

Evolving Social Network Analysis:
developments in statistical methodology
for dynamic stochastic actor-oriented
models



Charlotte C. Greenan

St Anne's College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

October 2015

Abstract

A social network will often be dynamic, with social connections changing over time. Incorporating temporal changes into a model for a social network will not only make it more realistic, but will allow us to capture mechanisms and uncover insights that would otherwise be impossible. In this thesis, we extend and improve the Siena methodology, proposed by Snijders (2001), the most prominent and widely used framework for modelling dynamic social networks.

In Part II, we propose a model for the diffusion of an innovation as it propagates through the dynamic network. Our model will synthesise the Siena network model with a proportional hazards model that is standard in survival analysis, providing us with easily interpretable parameters for the diffusion process.

Obtaining the maximum likelihood estimate given longitudinal data and a Siena model is a computationally demanding task, and in Part III, we consider improvements to the existing method. We show that these improvements can reduce computation times substantially.

In Part IV, we consider various uses of Monte Carlo simulation in the Siena framework. We focus on efficiently performing likelihood ratio tests, in order to perform model selection, and improving the estimation of standard errors.

Acknowledgements

I would like to thank my supervisor Professor Tom Snijders for invaluable comments, conversations and advice, and the Engineering and Physical Sciences Research Council for financial support.

Contents

I	Introduction	17
1	Introduction	19
1.1	Notation and some definitions	22
1.2	Social network mechanisms	23
1.2.1	Reciprocity	23
1.2.2	Triads	24
1.2.3	Interactions with actor attributes	24
1.2.4	Centrality	25
1.3	Outline and motivations	25
2	The Siena model	27
2.1	Model for a dynamic network	28
2.1.1	Notation and data structure	28
2.1.2	Overview and Assumptions	29
2.1.3	Model specification	31
2.1.4	Summary of the model	35
2.2	Coevolution of a dynamic network and individual behaviours	38
2.2.1	Notation and data structure	39
2.2.2	Overview and assumptions	40
2.2.3	Model specification	42

2.2.4	Discussion of the model	48
2.3	Parameter estimation	50
2.3.1	Robbins-Monro stochastic approximation algorithm	51
II	Diffusion of innovations	55
3	Diffusion of innovations in dynamic networks	57
3.1	Introduction	57
3.2	Notation and data structure	60
3.3	Model for the simultaneous diffusion of an innovation and evolution of a dynamic network	60
3.3.1	Adoption process	62
3.3.2	Discussion of the model	70
3.4	Parameter estimation	72
3.4.1	Asymptotic behaviour of estimates of α	72
3.5	Example: Glasgow school data	79
3.6	Simulation study	82
3.7	Discussion	87
III	Local network models	91
4	Local network models	93
4.1	Introduction	93
4.2	Local network models	95
4.3	Categorising polynomial effects	100
5	Maximum likelihood	103
5.1	Introduction	103

5.2	Metropolis-Hasting algorithm	104
5.2.1	Chain notation	106
5.2.2	Proposal distribution	106
5.3	Improving efficiency for local models	108
5.3.1	Example	112
5.4	Improving the proposal distribution	114
5.4.1	Moving a tie	116
5.5	Optimising the proposal distribution	117
5.5.1	Example 1: s50 data	119
5.5.2	Example 2: Glasgow data	124
5.6	Non-local models	127
5.7	Effective sample size	130
5.8	Summary of results	132

IV Importance Sampling in Maximum Likelihood Estimation **135**

6	Introduction	137
6.1	Simulations in <i>siena07</i>	138
6.2	Importance sampling	139
6.2.1	Entropy	145
7	Likelihood ratio test	147
7.1	Introduction	147
7.2	Likelihood ratio test	148
7.3	Approximating the test statistic	149
7.3.1	Fast forward selection	149
7.3.2	Bridge sampling	153

7.3.3	Thermodynamic integration	160
7.4	Evaluating the difficulty of the test	161
7.5	Simulation studies	164
7.5.1	Fast forward selection	165
7.5.2	Bridge sampling and thermodynamic integration . .	171
7.6	Discussion	177
8	Standard error estimation	181
8.1	Introduction	181
8.2	Covariance matrix estimation	182
8.2.1	Estimating the observed information in Phase 2 . . .	183
8.2.2	Combining the estimates	188
8.2.3	Practical Implications	191
8.3	Example: <i>s</i> 50 data	191
8.3.1	Improving performance	193
8.3.2	Saving time	198
8.4	Discussion	200

