section 5.1)

5.1 Previous Social Behavior Models

This section provides background on the two earlier efforts at group social behavior modeling that are the ancestors of the techniques presented later. Interestingly, though developed completely independently, both of these techniques use mixture models to discover patterns of influence and change in behavior.

5.1.1 Mixed-Memory Influence Model

Recall from Chapter 3, that Choudhury (2004) instrumented 23 people with a sociometer for 2 weeks in order to measure their network of face-to-face conversations. From the audio data that was collected she can automatically extract two-person conversations and infer, at a rate of 64 Hz, who was speaking when in the conversations. Periods with no speaker are assigned to whomever spoke last, so any conversation can be reduced to the binary sequence of turn indicators $\mathbf{b}$.

From these turn-taking sequences Choudhury and Basu (2004) estimate the 2×2 personal turn transition matrix $\mathbf{P}^i$ that encodes person i's turn-taking preferences. For simplicity, assume that person i's turns are always indicated with 1 in the binary turn taking sequence for all of her conversations, and that all of those sequences are concatenated into one long sequence $\mathbf{b}^i$ of length T. $\mathbf{P}^i$ is constructed so that

$$P^i_{11} = \frac{1}{T-1}\sum_{t=2}^{T} b^i_t b^i_{t-1} \qquad\qquad P^i_{12} = \frac{1}{T-1}\sum_{t=2}^{T}(1-b^i_t)b^i_{t-1}$$

$$P^i_{21} = \frac{1}{T-1}\sum_{t=2}^{T} b^i_t(1-b^i_{t-1}) \qquad P^i_{22} = \frac{1}{T-1}\sum_{t=2}^{T}(1-b^i_t)(1-b^i_{t-1})$$

In other words:

$$\mathbf{P}^i = \left(\begin{array}{c|c} p(i \text{ keeps the turn} \mid i \text{ currently has the turn}) & p(i \text{ relinquishes the turn} \mid i \text{ currently has the turn}) \\ \hline p(i \text{ takes the turn} \mid i \text{ does not have the turn}) & p(i \text{ still yields the turn} \mid i \text{ does not have the turn}) \end{array}\right)$$

Note that this matrix is only equivalent to the conversation turn transition matrix $\mathbf{T}$ defined in Equation 2.13 under the two simplifying assumptions above: (i) only conversations with two participants are considered, and (ii) periods of silence are counted as belonging to the last speakers turn. In that case, there is always a "speaker" and the conversation turn transition matrix will never need to count transitions to silence.

Choudhury and Basu hypothesize that a person's observed turn-taking behavior is a mixture of her own preferences and the preferences of her conversation partner. That hypothesis can be tested by modeling the turn-taking streams of the conversation as a mixed memory Markov process (Saul and Jordan, 1999).

A mixed memory Markov process reduces the number of parameters required to model a Markov chain by assuming that the transition kernel for the chain is a mixture of simpler kernels. In Choudhury and Basu's implementation, the binary turn-taking sequence for a conversation between a pair i and j is modeled with a first-order Markov chain. A complete model for the entire population would thus require learning $\binom{N}{2}$ separate transition kernels needing $2\binom{N}{2}$ total parameters. Instead of learning a separate transition kernel for each pair, Choudhury and Basu instead model the transition for a pair i and j as a mixture of their personal turn-taking preferences

$$p(\mathcal{B}_t^{ij} = x | \mathcal{B}_{t-1}^{ij} = y) = M_{ij} P_{xy}^i + (1 - M_{ij}) P_{r(xy)}^j \tag{5.1}$$

$\mathcal{B}^{ij}$ is the turn-taking sequence of a conversation between i and j. r is a function that maps indexes into $\mathbf{P}^i$ to complementary indexes into $\mathbf{P}^j$ so that the mixing is between appropriately matched transition probabilities. For example, the transition P_{11}^i—i keeps speaking—should be mixed with P_{22}^j: j stays silent. M_{ij} is the mixture coefficient for i and j: the probability that i's turn-taking preferences are honored. This mixed memory model reduces the number of parameters to $\binom{N}{2} + 2N$.

That mixture coefficient can be interpreted as the amount of influence that i has over j. If $M_{ij} > 0.5$ then i's conversations with j tend to proceed more according to her preferred turn-taking style than j's. Actual interpretation of the coefficients requires adjusting for absolute differences in i and j's styles. If they are already very similar, a large coefficient is not so meaningful. Conversely, if they are very different to begin with, then a small coefficient could still be meaningful. Choudhury and Basu accomplish this by scaling M_{ij} by the Jensen-Shannon divergence between $\mathbf{P}^i$ and $\mathbf{P}_{r(.)}^j$

To fit their mixture model to their real-world conversation data, Choudhury and Basu do not estimate a unique $\mathbf{P}^i$ for each person but instead approximate $\mathbf{P}^i$ with $\mathbf{P}^{i\backslash j}$ for learning M_{ij}. $\mathbf{P}^{i\backslash j}$ is computed identically to $\mathbf{P}^i$ but without using any data from conversations between i and j. This is done for all pairs of people, and expectation maximization is used to learn values for all components of (the upper triangle of) $\mathbf{M}$.

Choudhury and Basu find that their mixture model fits the data better than the complete model with separate transition matrices for each pair of people. Additionally, they can compute a person's aggregate influence as the mean of her scaled influence coefficients. By constructing a network with edges between pairs that have at least one conversation, they can also compute each person's betweenness centrality. They find there is a positive correlation between a person's aggregate influence and her betweenness centrality.

5.1.2 Author-Recipient-Topic Model

Another mixture model used to study social behavior data is the author-recipient-topic, or ART, model of McCallum et al. (2007). The ART model is an extension of textual topic models to include the identities of the people generating and receiving the text. Topic models, like probabilistic latent semantic analysis (Hofmann, 2001) and latent Dirichlet allocation (or LDA, Blei et al., 2003), model a topic in a textual corpus as a multinomial distribution over words. Each document has its own mixture of those word multinomials with the mixture weights reflecting the prevalence of topics in the document. In LDA, the parameters for a document's multinomial over topics are themselves latent variables governed by a Dirichlet prior, hence the name latent Dirichlet allocation. With W unique words in the corpus and $Z \ll W$ unique topics, documents can then be summarized and compared through their distributions over topics. And latent topics can be discovered by examining the "unmixed" word distributions that are learned.

The ART model extends LDA to handle email data by considering not only the words within the emails but also the author and recipients of the email. In the ART model the latent multinomial distribution over topics is no longer associated with the document (an email) but instead with an author-recipient pair.

A directed graphical model representation of the ART model is in Figure 5.1 (rectangles indicate repeated copies of exchangeable variables). For an entire corpus the "generative narrative" of the ART model is

$$\boldsymbol{\theta} \sim \text{Dir}(\boldsymbol{\alpha}) \qquad \text{Draw specific mixtures of topics for all pairs } (a, r) \tag{5.2}$$

$$\boldsymbol{\phi} \sim \text{Dir}(\boldsymbol{\beta}) \qquad \text{Draw specific mixtures of words for all topics } t \tag{5.3}$$

Then, for each email d, for each word w in d

$$r \sim \text{Uniform}(\mathbf{p}) \qquad \text{Draw a single recipient for this word} \tag{5.4}$$

$$t \sim \text{Mult}(\boldsymbol{\theta}_{ar}) \qquad \text{Draw a single topic for this word} \tag{5.5}$$

$$w \sim \text{Mult}(\boldsymbol{\phi}_t) \qquad \text{Draw the word from the topic distribution} \tag{5.6}$$

The posterior distribution of unknown quantities $p(\boldsymbol{\theta}, \boldsymbol{\phi}, t, r | a, \mathbf{p}, w, \boldsymbol{\alpha}, \boldsymbol{\beta})$ can be sampled using Gibbs sampling. Additionally, the conjugate form of the Dirichlet prior makes it easy to marginalize out $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ and sample directly from $p(t, r | a, \mathbf{p}, w, \boldsymbol{\alpha}, \boldsymbol{\beta})$—a process known as *collapsed Gibbs sampling* since the latent parameters have been "collapsed" in the graphical model (Griffiths and Steyvers, 2004). McCallum et al. use collapsed Gibbs sampling to repeatedly sample t and r and aggregate those samples into simple counts of how

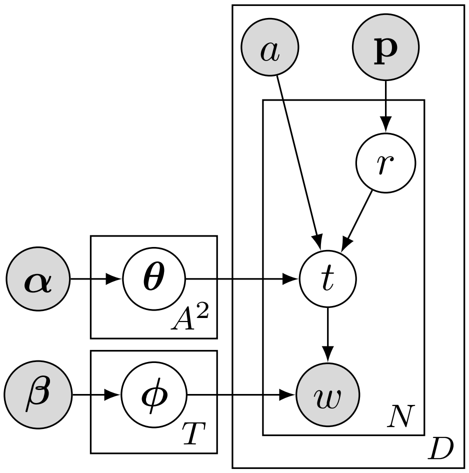


Figure 5.1: The author-recipient-topic model. a: the author; $\mathbf{p}$: the set of recipients; w: one of N words in an email. r: the single recipient of word w. t: a latent topic (multinomial over words) chosen from $\boldsymbol{\theta}_{ar}$ a mixture of multinomials specific to a and r. $\boldsymbol{\phi}_t$ are the parameters of the word multinomial for topic t. $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ are fit over a corpus of D separate emails. $\boldsymbol{\alpha}$ and β are hyperparameters governing the prior distributions of $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$.

often each word is assigned to a given topic and recipient. Those counts are assembled into a four-dimensional array $\mathbf{M}$ where M_{artw} is a count of how often word w was assigned to topic t and recipient r in emails from author a.

Mean-value posterior point estimates of $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\phi}}$ can be computed from $\mathbf{M}$ as

$$\hat{\theta}_{art} = \frac{\sum_w \alpha_t + M_{artw}}{\sum_{t',w} \alpha_{t'} + M_{art'w}} \qquad \text{probability of topic } t \text{ between } a \text{ and } r \tag{5.7}$$

$$\hat{\phi}_{tw} = \frac{\sum_{a,r} \beta_w + M_{artw}}{\sum_{a,r,w'} \beta_{w'} + M_{artw'}} \qquad \text{probability of word } w \text{ in topic } t \tag{5.8}$$

Of course those estimates will have lost any information about the posterior distribution of $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\phi}}$ other than their means, but that is a cost of collapsed sampling.

McCallum et al. also derive an author's marginal topic distribution as

$$p(t|a) = \frac{\sum_{r,w} \alpha_t + M_{artw}}{\sum_{r,t',w} \alpha_{t'} + M_{art'w}} \qquad \text{probability of topic } t \text{ from author } a \tag{5.9}$$

These author-topic distributions can be used to compare and group people according to the similarity of emails they send.

An aside McCallum et al. do not explore them, but other marginalizations of $\mathbf{M}$ are possible. Summing over a instead of r in (5.9) would give the analogous recipient-topic distribution and provide another way of comparing people. Interestingly, computing

$$p(r|a) = \frac{\sum_{t,w} M_{artw}}{\sum_{r',t,w} M_{ar'tw}} \qquad \text{the probability of recipient } r \text{ for author } a \tag{5.10}$$

would provide the proportion of all words written by a that were "intended" for recipient r.

A problem with network data derived from email is the ease with which one person can send a single message to numerous others. Constructing a network from emails usually entails placing edges between the and author and all recipients, and those edges usually all have the same weight. That practice is questionable, especially for emails with large numbers of recipients. Some researchers define simple thresholds for numbers of recipients above which the email is not considered a significant interpersonal communication (Kossinets and Watts, 2006). Equation (5.10) provides a mechanism for un-mixing how much of an email is intended for each recipient. That in turn provides a way to assign strengths to ties when constructing a network.

5.2 Influence in the Spoken Networks Data

We have used two approaches to examine influence in the Spoken Networks data. Section 5.2.1 describes a method that estimates influence from simple descriptive statistics and compares those influence estimates to network centrality. Section 5.2.2 describes a model-based approach to estimating influence that overcomes some of the short-comings of the descriptive approach.

5.2.1 Descriptive Influence Metrics

Perhaps the most immediate method for measuring change in behavior is to simply compute the distance between a person's "usual" behavior and his behavior when speaking with a specific conversational partner. We consider two measures of speaking behavior: rate and pitch (computed as described in Section 2.3.1). Let b_t^{ij} be the value computed for one of these behaviors for i while in conversation with j at time t.

It is easy to estimate $\hat{b}^{ij}$ the mean of i's behavior when speaking with person j. Additionally, after Choudhury and Basu, for comparisons to person j we estimate i's "usual" behavior as

$$\hat{b}^{i\setminus j} \triangleq \frac{1}{Z} \sum_{k \neq j} \sum_{t} b_t^{ik} \qquad i\text{'s average behavior with everyone except } j \tag{5.11}$$

Finally, to account for i's usual variation in behavior we also estimate $\hat{\sigma}^i$, the standard deviation of i's behavior for all partners (including j). We then define the change in i's behavior with j as

$$d^{ij} \triangleq \frac{|\hat{b}^{ij} - \hat{b}^{i\setminus j}|}{\hat{\sigma}^i} \tag{5.12}$$

(5.12) is the univariate Mahalanobis distance but it is also easily interpretable as the number of sample standard deviations that i changes his behavior when speaking to j. This simple change metric measures only raw accommodation or influence. It does not—due to the absolute value—distinguish between convergence and divergence.

To obtain a single measure of i's influence over her conversational partners, we compute her *incoming change* as

$$f^i \triangleq \frac{1}{P} \sum_j d^{ji} \tag{5.13}$$

where P is the number of conversation partners i has in the data.