List of Figures

3.1	Deviation between observed and expected value of the moment statistic associated with the average exposure effect.	86
5.1	Computation times for a Siena analysis of the three datasets described in Section 5.3.1.	115
5.2	Maximum t convergence statistics using different values of (L, K_3, K_6) with autocorrelations at most 0.4.	123
5.3	Maximum t convergence statistics using different values of (L, K_3, K_6) with autocorrelations at most 0.4.	126
5.4	Proportion of accepted ‘move’ updates for a non local model using different values of (L, K_3, K_6)	129
5.5	Maximum autocorrelations for a non local model using different values of (L, K_3, K_6)	130
5.6	Computation times for a non local model using different values of (L, K_3, K_6)	131
5.7	Maximum t convergence statistics for a non local model using different values of (L, K_3, K_6)	131
5.8	Effective sample sizes by parameter for a local model using different values of (L, K_3, K_6)	133
7.1	Fast forward selection; empirical distribution of test statistics are plotted in red; translucent distributions show the corresponding χ_k^2 distribution, where k is the degrees of freedom of the test.	168

- 7.2 Fast forward selection; type 1 errors for different significance levels, for tests where the null hypothesis is true. . . . 169
- 7.3 Fast forward selection; (left) power for different significance levels, for the test where the null hypothesis is false; (right) empirical distribution of test statistics are plotted in red, and translucent distributions shows the χ_1^2 distribution. . . . 170
- 7.4 For different values of (c, N) , the empirical distribution of test statistics in shown in red; translucent distributions show the corresponding χ_k^2 distribution, where k is the degrees of freedom of the test: (top) bridge sampling; (middle) thermodynamic integration, $H = 10$; (bottom) thermodynamic integration, $H = 20$ 172
- 7.5 Type 1 errors for different significance levels, using different values of (c, N) and: (top) bridge sampling; (middle) thermodynamic integration, $H = 10$; (bottom) thermodynamic integration, $H = 20$ 173
- 7.6 Time taken to perform test for different values of (c, N) and: (top) bridge sampling; (middle) thermodynamic integration, $H = 10$; (bottom) thermodynamic integration, $H = 20$ 176
- 7.7 Using bridge sampling and different values of (c, N) : (top) empirical distribution of test statistics compared to χ_1^2 distribution; (middle) power for different significance levels; (bottom) time taken to estimate test statistics. 178
- 8.1 Top plot shows how the estimate of one component of θ (the rate parameter) varies throughout the estimation procedure, in Phases 2 and 3; bottom plot shows the corresponding values of $(\alpha_0, \dots, \alpha_4)$ 192
- 8.2 $n_3 = 500$; the density of the estimates of the variance (results using the new proposed method are shown in red, and the translucent distribution shows results using Phase 3 simulations only). The vertical line shows the overall best estimate, obtained by taking the inverse of the average of the information matrices across all $N = 100$ repetitions. 194
- 8.3 $n_3 = 1000$ 195
- 8.4 $n_3 = 1500$ 196

8.5 $n_3 = 2000$ 197

8.6 Ratio of root mean squared error for all components of the covariance matrix against the number of simulations in Phase 3, n_3 199

8.7 Proposed \hat{n}_3 to obtain a smaller root mean squared error, for all components of the covariance matrix, using the new method, than that obtained using n_3 and Phase 3 simulations only. 201

List of Tables

2.1	Structural effects. A dashed line indicates that were that tie to exist, then the effect measure $s_i(x)$ would increase.	36
2.2	Covariate effects for a binary actor covariate. A dashed line indicates that were that tie to exist, then the effect measure $s_i(x, z)$ would increase. A shaded node indicates that the corresponding covariate is one; otherwise it is zero.	37
3.1	Results for the Glasgow school dataset.	81
3.2	Results for the simulation study with 129 actors.	84
5.1	Updated chain of ministeps, and their probabilities, after adding a cancelling pair of ministeps toggling (l, m)	110
5.2	Updated chain of ministeps, and their probabilities, after deleting a cancelling pair of ministeps m_{r_1} and m_{r_2} which both toggle (l, m)	110
5.3	Using permutations with s50 data: autocorrelations and computation times for different pairs of (L, K_3) , where $K_6 = 0$	122
5.4	Using moves with s50 data: autocorrelations and computation times for different pairs of (L, K_6) , where $K_3 = 0$	122
5.5	Using permutations with Glasgow data: autocorrelations and computation times for different pairs of (L, K_3) , where $K_6 = 0$	125
5.6	Using moves with Glasgow data: autocorrelations and computation times for different pairs of (L, K_6) , where $K_3 = 0$	125