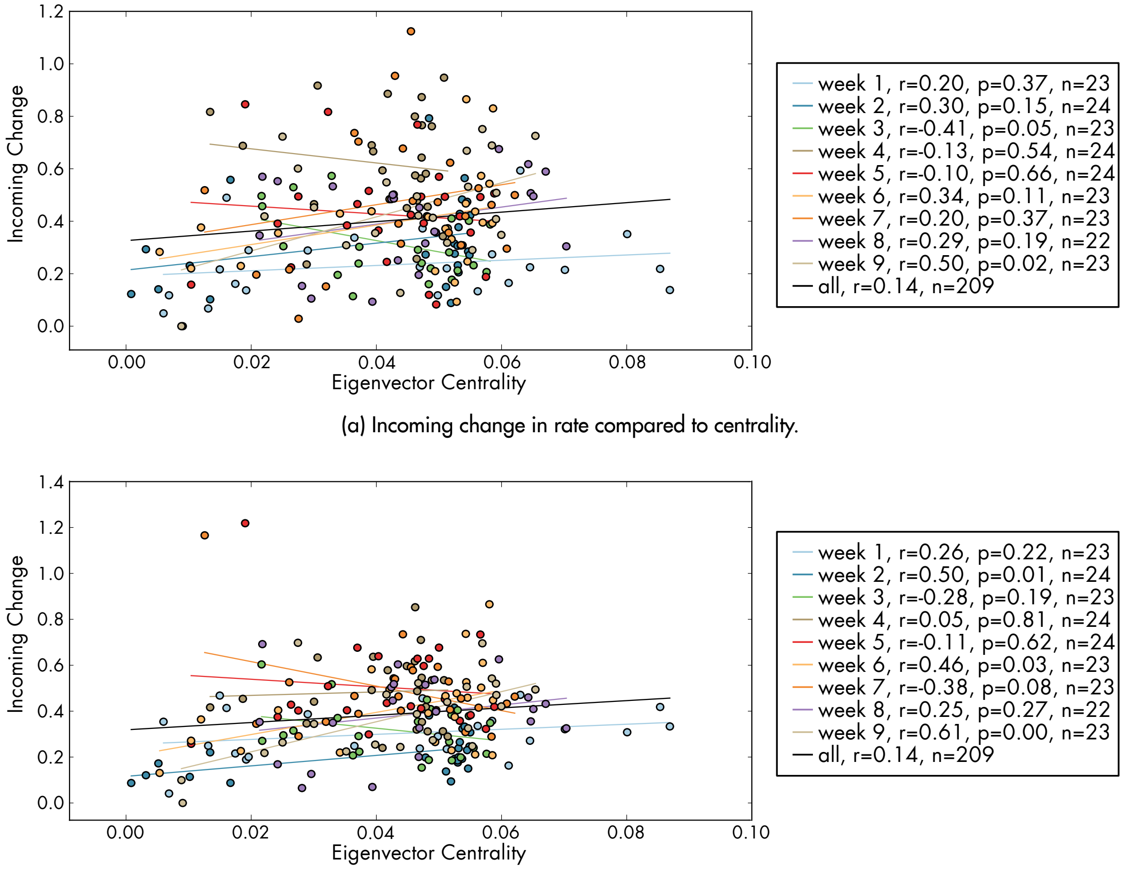
Choudhury (2004) only extracted dyadic conversations from her data, but the new methods we presented in Chapter 2 are capable of finding multi-party conversations. In multi-party conversations there could be more than one source of influence and the simple metrics above cannot untangle such overlapping influences. To avoid potential confusion caused by multiple influences we compute the above quantities using only dyadic conversations.

Having computed the incoming change for each person in our corpus, we compare it to that person's *eigenvector centrality* (Wasserman and Faust, 1994). Person i's eigenvector centrality c_i is the value of the i-th component in the principal eigenvector of $\mathbf{Y}'$, the network's adjacency matrix with rows normalized to sum to 1. Intuitively, this expresses centrality recursively: a person's centrality is a linear combination of the centralities of the others to whom she is connected. The eigendecomposition of $\mathbf{Y}'$ solves this system of equations to find each person's centrality. To compute centrality, we use the weighted conversation network defined in Section 3.5.4 (constructed from all conversations, not just dyadic ones).

Figure 5.2 shows the comparison of centralities (x axes) to incoming changes (y axes) in rate (top) and pitch (bottom) for networks computed from each week of the Spoken Networks data. Correlation coefficients for each week and their p-values are shown in the legend. Positive correlation coefficients would agree with the earlier result of Choudhury and Basu (2004): people change their behavior more when conversing with more central partners. Unfortunately, the results are inconclusive. A few weeks have negative correlations, and not all correlations are significant. The aggregate correlations are positive, but traditional significance tests cannot be performed because the observations are not independent (an aspect discussed below in Section 5.3).

5.2.2 General Influence Model

The descriptive influence metrics described above are coarse in many ways. First, they must be constrained to dyadic conversations, potentially discarding much useful data. Second, while they assume the existence of a single personal behavior, measures of that individual behavior must be made using data from all partners except one. That assumes that multiple influence are independent, and the mean of i's behavior with different



(a) Incoming change in rate compared to centrality.

(b) Incoming change in pitch compared to centrality.

Figure 5.2: Descriptive change in behavior compared to eigenvector centrality. Each point is a person, with measurements from separate weeks shown in different colors. Best fit linear regressions are shown as solid lines, with the black line representing the regression to all points from all weeks.

partners will be representative of her "true" individual behavior. If a person has few conversation partners, then that mean will have high variance. Imagine the extreme situation where i converses only with j and k, in that case $\hat{b}^{i\backslash j} = \hat{b}^{ik}$ and i's individual behavior measurement will always be "contaminated" with influence from either j or k.

A better approach is to explicitly model each person's individual behavior distribution. Of course, since social behavior always requires more than one person, there is no way to ever observe an individual's true distribution. Rather, it must be inferred from observations with multiple partners. By jointly modeling both these latent individual behavior distributions and the process of influence that causes a person to change his behavior, we may be able to extract more useful information from the data. The joint model that we consider here is a generalization of the mixed-memory influence model of Choudhury and Basu (2004) inspired by the ART model of McCallum et al. (2007).

Recall that Choudhury and Basu (2004) transformed the turn-taking signal from every two person conversation into a binary sequence and that each person's turn-taking preferences were modeled with a 2×2 transition matrix $\mathbf{P}^i$. That transition matrix has only two free parameters and those are interpretable as the single parameters of two separate geometric distributions: one for how long i prefers her turns to be, and the other for how long she prefers her partner's turns to be. Viewed that way, the ordering of turns in a conversation does not matter: it is simply a "bag-of-turns" with varying lengths. In that case, the mixture model of Choudhury and Basu can be rewritten in a form very similar to the ART model.

Figure 5.3 shows this new form. Call this the general influence model. There are C "interaction events" being modeled. Those could be conversations, email messages, phone calls, meetings, discussion threads in social media, etc. Each interaction event has participants $\mathbf{p}$ and is broken into T turns. During a turn only person s exhibits any behavior. Within a turn there are B *instantaneous behaviors*. Those may be observations of pitch or rate, or the length of the turn (in which case $B = 1$), or even words.

The basic mixture assumption that the model makes is that each person has two personal distributions over behaviors: (i) a sender distribution governing the behavior she exhibits during an interaction, and (ii) a receiver distribution governing the behavior her "ideal" partner would exhibit during interactions with her. When a person interacts with others, her observed behavior is a mixture of her sender distribution and the receiver distributions of the *other* participants in the interaction.

The generative narrative for the model is:

For each instantaneous behavior b in each turn

$$r \sim \text{Uniform}(\mathbf{p}) \qquad \text{Draw a recipient} \tag{5.14}$$

$$f \sim \text{Bernouli}(\mathbf{M}_{sr}, s, r) \qquad \text{Draw an influencer} \tag{5.15}$$

$$b \sim \text{W}(\boldsymbol{\theta}_{fa}) \qquad \text{Draw a behavior} \tag{5.16}$$

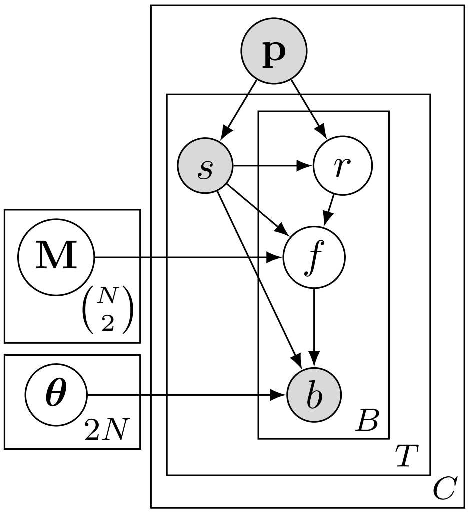


Figure 5.3: A general influence model. $\mathbf{p}$ are the participants in one of C conversations. s is the speaker in each of T turns. Turns have observable behavior b that is "intended for" person r. f is the person who determines the behavior and takes a value of either s or r, governed by the mixture coefficient M_{sr}. $\boldsymbol{\theta}$ are the parameters for each person's individual sender and receiver behavior distribution.