7.1	Effects included in each model (density and reciprocity included in every model).	163
7.2	Relative importance of effects.	164
7.3	Fast forward selection with $\theta_0 = (6.2, -2.50, 2.11, 0.58, 0, 0, 0)$	166

Part I

Introduction

Chapter 1

Introduction

Society is a 'web or tissue of human interactions and interrelations' (Ginsberg, 1939); it is 'the sum of formal relations in which associating individuals are bound together' (Giddings, 1896); it 'does not consist of individuals but expresses the sum of interrelations, the relations within which these individuals stand' (Marx, 1939). When defining society, the importance of social relations is often emphasised, and so if we wish to further our understanding of a range of disciplines, such as sociology, anthropology or politics, then it seems important to look to the connections between people. Social network analysis provides us with a flexible and quantitative way to do this.

In an early example of social network analysis, while studying Norwegian fishermen, Barnes (1954) noted the similarities between the nets that they used and the structure of the community in which they lived. Social network analysis has since flourished as a framework for representing so-

cial structures (Wasserman and Faust, 1994), accounting for patterns and regularities, and depicting the social world as ‘an intertwined mesh of connections through which individuals [are] bound together’ (Scott, 1988).

Typically, a social network consists of a set of social actors and a finite number of predefined relations between them; this framework allows us to represent a huge variety of social structures. The social actor is very often an individual person, where examples of relations are friendship, trust, advice, or collaboration (Wasserman and Faust, 1994). For example, the widely used ‘Bank-wiring room’ data (Roethlisberger and Dickson, 1939) measures a variety of relations ranging from standard (such as ‘liking’ and ‘antagonism’), to quite idiosyncratic (e.g. participation in ‘controversies about windows’). However, the social actors need not be individual people: social network analysis is used to study incredibly diverse scenarios, ranging from countries connected by export relationships (Rhue and Sundararajan, 2014) to birds related by pecking (Allee, 1958; Faust, 2014) (with the latter research area reputedly giving rise to the phrase ‘pecking order’ (Perrin, 1955)).

Social network analysis is important because relationships can be as important, and in many applications, more important, than actor attributes, when trying to understand observed behaviour (Knoke and Yang, 2008). For example, many of a person’s attributes (such as gender and race) are unchanged across the variety of social contexts in which they participate (such as home or work), whereas structural relations may vary and be defined in each place. An example given by Knoke and Yang (2008) suggests

a scenario where a person has ‘little initiative’ at work but is the ‘dynamic leader of [their] neighbourhood association’. Such contrasting behaviours may be difficult to reconcile with their unchanging attributes, but could be captured were we to observe their positions in the distinct social networks based at their work and neighbourhood.

Looking at the structure of a network can provide us with a lot of information, but for many applications, it is desirable, and even necessary, to also look at a network as a dynamic process, where the social relations between social actors change over time (Doreian and Stokman, 2013). If we consider the case where the social actors are people, and the relation between them friendship, then it is an empirical fact that the social ties are not constant: the relationship had to begin at some point, and can certainly end.

If we know that the social network is not constant, and we wish to create as realistic a model as possible, then one which incorporates the network changes is highly desirable. Moreover, a model with a network that is dynamic allows us to capture mechanisms and uncover insights which would be impossible considering only a static network. One of the most important, which we will discuss further in Chapter 2, is the idea of ‘disentangling selection and influence’ (Steglich et al., 2010). This can be summarised colloquially as attempting to simultaneously answer the two questions: *Do friends become similar? Do similar people become friends?* We must look at social actors, and their network relations, and how both change over time, to answer these questions.