Bernouli(p, a, b) is a simple two category multinomial that takes value a with probability p and value b with probability $(1 - p)$. W can be any distribution parameterized by $\boldsymbol{\theta}$. There are $2N$ such distributions, one for each person's sender and receiver distribution. These distributions are chosen with the two dimensional index fa where $a = 1$ if $f = s$ and $a = 2$ if $f = r$. Note that $\mathbf{M}$ and $\boldsymbol{\theta}$ are not observed: they are to be learned from data. And while we have not drawn them, it is straightforward to add hyperparameters for priors of $\mathbf{M}$ and $\boldsymbol{\theta}$.

Altogether, this is a very flexible model for capturing influence in social behavior where it is assumed that the observed behavior is a mixture of each participant's sender and receiver behavior.

The Choudhury and Basu model can be re-written as a general influence model. For that, b are turn lengths and $\mathbf{M}$ is the mixing matrix. W is a geometric distribution. $\boldsymbol{\theta}_{i1}$ is person i's preferred turn length when speaking, and $\boldsymbol{\theta}_{i2}$ is i's preferred length for her partners' turns. The advantage of this formulation is that it can model multi-person conversations through the use of the latent recipient r.

A model similar to a factored version of the ART model can also be re-written as a general influence model. For that, b are words in an email. W is a multinomial and the sender and receiver distributions are author and recipient distributions over words.[1] Of course, that factorization comes at the price of losing any unique aspects

[1] As written, the model has no topics, but they could be introduced by putting a latent topic t in b's place in the graphical model, making b a child of t and adding topic parameters ϕ as parents of b.

of a distribution specific to a single author-recipient pair. For example, if a person sends very different emails to close friends than he does to colleagues those two would be blurred into a single sender distribution.

For some behaviors, it may not be necessary to make the distinction between sender and receiver distributions. In spoken conversations, for example, it is unlikely that a person has distinct distributions for rates at which she wishes to speak and rates at which she wants her partners to speak. A simpler model can use a single personal distribution for a person's preferred pitch or rate. In that case, the graphical model would omit the edge from s to b and there would be only N different $\boldsymbol{\theta}$ parameters. That is precisely the model considered in the remainder of this section.

One basic difference between the ART model shown in Figure 5.1 and the general influence model in Figure 5.3 is the lack of hyperparameters for prior distributions on $\mathbf{M}$ and $\boldsymbol{\theta}$ and the general influence model. That is because we have so far only fit this model using expectation maximization to obtain point estimates of $\mathbf{M}$ and $\boldsymbol{\theta}$, and not through e.g. Gibbs sampling to approximate their posterior distribution.

To adapt the general influence model to the pitch and rate data considered above, we take W to be a simple univariate Gaussian with the usual parameters $\boldsymbol{\theta}_i = (\mu_i, \sigma_i^2)$. After fitting that model to our data, we can compute a quantity similar to (5.12)

$$c^{ij} \triangleq \frac{M_{ij}\mu_i - (1 - M_{ij})\mu_j}{\sigma_i} \tag{5.17}$$

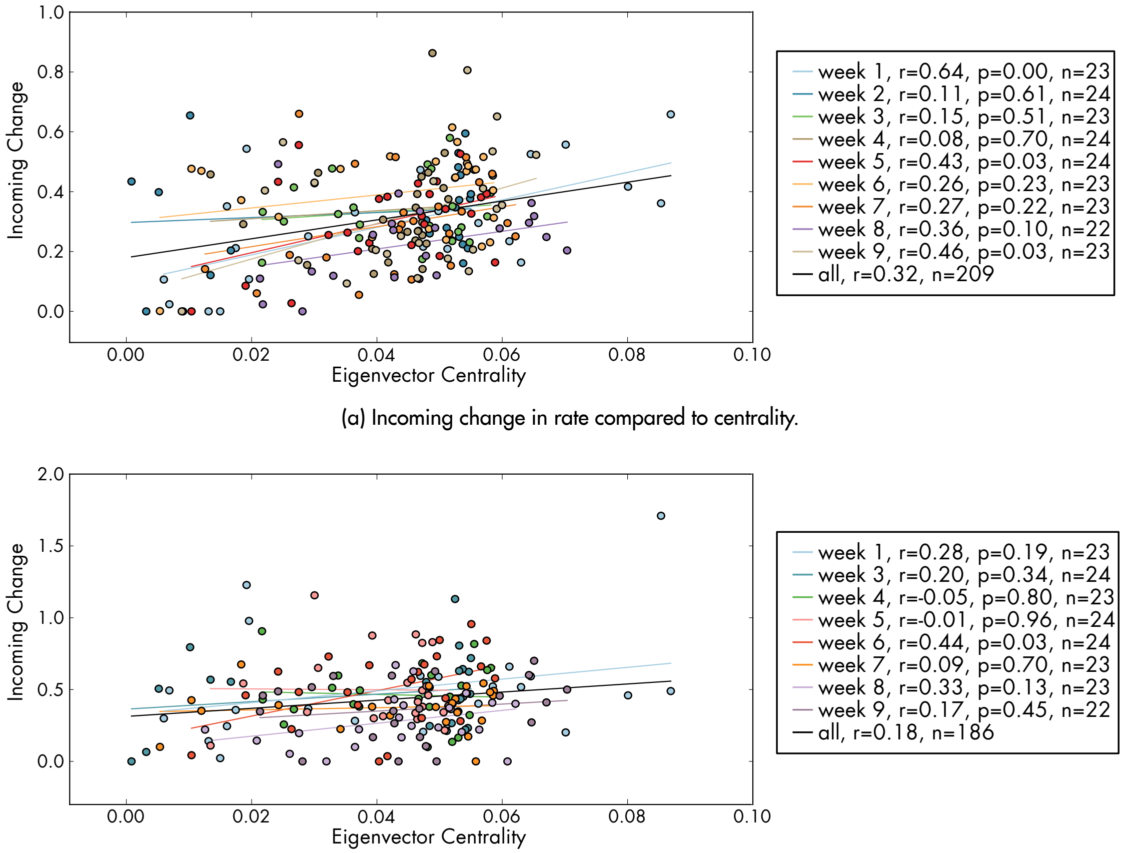
(5.17) is essentially the model-based equivalent of (5.12): how many standard deviations is i expected to change her behavior when speaking to j? Note that (5.17) can be computed even when i and j have had no one-on-one conversations—a key strength of explicitly modeling latent personal distributions. Additionally, analogous to (5.13), we can also compute the average of (5.17) across all partners to find a person's expected incoming change.

Figure 5.4 shows the comparison of centralities to expected incoming changes in rate and pitch as they are estimated using the general influence model. The results of this comparison are slightly stronger than those for the simple descriptive approach. All weeks show a positive correlation between incoming change in rate and centrality. Only two weeks show a negative correlation for pitch, and the coefficient value is very small. As above, however, not all correlations are significant, though the aggregate correlation coefficient for rate is much stronger than above.