Dynamic network analysis has been described as ‘a Holy Grail for network researchers’ (Carrington et al., 2005). This thesis will use the Siena model, which was developed by Snijders (2001) to serve this purpose. This model is popular and widely used; it is described by Veenstra and Steglich (2012) as ‘the prominent tool for the analysis of longitudinal network data’. It is referred to as a ‘stochastic actor-oriented’ model (Snijders et al., 2010), because its formulation is such that it can be interpreted that changes to the network are driven by the actors. In other words, each actor is in control of their outgoing ties, and they may choose to change or maintain their relationships according to certain tendencies and inclinations. These can be based on the network, such as a preference for reciprocated relationships or simply a desire for a higher number of connections; however, they can also depend on actor attributes, such as a predilection towards homophily, where actors prefer those with similar behaviours, beliefs or practices. In Section 1.2 we will describe some basic social mechanisms that can be operationalised to express tendencies such as these.

1.1 Notation and some definitions

A social network consisting of n actors is modelled as a directed graph, with an edge (which we will often refer to as a *tie*) from one node to another corresponding to a specific social relation between them, such as friendship, trust or collaboration. The directed graph can be represented by an $n \times n$ adjacency matrix $x = (x_{ij})$, where $x_{ij} = 1$ if and only if a tie

exists from actor i to actor j ; otherwise $x_{ij} = 0$. The ties are assumed to be directed (so that $x_{ij} = 1$ does not imply $x_{ji} = 1$) and non-reflexive (so that $x_{ii} = 0$ for all $i \in \{1, \dots, n\}$).

We refer to the 'sender' of a tie as the *ego* actor, and the 'receiver' as the *alter*; we will sometimes refer to the set of actors to whom an ego has ties as their *alters*.

1.2 Social network mechanisms

In this thesis, we are primarily interested in local structures of the network; although we model an entire network, our analysis is usually interpreted in terms of the actors within. In this section we briefly describe some mechanisms that may be observed at the micro-level of a social network, and why they may be of interest.

1.2.1 Reciprocity

In this thesis, we consider directed networks, and in such networks there is often a tendency towards reciprocity, or mutuality: a tie from one actor to another will often be returned. This is one of the most basic mechanisms and will usually be incorporated into a model.

1.2.2 Triads

The triads of a network are the subgraphs on sets of three nodes. Much of the local structure of the network can be described using the triads (Holland and Leinhardt, 1977). One important configuration of the triads is transitivity, which can be described as the tendency of a social actor to become ‘friends’ with a ‘friend of a friend’.

Another configuration widely examined is known as a ‘three-cycle’, where the ties between the three actors forms a cycle of length three. Often it is found that there is a tendency *against* this type of configuration, which can be interpreted as evidence of local hierarchy (Davis, 1970).

1.2.3 Interactions with actor attributes

Actor attributes, such as demographics, can interact with the network in significant ways. Actors with different characteristics may prefer different amount of network connections, or have differing levels of popularity: for example, Pearson et al. (2006) found that girls preferred to have more friends but were themselves less attractive as friends.

Researchers may often be interested in tendencies for actors to prefer to maintain relationships with those with whom they share similarities, known as homophily (McPherson et al., 2001). For example, in the same study mentioned above, Pearson et al. (2006) found that tobacco and alcohol users preferred other users as friends.

1.2.4 Centrality

There are various ways to measure a social actor's centrality in the network. The most obvious are the number of incoming ties, known as *indegree*, or the number of outgoing ties, known as *outdegree*.

Another centrality measure is *betweenness*; this measures how many times an actor is connected to two unconnected actors, and can be interpreted as a measure of their 'brokerage' (Marsden, 2002).

Centrality measures can be important in determining who is important in the network, whether often we define 'important' to mean influential or an opinion leader (Valente, 1996). Considering the micro-level of the network, social actors may take steps to obtain high centrality, in an attempt to be popular and influence others, or they may prefer to connect with others with high centrality, as this may manifest itself as an attractive quality.

1.3 Outline and motivations

In this thesis, our overall goal is to extend and improve the Siena methodology for analysing dynamic social networks. We therefore begin by describing the Siena model, in Chapter 2.

In Part II, we propose a model for the diffusion of an innovation as it propagates through the dynamic network. Our model will synthesise the Siena network model with a proportional hazards model that is standard in sur-

vival analysis, providing us with easily interpretable parameters for the diffusion process.

As will be discussed in Chapter 2, parameter estimation for Siena model is usually achieved using method of moments (Bowman and Shenton, 1985), mainly due to its computational efficiency. In Part III, we consider maximum likelihood estimation, which is already available in Siena, but is computationally demanding (Snijders et al., 2010). We consider special cases of the Siena model conditional on which substantial efficiency improvements can be made. The aim is to improve performance enough to make maximum likelihood a practically viable method for estimation.