5.3 Future Work

The work in this chapter is on-going, and there are several immediate improvement that can be made.

First, since all weeks of data come from the same population, each new week does provide more information about the possible correlation between change in behavior and network centrality. Traditional tests



(a) Incoming change in rate compared to centrality.

(b) Incoming change in pitch compared to centrality.

Figure 5.4: Model-based change in behavior compared to eigenvector centrality.

of significance cannot be done on the correlation coefficient computed for the aggregated weeks since that aggregated data set contains 9 separate observations for each subject. Obviously, those observations are not independent and any resulting confidence interval (and p-values derived from it) will be overly optimistic. A better approach might be to use a hierarchical model with a distribution over latent correlation coefficients. Learned properties of that distribution (e.g. mean and variance, if it is Gaussian) could show whether a correlation hold across all weeks.

Second, all of the approaches above ignore the temporal aspect of the data. That aspect is important for two reasons: (1) the strength of the relationship between behavior and centrality in one week may depend on the strength of the relationship in the previous weeks, and (2) influence may have a durable effect on a person's individual behavior distribution. The first reason would change how some global correlation is to be found in the entire data set. The second would change how influence is interpreted from week to week. In an extreme case, a person i may completely adopt the behavior of some j over time. The static approaches above would see that as a loss of influence: i no longer changes his behavior when speaking to j. The fact that i now speaks like j in all his interactions—and did not speak that way before—should be taken into account.

Third, like any model, the general influence model could be checked for goodness-of-fit. Data could be held out and predicted based on the fitted model. For example, we could attempt to predict how two people who have never had a one-on-one conversation will speak to each other. We could also test the likelihood of held out data, or sample synthetic data from the fitted model (as in Section 4.3.3), or use any other goodness-of-fit test.

Finally, there is the possibility that any correlation found is induced by our data processing pipeline. Consider the path of information shown in Figure 2.3. The voicing inference is used to find colocated people, which is then used to determine who could be in conversation. It is also used to disambiguate speakers during speaker segmentation, and used to decide which regions of signal from which to infer pitch. Energy is used to segment speaker turns, after which it is also used to compute rate. It will be necessary to determine how much—if any—of an observed correlation is process-induced.

Chapter 6

Conclusion and Future Work

The work presented in this dissertation has extended the state-of-the-art in measuring and modeling networks of human social behavior—specifically, networks expressed through real-world, face-to-face conversations.

We have outlined a set of a set of privacy-sensitive features that can be computed from incoming audio data in real-time. We have shown how to use those features to determine who was physically colocated with whom, both at the granularity of a room in a building and at the more elastic "acoustic proximity" needed to have a face-to-face conversation. We have used the privacy sensitive features to infer who was speaking when, and combined those inferences with colocation inference to determine who was in conversation with whom. This conversation detection can handle conversations with any number of participants, extending beyond previous methods that were limited to dyadic conversations only.

We have recounted the year long collection of privacy-sensitive situated speech data from a subject population of 24 graduate students. We applied our colocation and conversation detection methods to this data to extract a year's worth—426 person hours—of real-world face-to-face conversations within the study cohort. We have constructed weighted networks of social behavior from that data and examined basic descriptive statistics in order to compare networks of colocation events to networks of conversation events. The two are found to be very different, providing new insight into earlier studies that had access to only colocation data.

We have extended exponential random graph models so that they may more robustly handle social behavior data. A latent ERGM was proposed that simultaneously models both the structural properties of a network of abstract, hidden social ties as well as the relationship between those latent ties and their expression in noisy, observable behavior. A time-inhomogeneous ERGM was used to discover long-range trends in the evolution of underlying properties of the network. The time-inhomogeneous model was also found to provide much better fits to real data than standard time-homogeneous models, which displayed symptoms of model degeneracy.

Finally, we have examined two methods for discerning influence in social behavior data and comparisons between the measured influence and network centrality. A simple difference between average behaviors provided inconclusive results, but a general influence mixture model yielded much stronger evidence of a positive correlation between how much a person changes her behavior and the eigenvector centrality of her conversational partner. Additionally, this general influence model can be easily extended to any kind of social behavior data—emails, phone calls, social media posts—and is thus capable of discerning influence in a broad class of observations.

6.1 Future Work

The methods and results presented in this work are only very small steps in directions to be explored as social behavior modeling becomes more commonplace.

Real-time Conversation Detection All of the low-level speech processing techniques presented in Chapter 2 could theoretically run on any modern cell phone. Additionally, even though the colocation results presented in Chapter 2 use voicing posteriors computed from 50 minutes of data, we have empirically determined that fixed lag smoothing with a lag of 916 ms is enough to yield identical posterior distributions. If devices could compute their own voicing posteriors, they could share them with one another and infer whether their wearers are colocated. If the devices share their observed mean energies along with their voicing posteriors, then speaker segmentation (with independent pairwise speaker segmentations distributed across devices) and conversation detection could also be performed in real-time. Those inferences could then be used by applications that need to know their users' immediate social context.

Improved Measurement Methods As described in Sections 1.1.2 and 4.4, there are sources of measurement error that could be incorporated into our models. In addition to modeling error, there are ways that new measurements can be incorporated to improve the quality of the resulting data. Choudhury (2004) asked her participants to label some of their own conversations in her data. That, of course, requires saving raw audio which does involve sacrificing privacy (both the subjects' and others') for data quality. An alternative—particularly if combined with real-time conversation inference—would be to use experience sampling (Larson and Csikszentmihalyi, 1987) to occasionally prompt users to confirm or correct the automated inferences about their conversations.

Joint Models of Low-Level Behavior and High-Level Networks Current research (both that presented here and others') is generally split between models that treat the network as a random variable but use very little of the fine-grained behavior data (i.e. only proportion of time spent in conversation), and models that use much of the behavior data while treating the network as a non-random constant (the observed $\mathbf{p}$ in Figure 5.3). Joining these two approaches would realize much more of the power of social behavior data. Joint models could learn how specific low-level behaviors effect the probability of ties forming or dissolving and use that to better predict future networks.

Spatial Inhomogeneity Analogous to the time-inhomogeneity exhibited in longitudinal data, there may be "spatial inhomogeneity" exhibited in very large networks. Local portions of the network may display homogeneous properties, but those properties may change as one moves to different parts of the network. Indeed, modeling spatial inhomogeneity may lead to new ways to discover sub-networks corresponding to communities, such as a *de facto* working division within a company or an extended social circle within a school. Additionally, it may be more profitable (and tractable!) to model influence within these sub-networks since they may correspond to practical extents of influence in the larger network.

An Automated Sociologist's Assistant Currently, new ERGM features are discovered through careful sociological reasoning followed by trial and error—and still most features do not yield good model fits. New research into feature discovery in statistical relational machine learning (like Markov logic) shows promise, but has not been extended beyond small domains of simple, binary data. Methods for the automated discovery of relevant social features from complex low-level behavioral data could lead to an "automated sociologist's assistant" capable of mining novel patterns from social behavior data and rapidly advancing the frontiers of computational social science.