In Part IV, we continue with maximum likelihood estimation, and consider various uses of Monte Carlo simulation for maximum likelihood estimation. We focus on efficiently performing likelihood ratio tests, in order to perform model selection, and improving the estimation of standard errors.

Chapter 2

The Siena model

Dyads in a social network are inherently interdependent: whether or not there is a relationship between two social actors will depend on the actors' other relationships, or lack thereof (Robins and Pattison, 2005) . There are few models for dynamic social networks which account for these dependencies, and, of them, the Siena model (Snijders, 2001), is the most prominent and widely used (Veenstra and Steglich, 2012).

For a collection of social actors, the Siena model assumes that a dynamic network process evolves as a continuous time Markov chain on the space of all possible networks between the actors; the parametrisation of the transition intensity is such that one can interpret that the process changes as actors stochastically maximise an objective function which encapsulates their relationship preferences; for example, the function may increase when an ego has more reciprocated ties, or ties with alters similar to the ego.

In this chapter we review the Siena model, firstly describing the model for dynamic networks (Snijders, 2001), and then explaining how it is extended to model the coevolution of dynamic networks and social actor attributes (Snijders et al., 2007). After detailing these models, we describe the algorithm used for parameter estimation in Section 2.3.

2.1 Model for a dynamic network

In this section, we consider the case where we wish to model a single dynamic network. In Section 2.1.1, we describe the some of the notation we will use throughout the thesis. We give an overview of the model and its assumptions in Section 2.1.2, before detailing the model specification in Section 2.1.3.

2.1.1 Notation and data structure

Recall from Chapter 1 that we model a social network of n actors using a direct graph which can be represented using an $n \times n$ adjacency matrix x . In general, the network varies in time, so that $X = X(t)$, for $t \in \mathcal{T} \subset \mathbb{R}$, where \mathcal{T} is a closed and bounded interval of the real line; the upper case notation is used to indicate that the process is modelled as stochastic. An observation of the network at time t is denoted by $x(t)$.

We study longitudinal data: at a finite number (strictly larger than one) of observation times $t_1 < \dots < t_M$, the network is observed, giving data

$X(t_1) = x(t_1), \dots, X(t_M) = x(t_M)$. The fact that the variables are observed only at finitely many time points will later be seen to pose special requirements on the estimation procedure. Covariates, either individual (recorded for each actor) or dyadic (recorded for each pair of actors), can also be measured at each observation; these are assumed to be fixed between observation times and non-stochastic. Individual and dyadic covariates are referred to as $v = (v_1, \dots, v_n)$ and $w = (w_{ij})_{1 \leq i, j \leq n}$ respectively, with their possible dependence on the interval $[t_{m-1}, t_m]$ for $m \in \{2, \dots, M\}$ suppressed in the notation. Moreover, at times when it is more convenient, we will refer to the process $(X(t), v, w)$ as $X(t)$, with the covariates included only implicitly. An observation of the process at time t is denoted by $x(t)$. We refer to the multi-dimensional parameter governing this stochastic process as θ .

2.1.2 Overview and Assumptions

We assume that the observed networks are outcomes of an underlying continuous time Markov chain, as has been suggested by Holland and Leinhardt (1977); this means that, between two observation times t_{m-1} and t_m , the process $\{X(t) : t_{m-1} \leq t < t_m\}$, is a continuous time Markov chain with initial state given by the observation $X(t_{m-1}) = x(t_{m-1})$. The state space of the process is \mathcal{X} , where \mathcal{X} is the set of binary adjacency matrices, and given two distinct states $x, \hat{x} \in \mathcal{X}$, we denote the transition intensity for moving from x to \hat{x} at time t by $Q_\theta(x, \hat{x}, t)$.

An important assumption of the model is that, at time t , given the current state of the process $X(t)$, the times until the next changes by the members of the set of network ties are conditionally independent random variables. This means that at any one time point, the probability of more than one change to these random variables is zero (Billingsley, 1995); therefore, there can be at most one tie change to the network at any one point in time. We refer to such a change as a *mini-step*. This assumption excludes scenarios such as a mutual decision by two social actors to extend ties to one another simultaneously; instead, each must do so one at a time. This formulation also means that there will be lots of inertia in the model, and so it is appropriate only for a social network in which the ties are enduring in nature; for example, friendship and trust would be suitable, but highly transient ties such as email contact or meeting at events would not.