Intelligent Social Systems A relatively unexplored side of the increasing availability of social behavior data is that it is due to introduction of computers into communications media. In the past it was theoretically possible—though perhaps prohibitively laborious and privacy-invasive—to record telephone calls, or keep records of physical mail sent. Such hypothetical data collection methods could record social behavior data—but they could only record it. With computers as components of a communication system, it is now possible not just to record data, but to compute on that data: to draw inferences, and possibly even to react or intervene.[1]

It is this capacity for automated reaction that could lead to the development of *intelligent social systems*: software that understands and engages with the social behavior of its users. An intelligent social system will need to understand how its users' social behaviors reflect the nuances of their interpersonal relationships, how the users fit together into a larger social network (not all of which may be observable), and how best to assist people with their interactions. Such a system could prioritize communications based on knowledge about both

[1] This observation comes from Haym Hirsh.

the relationship between the sender and receiver, as well as the benefit the communication may provide to the larger network.

Such a system could also leverage repetitions in interaction patterns that are observed across many different communities. For example, people may fall into certain roles within a group (e.g. leader, mediator, critic) and configurations of these roles might repeat across disparate groups. The system could learn these patterns, possibly enabling it to better predict how behavior might change (to fit a potential role for example) or how ties may dissolve (to remove a detrimental structure). The system could learn how interaction patterns are associated with the ultimate success of a group. That could allow for the early detection of dysfunctional groups, and provide an opportunity for appropriate intervention—automated or otherwise. The system could also learn to route information through the person occupying the role best suited to spreading the information—a distinction revealed by both network structure *and* behavior.

We are a long way from such systems, but continuing advances in our ability to measure and model human social behavior will enable steady progress to be made towards intelligent social systems.

Bibliography

Ajmera, J., Lathoud, G. and McCowan, I. (2004), Clustering and segmenting speakers and their locations in meetings, *in* 'Proc. of ICASSP'.

Anderson, C. J., Wasserman, S. and Crouch, B. (1999), 'A $p*$ primer: logit models for social networks', *Social Networks* **21**, 37–66.

Ang, J. (2002), Prosody-based automatic detection of annoyance and frustration in human-computer dialog, *in* 'Proc. of ICSLP'.

Anguera, X. (2006), Robust Speaker Diarization for Meetings, PhD thesis, Universitat Politècnica de Catalunya.

Barndorff-Nielsen, O. (1978), *Information and Exponential Families*, Wiley.

Basu, S. (2002), Conversational Scene Analysis, PhD thesis, Massachusetts Institute of Technology.

Basu, S. (2003), A linked-HMM model for robust voicing and speech detection, *in* 'Proc. of ICASSP'.

Batliner, A., Fisher, K., Huber, R., Spilker, J. and Nöth, E. (2000), Desperately seeking emotions or: actors, wizards and human beings, *in* 'Proc. of the ISCA ITRW on Speech and Emotion'.

Baym, N., Zhang, Y. B. and Lin, M. C. (2004), 'Social interactions across media: Interpersonal communication on the internet, telephone and face-to-face', *New Media and Society* **6**, 299–318.

Bernard, H. R. and Killworth, P. D. (1977), 'Informant accuracy in social networks II', *Human Communication Research* **4**(1), 3–18.

Bernard, H. R., Killworth, P. D. and Sailer, L. (1980), 'Informant accuracy in social network data IV: a comparison of clique-level structure in behavioral and cognitive network data', *Social Networks* **2**(3), 191–218.

Bernard, H. R., Killworth, P. D. and Sailer, L. (1982), 'Informant accuracy in social network data V: An experimental attempt to predict actual communication from recall data', *Social Science Research* **11**, 30–66.

Besag, J. (1975), 'Statistical analysis of non-lattice data', *Journal of the Royal Statistical Society, Series D* **24**(3), 179–195.

Besag, J. (2000), Markov chain monte carlo for statistical inference, Technical Report 9, University of Washington, CSSS.

Bilmes, J. (2004), On soft evidence in bayesian networks, Technical Report 16, University of Washingon Department of Electrical Engineering.

Blei, D., Ng, A. and Jordan, M. (2003), 'Latent dirichlet allocation', *Journal of Machine Learning Research* **3**, 993–1022.

Borovoy, R. (2002), Folk Computing: Designing Technology to Support Face-to-Face Community Building, PhD thesis, MIT MediaLab.

Brennan, S. E. and Clark, H. H. (1996), 'Conceptual pacts and lexical choice in conversations', *Journal of Experimental Psychology* **22**(6), 1482–1493.

Brown, L. D. (1986), *Fundamentals of Statistical Exponential Families with Applications in Statistical Decision Theory*, Institute of Mathematical Statistics.

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

Campbell, N. (2002), The recording of emotional speech: JST/CREST database research, *in* 'Proc. of LREC'.

Cappella, J. N. and Planalp, S. (1981), 'Talk and silence sequences in informal conversations, III: Interspeaker influence', *Human Communication Research* **7**(2), 117–132.

Choudhury, T. (2004), Sensing and Modeling Human Networks, PhD thesis, MIT Media Lab.

Choudhury, T. and Basu, S. (2004), Modeling conversational dynamics as a mixed-memory markov process, *in* 'Proc. of NIPS'.

Choudhury, T. and Pentland, A. S. (2003), Sensing and modeling human networks using the sociometer, *in* 'Proc. of the International Conference on Wearable Computing'.

Connolly, C. I., Burns, J. B. and Bui, H. H. (2008), Recovering social networks from massive track datasets, *in* 'Proc. of the IEEE Workshop on Applications of Computer Vision'.

Corman, S. R. and Scott, C. R. (1994), 'A synchronous digital signal processing method for detecting face-to-face organizational communication behavior', *Social Networks* **16**(2), 163–179.

Davis, J. A. (1970), 'Clustering and hierarchy in interpersonal relations: Testing two graph theoretical models on 742 sociomatrices', *American Sociological Review* **35**(5), 843–851.

Dellaert, F., Polzin, T. and Waibel, A. (1996), Recognizing emotion in speech, *in* 'Proc. of ICSLP'.

Dielmann, A. and Renals, S. (2004), Multi-stream segmentation of meetings, *in* 'Proc. of the IEEE Workshop on Multimedia Signal Processing'.

Donovan, R. (1996), Trainable Speech Synthesis, PhD thesis, Cambridge University.

Douglas-Cowie, E., Campbell, N., Cowie, R. and Roach, P. (2003), 'Emotional speech: towards a new generation of databases', *Speech Communication* **40**, 33–60.

Douglas-Cowie, E., Cowie, R. and Schroeder, M. (2000), A new emotion database: considerations, sources and scope, *in* 'Proc. of the ISCA ITRW on Speech and Emotion'.

Eagle, N. and Pentland, A. S. (2006), 'Reality mining: Sensing complex social systems', *Personal and Ubiquitous Computing* **10**(4), 255–268.

Eagle, N., Pentland, A. S. and Lazer, D. (2009), 'Inferring friendship network structure by using mobile phone data', *PNAS* **106**(36), 15274–15278.

Efron, B. (1975), 'Defining the curvature of a statistical problem (with applications to second order efficiency)', *The Annals of Statistics* **3**(6), 1189–1242.

Ennett, S. T. and Bauman, K. E. (1993), 'Peer group structure and adolescent cigarette smoking: A social network analysis', *Journal of Health and Social Behavior* **34**(3), 226–236.

Ferris, B., Haehnel, D. and Fox, D. (2006), Gaussian processes for signal strength-based location estimation, *in* 'Proc. of Robotics: Science and Systems'.

Fowler, J. H. (2006*a*), 'Connecting the congress: A study of cosponsorhsip networks', *Political Analysis* **14**(4), 456–487.