Given the current state of the process $X(t) = x$ (we suppress the dependence of x on t to simplify the notation), only a single tie can change; in other words, the network moves from x to a member of the set

$$\mathcal{A}(x) = \bigcup_i \bigcup_{j \neq i} \{x(i \rightsquigarrow j)\}, \quad (2.1)$$

where $x(i \rightsquigarrow j) \in \mathcal{X}$ is the adjacency matrix that is the same as x in all but the (i, j) th entry, so that

$$x(i \rightsquigarrow j)_{hk} = \begin{cases} 1 - x_{hk} & \text{if } (h, k) = (i, j), \\ x_{hk} & \text{if } (h, k) \neq (i, j). \end{cases}$$

If $\hat{x} \in \mathcal{X}$ is not a member of the set $\mathcal{A}(x) \cup \{x\}$ then the transition intensity $Q_\theta(x, \hat{x}, t)$ is zero.

2.1.3 Model specification

We consider the transition intensity of an actor changing an outgoing network tie. Given the current state of the process $X(t) = x$, and given a state $x(i \rightsquigarrow j) \in \mathcal{A}(x)$ that can be reached from x by a single mini-step, we model the transition intensity using the following decomposition

$$Q_\theta(x, x(i \rightsquigarrow j), t) = \lambda_i(x, t; \theta) p_{ij}(x; \theta). \quad (2.2)$$

This decomposition can be interpreted in the following way, proposed by Snijders (2001): the process by which actor i changes one of their outgoing ties $\{X_{ij} : j \in \{1, \dots, n\}, j \neq i\}$ has two stages; firstly, they must obtain an opportunity to alter their ties, and secondly, they must choose which of these to change. The time taken until a change opportunity arises for actor i is modelled as being governed by the *rate function* $\lambda_i(x, t; \theta)$, and the subsequent choice of what change to make by the *conditional choice probabilities* $(p_{i1}(x; \theta), \dots, p_{in}(x; \theta))$.

Rate function

The time until actor i has an opportunity to change their outgoing ties is modelled as an exponential random variable, with rate parameter depend-

ing on the time and current state of the process. The rate parameter need not be the same for all actors; instead, covariates and some functions of the network, such as the number of outgoing ties that an actor has, can be incorporated into the rate function. However, in the simplest case (which we consider in this thesis) it is modelled as the same for each actor, and not dependent on the current state of the network, so that the rate function for actor i is defined as

$$\lambda_i(x, t; \theta) = \rho^{[X]}(t). \quad (2.3)$$

Between observations at times t_{m-1} and t_m , the number of opportunities that each actor has to change their ties is distributed as a Poisson random variable with mean

$$\rho_m^{[X]} := \int_{t_{m-1}}^{t_m} \rho^{[X]}(t) dt. \quad (2.4)$$

Conditional choice probabilities

Given the current state of the network $X(t) = x$, and given that actor i has an opportunity to change one of their outgoing ties, they must choose whether to extend a new tie, dissolve an existing tie, or leave the network unchanged. The choice is governed by the conditional choice probabilities $p_{i1}(y; \theta), \dots, p_{in}(y; \theta)$ (seen in equation (2.2) for $j \neq i$). We define $p_{ii}(x; \theta) = 1 - \sum_{j \neq i} p_{ij}(x; \theta)$. For $j \in \{1, \dots, n\}$ the network changes from x to $x(i \rightsquigarrow j)$ (where we define $x(i \rightsquigarrow i) = x$) with probability $p_{ij}(x; \theta)$. The

inclusion of a potentially non-zero probability $p_{ii}(x; \theta)$ that the network remains unchanged can be interpreted as accounting for the fact that actors satisfied with the state of network will prefer to leave their ties unaltered. We model the conditional choice probabilities in a multinomial logit form: for $j \neq i$, given a change opportunity and the current state of the process $X(t) = x$, actor i changes their tie to actor $j \in \{1, \dots, n\}$ with probability

$$p_{ij}(x; \theta) = \frac{\exp(f_i(x(i \rightsquigarrow j); \theta))}{\sum_{k=1}^n \exp(f_i(x(i \rightsquigarrow k); \theta))}, \quad (2.5)$$