Fowler, J. H. (2006*b*), 'Legislative cosponsorhsip networks in the U.S. house and senate', *Social Networks* **28**(4), 454–465.

Frank, O. and Strauss, D. (1986), 'Markov graphs', *Journal of the American Statistical Association* **81**(395), 832–842.

Freeman, L. (1992), 'Filling in the blanks: A theory of cognitive categories and the structure of social affiliation', *Social Psychology Quarterly* **55**(2), 118–127.

Freeman, L., Romney, A. K. and Freeman, S. C. (1987), 'Cognitive structure and informant accuracy', *American Anthropologist* **89**, 311–325.

Gatica-Perez, D., McCowan, I., Zhang, D. and Bengio, S. (2005), Detecting group interest-level in meetings, *in* 'Proc. of ICASSP'.

Geyer, C. J. and Thompson, E. (1992), 'Constrained monte carlo maximum likelihood for dependent data', *Journal of the Royal Statistical Society* **54**(3), 657–659.

Gibson, D. R. (2005), 'Taking turns and talking ties: Networks and conversational interaction', *American Journal of Sociology* **110**(6), 1561–97.

Gile, K. and Handcock, M. (2007), Modeling social networks with sampled or missing data, Technical Report 75, UW CSSS.

Giles, H. (1973), 'Accent mobility', *Anthropological Linguistics* **15**(2), 87–105.

Giles, H., Bourhis, R. Y. and Taylor., D. M. (1977), *Language, ethnicity, and intergroup relations*, Academic Press, chapter Towards a theory of language in ethnic group relations, pp. 307–348.

Giles, H., Coupland, N. and Coupland, J. (1991), *Contexts of accommodation: developments in applied sociolinguistics*, Cambridge University Press, chapter Accomodation theory: Communication, Context and Consequence, pp. 1–68.

Goodreau, S. M., Kitts, J. A. and Morris, M. (2009), 'Birds of a feather, or friend of a friend? Using exponential random graph models to investigate adolescent social networks', *Demography* **45**(1), 103–125.

Gray, R. M. and Davisson, L. D. (2004), *An Introduction to Statistical Signal Processing*, Cambridge University Press.

Greasley, P., Setter, J., Waterman, M., Sherrard, C., Roach, P., Arnfield, S. and Horton, D. (1995), Representation of prosodic and emotional features in a spoken language database, *in* 'Proc. of the International Congress of Phonetic Sciences'.

Gregory, Jr., S. W., Dagan, K. and Webster, S. (1997), 'Evaluating the relation of vocal accommodation in conversation partners' fundamental frequencies to perceptions of communication quality', *Journal of Nonverbal Behavior* **21**(1), 23–43.

Gregory, Jr., S. W. and Hoyt, B. R. (1982), 'Conversation partner mutual adaptation as demonstrated by fourier series analysis', *Journal of Psycholinguistic Research* **11**(1), 35–46.

Gregory, Jr., S. W. and Webster, S. (1996), 'A nonverbal signal in voices of interview partners effectively predicts communication accommodation and social status perceptions', *Journal of Personality and Social Psychology* **70**(6), 1231–1240.

Griffiths, T. L. and Steyvers, M. (2004), 'Finding scientific topics', *PNAS* **101**, 5228–5235.

Guo, F., Hanneke, S., Fu, W. and Xing, E. C. (2007), Recovering temporally rewiring networks: A model-based approach, *in* 'Proc. of ICML'.

Handcock, M. (2003), Assessing degeneracy in statistical models of social networks, Technical Report 39, UW CSSS.

Hanneke, S., Fu, W. and Xing, E. (2010), 'Discrete temporal models of social networks', *Electronic Journal of Statistics* **4**, 585–605.

Hastings, W. K. (1970), 'Monte carlo sampling methods using markov chains and their applications', *Biometrika* **57**(1), 97–108.

Hawkins, K. (1991), 'Some consequences of deep interruption in task-oriented communication', *Journal of Language and Social Psychology* **10**, 185–203.

Hofmann, T. (2001), 'Unsupervised learning by probabilistic latent semantic analysis', *Machine Learning* **42**(1), 177–196.

Holland, P. W. and Leinhardt, S. (1975), The statistical analysis of local structure in social networks, *in* 'Sociological Methodology', Jossey-Bass, pp. 1–45.

Holland, P. W. and Leinhardt, S. (1977), 'A dynamic model for social networks', *Journal of Mathematical Sociology* **5**, 5–20.

Holland, P. W. and Leinhardt, S. (1981), 'An exponential family of probability distributions for directed graphs', *Journal of the American Statistical Association* **76**(373), 33–50.

Hunter, D., Goodreau, S. and Handcock, M. (2008), 'Goodness of fit of social network models', *Journal of the American Statistical Association* **103**(481), 248–258.

Hunter, D. R. and Handcock, M. (2006), 'Inference in curved exponential family models for networks', *Journal of Computational and Graphical Statistics* **15**(3), 565–583.

Hurlburt, R., Koch, M. and Heavey, C. (2002), 'Descriptive experience sampling demonstrates the connection of thinking to externally observable behavior', *Cognitive Therapy and Research* **26**(1), 117–134.

Kampstra, P. (2008), 'Beanplot: A boxplot alternative for visual comparison of distributions', *Journal of Statistical Software* **28**(1), 1–9.

Kass, R. E. and Vos, P. W. (1997), *Geometrical Foundations of Asymptotic Inference*, Wiley.

Killworth, P. D. and Bernard, H. R. (1976), 'Informant accuracy in social network data', *Human Organization* **35**(3), 269–286.

Killworth, P. D. and Bernard, H. R. (1979), 'Informant accuracy in social network data: III a comparison of triadic structure in behavioral and cognitive datasets', *Social Networks* **2**, 10–46.

Klovdahl, A. S. (1985), 'Social networks and the spread of infectious diseases: The AIDS example', *Social Science and Medicine* **21**(11), 1203–1216.

Koller, D. and Friedman, N. (2009), *Probabilistic Graphical Models*, MIT Press.

Kossinets, G. and Watts, D. J. (2006), 'Empirical analysis of an evolving social network', *Science* **311**, 88–90.

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

Larson, R. and Csikszentmihalyi, M. (1987), 'Validity and reliability of the experience-sampling method', *Journal of Nervous and Mental Diseases* **175**(9), 526–536.

Lauritzen, S. (1996), *Graphical Models*, Oxford UP.

Lazega, E. and van Duijn, M. (1997), 'Position in formal structure, personal characteristics and choices of advisors in a law firm: A logistic regression model for dyadic network data', *Social Networks* **19**, 375–397.

Lehmann, E. L. and Casella, G. (1998), *Theory of Point Estimation*, Springer.

Leskovec, J., Lang, K. J., Dasgupta, A. and Mahoney, M. W. (2008), Statistical properties of community structure in large social and information networks, *in* 'Proc. of WWW'.

Lester, J., Choudhury, T., Kern, N., Borriello, G. and Hannaford, B. (2005), A hybrid discriminative-generative approach for modeling human activities, *in* 'Proc. of IJCAI'.

Lin, N. (1999), 'Social networks and status attainment', *Annual Review of Sociology* **25**, 467–487.

Marsden, P. V. (1990), 'Network and data measurement', *Annual Review of Sociology* **16**, 453–463.

McCallum, A., Wang, X. and Corrada-Emmanuel, A. (2007), 'Topic and role discovery in social networks with experiments on Enron and academic email', *Journal of Artificial Intelligence Research* **30**, 249–272.

McCowan, I., Bengio, S., Gatica-Perez, D., Lathoud, G., Monay, F., Moore, D., Wellner, P. and Bourlard, H. (2003), Modeling human interaction in meetings, *in* 'Proc. of ICASSP'.

Moreno, J. L. and Jennings, H. H. (1938), 'Statistics of social configurations', *Sociometry* **1**(3–4), 342–374.

Morgan, N., Fosler, E. and Mirghafori, N. (1997), Speech recognition using on-line estimation of speaking rate, *in* 'Proc. of Eurospeech'.

NIST (2009), 'NIST rich transcription evaluations', http://www.itl.nist.gov/iad/mig/tests/rt/2009/index.html.

Onnela, J.-P., Saramäki, J., Hyvönen, J., Szabó, G., de Menezes, M. A., Kaski, K., Barabási, A.-L. and Kertész, J. (2007), 'Analysis of a large-scale weighted network of one-to-one human communication', *New Journal of Physics* **9**, 179.

Onnela, J.-P., Saramäki, J., Hyvönen, J., Szabó, G., Lazer, D., Kaski, K., Kertész, J. and Barabási, A.-L. (2007), 'Structure and tie strengths in mobile communication networks', *PNAS* **104**(18), 7332–7336.

Palla, G., Barabási, A.-L. and Vicsek, T. (2006), 'Quantifying social group evolution', *Nature* **446**, 664–667.

Pentland, A. S. (2007), 'Automatic mapping and modeling of human networks', *Physica A* **378**(1), 59–67.

Quatieri, T. (2001), *Discrete-Time Speech Signal Processing: Principles and Practice*, Prentice Hall.

Rabiner, L. (1977), 'On the use of autocorrelation analysis for pitch detection', *IEEE Transactions on Acoustics, Speech, and Signal Processing* **25**(1), 24–33.

Rabiner, L. R. (1989), 'A tutorial on hidden markov models and selected applications in speech recognition', *Proceedings of the IEEE* **77**(2), 257–286.

Radcliffe-Brown, A. R. (1940), 'On social structure', *Journal of the Royal Anthropological Institute of Great Britain and Ireland* **70**(1), 1–12.

Reynolds, D. A. and Torres-Carrasquillo, P. (2005), Approaches and applications of audio diarization, *in* 'Proc. of ICASSP'.

Robert, C. P. and Casella, G. (2004), *Monte Carlo Statistical Methods*, Springer.

Robins, G. and Pattison, P. (2001), 'Random graph models for temporal processes in social networks', *Journal of Mathematical Sociology* **25**(1), 5–41.

Robins, G., Pattison, P. and Elliott, P. (2001), 'Network models for social influence processes', *Psychometrika* **66**(2), 161–190.

Robins, G., Snijders, T., Wang, P., Handcock, M. and Pattison, P. (2007), 'Recent developments in exponential random graph (p*) models for social networks', *Social Networks* **29**(2), 192–215.

Sampson, F. (1969), Crisis in a Cloister, PhD thesis, Cornell University, Dept. of Sociology.

Saramäki, J., Kivelä, M., Onnela, J.-P., Kaski, K. and Kertész, J. (2007), 'Generalizations of the clustering coefficient to weighted complex networks', *Physical Review E* **75**(027105), 1–4.

Saul, L. K. and Jordan, M. I. (1999), 'Mixed memory markov models: Decomposing complex stochastic processes as mixtures of simpler ones', *Machine Learning* **37**, 75–86.

Schuller, B., Rigoll, G. and Lang, M. (2004), Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine-belief network architecture, *in* 'Proc. of ICASSP'.

Smith, K. P. and Christakis, N. A. (2008), 'Social networks and health', *Annual Review of Sociology* **34**, 405–429.

Snijders, T. (2002), 'Markov chain monte carlo estimation of exponential random graph models', *Journal of Social Structure* **3**(2), 1–40.

Snijders, T. A., Pattison, P., Robins, G. L. and Handcock, M. S. (2006), 'New specifications for exponential random graph models', *Sociological Methodology* **36**, 99–153.

Solzhenitsyn, A. (1968), *In the First Circle*, Harper Perennial, chapter Voiceprints, pp. 233–239. Trans. Harry T. Willets, 2009.

Strauss, D. (1986), 'On a general class of models for interaction', *SIAM* **28**(4), 513–527.

Stupakov, A., Hanusa, E., Bilmes, J. and Fox, D. (2009), COSINE - a corpus of multi-party conversational speech in noisy environments, *in* 'Proc. of ICASSP'.

Valente, T. W. (1996), 'Social network thresholds in the diffusion of innovations', *Social Networks* **18**, 69–89.

Wasserman, S. and Faust, K. (1994), *Social Network Analysis*, Cambridge University Press, Cambridge.

Wasserman, S. and Iacobucci, D. (1988), 'Sequential social network data', *Psychometrika* **53**(2), 261–282.

Wasserman, S. and Pattison, P. (1996), 'Logit models and logistic regression for social networks: I. an introduction to markov graphs and (p*)', *Psychometrika* **61**(3), 169–193.

Watts, D. J. and Strogatz, S. H. (1998), 'Collective dynamics of 'small-world' networks', *Nature* **393**, 440–442.

White, H. C., Boorman, S. A. and Breiger, R. L. (1976), 'Social structure from multiple networks, I: Blockmodels for roles and positions', *American Journal of Sociology* **81**, 730–739.

Wren, C. R., Ivanov, Y. A., Leigh, D. and Westhues, J. (2007), The MERL motion detector dataset, Technical Report 2007-069, MERL.

Wyatt, D. (2009), Collective modeling of human social behavior, *in* 'Proc. of AAAI Spring Symposium on Human Behavior Modeling'.

Wyatt, D., Choudhury, T. and Bilmes, J. (2007), Conversation detection and speaker segmentation in privacy-sensitive situated speech data, *in* 'Proc. of Interspeech'.

Wyatt, D., Choudhury, T. and Bilmes, J. (2008), Learning hidden curved exponential random graph models to infer face-to-face interaction networks from situated speech data, *in* 'Proc. of AAAI'.

Wyatt, D., Choudhury, T. and Bilmes, J. (2009), Dynamic multi-valued network models for predicting face-to-face conversations, *in* 'NIPS workshop on Analyzing Networks and Learning with Graphs'.

Wyatt, D., Choudhury, T. and Bilmes, J. (2010), Discovering long range properties of social networks with multi-valued time-inhomogeneous models, *in* 'Proc. of AAAI'.

Wyatt, D., Choudhury, T., Bilmes, J. and Kautz, H. (2007), A privacy-sensitive approach to modeling multi-person conversations, *in* 'Proc. of IJCAI'.

Wyatt, D., Choudhury, T., Bilmes, J. and Kitts, J. (2008), Towards the automated social analysis of situated speech data, *in* 'Proc. of UbiComp'.

Wyatt, D., Choudhury, T. and Kautz, H. (2007), Capturing spontaneous conversation and social dynamics: A privacy sensitive data collection effort, *in* 'Proc. of ICASSP'.

Vita

Daniel Wyatt was born in Houston, Texas in 1975 and shortly thereafter began going by Danny. In 1994 he moved to New York to attend Columbia University, where he received his Bachelor of Arts, *cum laude*, in English and Comparative Literature in 1998. In 2003 he moved to Seattle to start his graduate studies. He received his Master of Science in 2005 and his Doctor of Philosophy in 2010, both in Computer Science and Engineering from the University of Washington.