where $f_i(x; \theta)$ is referred to as the *objective function*. This parametrisation is a common choice in random utility modelling (see, for example, Maddala (1983)); one interpretation of this is that actor i prefers networks for which $f_i(x; \theta)$ is high; Maddala (1983) shows that this formulation is equivalent to saying that, given the current state of the process $X(t) = x$, and given a change opportunity, actor i chooses $j \in \{1, \dots, n\}$ to maximise the function

$$M(j) = f_i(x(i \rightsquigarrow j); \theta) + \epsilon(j), \quad (2.6)$$

where $\epsilon(1), \dots, \epsilon(n)$ are independent random variables drawn from the type-1 extreme value distribution (also known as the standard Gumbel distribution).

The objective function is modelled as a weighted linear combination of

effects $s_{i1}(x), \dots, s_{iK^{[X]}}(x)$, for some known $K^{[X]}$, and has the form

$$f_i(x; \theta) = \sum_{k=1}^{K^{[X]}} \beta_k^{[X]} s_{ik}(x). \quad (2.7)$$

The weights $\beta_1^{[X]}, \dots, \beta_{K^{[X]}}^{[X]}$ are parameters to be estimated. *Effects* are functions of the current state of the process which are intended to encapsulate the different reasons an actor considers when making their tie change. There are many possible effects, and a number of them are detailed in the Appendix of Snijders et al. (2010). Some common choices, which we include in examples throughout this thesis, are:

1. *Outdegree effect*, the number of actor i 's outgoing ties.
2. *Reciprocity effect*, the number of actor i 's reciprocated ties.
3. *Transitive triplet effect*, the number of transitive patterns in actor i 's outgoing ties (for example, friends who are also a friend of a friend).
4. *'Number of actors at distance 2' effect*, the number of actors to whom actor i is not directly tied but who can be reached by a path of outgoing ties of length 2 (for example, a friend of a friend who is not themselves a friend).
5. *3-cycles effect*, the number of cyclical triangles that actor i appears in.
6. *Betweenness effect*, the number of pairs of actors (j, h) for whom h is tied to actor i and i is tied to actor j , but h is not tied to j . This effect represents the amount of 'brokerage' by actor i .

7. *Covariate ego effect*, the outdegree of actor i multiplied by the value of their covariate.
8. *Covariate alter effect*, the sum of the covariates of actor i 's ties.
9. *Covariate similarity effect*, the sum of the similarities between the covariate for actor i and the covariates for each of actor i 's. (This effect includes a centering constant c depending on the observed data is included to reduce correlations with the outdegree effect. The value of c is given by Ripley et al. (2011).)

Assuming that the weight parameter associated with one of these effects is positive, we can interpret the maximisation of (2.6) as a preference for tie changes which increases the number of relevant ties or patterns.

Tables 2.1 and 2.2 show the formulas and network diagrams for these effects (using a binary covariate in the diagrams, to simplify the illustration).

2.1.4 Summary of the model

In summary, we refer to the collection of all the unknown parameters as $\theta = (\rho, \beta)$, where $\rho = (\rho_2^{[X]}, \dots, \rho_M^{[X]})$ are those parameters associated with the rate function, whilst $\beta = (\beta_1^{[X]}, \dots, \beta_{K^{[X]}}^{[X]})$ are those associated with the network effects. Then, between observations at t_{m-1} and t_m , the continuous time Markov process $\{X(t) : t_{m-1} \leq t < t_m\}$ on state space \mathcal{X} has initial state $X(t_{m-1}) = x(t_{m-1})$, where $x(t_{m-1})$ is the observation of the process at t_{m-1} , and then evolves according to transition intensity

Table 2.1: Structural effects. A dashed line indicates that were that tie to exist, then the effect measure $s_i(x)$ would increase.

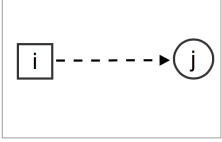
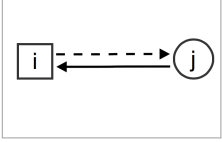
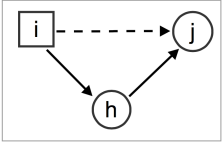
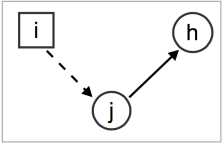
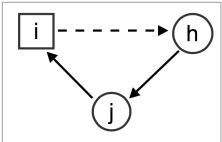
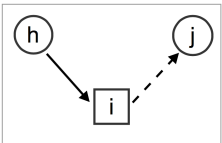
Effect	$s_i(x)$	
Outdegree	$\sum_j x_{ij}$	
Reciprocity	$\sum_j x_{ij}x_{ji}$	
Transitive triplets	$\sum_{j,h=1}^n x_{ij}x_{jh}x_{ih}$	
# actors at distance 2	$\sum_{h=1}^n (1 - x_{ih})1_{\{\max_j(x_{ij}x_{jh})>0\}}$	
3-cycles	$\sum_{j,h=1}^n x_{ij}x_{jh}x_{ih}$	
Betweenness	$\sum_{j,h=1}^n x_{ij}x_{hi}(1 - x_{hj})$	

Table 2.2: Covariate effects for a binary actor covariate. A dashed line indicates that were that tie to exist, then the effect measure $s_i(x, z)$ would increase. A shaded node indicates that the corresponding covariate is one; otherwise it is zero.

Effect	$s_i(x, z)$	
Covariate ego	$w_i \sum_{j=1}^n x_{ij}$	
Covariate alter	$\sum_{j=1}^n x_{ij} w_j$	
Covariate similarity	$\sum_{j=1}^n x_{ij} \left(1 - \frac{ w_j - w_i }{\max_{i,j} \{ w_j - w_i \}} - c \right)$	

$Q_\theta(x, \hat{x}, t) = \tilde{Q}_\theta(x, \hat{x}) \left(\rho^{[X]}(t) / \rho_m^{[X]} \right)$, for $\hat{x} \neq x$, where

$$\tilde{Q}(x, \hat{x}) = \begin{cases} \rho_m^{[X]} p_{ij}(x; \theta), & \text{if } \hat{x} = x(i \rightsquigarrow j) \in \mathcal{A}(x), \\ 0, & \text{otherwise,} \end{cases} \quad (2.8)$$

where $p_{ij}(y; \theta)$ is given by (2.5).

2.2 Coevolution of a dynamic network and individual behaviours

So far, in this chapter, we have described how to model the evolution of a dynamic network; however, often we would like to jointly model this process with an actor variable, which we will refer to as a *behaviour* (Snijders et al., 2007). This behaviour is modelled as a stochastic, continuous time variable.

If, for a collection of social actors, we were interested in both a dynamic social network connecting them, and a time varying individual level behaviour, it may often be the case that the network and behaviour will be interdependent, and so it is desirable to model them jointly. If we do this, as well as creating a model which is arguably more representative of the unknown underlying process, it also enables us to disentangle the distinct effects of influence and selection (Steglich et al., 2010). *Influence* refers to the way that social actors may choose their behaviour depending on the behaviours of those around them, perhaps due to a tendency to behave

similarly to one's peers, whilst homophilous *selection* can be described as the preference for instigating and maintaining relationships with those with whom they are similar with respect to some relevant behaviour. The Siena co-evolution model accounts for both of these effects by making the evolutions of the network and the behaviours dependent on one another; it is formulated so that at each point in time, the transition intensity determining the change to the network is dependent on the current state of the behaviour variable, and vice versa. The contrasting ideas of influence and selection (and the different implications to causality that they imply), are acknowledged by sociologists to exist jointly in many scenarios (e.g. Fisher and Bauman (1988), Pearson and Michell (2000), Kirke (2004)), and so it is important that they are both accounted for. Neglecting one may erroneously attribute too much importance to the other; for example, Bauman and Ennett (1996) argue that the impact of selection when studying adolescent smoking has been often neglected, but is important, and may partially confound the effect of influence, a widely accepted mechanism in such a scenario.

In this section, we describe the model for the co-evolution of a network and an individual-level variable formulated by Snijders et al. (2007).

2.2.1 Notation and data structure

We use the same notation for the network process $X(t)$, but now also model a behaviour process over the same time period \mathcal{T} . An actor's be-

haviour is modelled as random process which can take values in \mathcal{B} , a pre-determined finite interval of integers. For actor i , and at time t , it is denoted by $Z_i(t)$. Collectively, the vector $Z(t) = (Z_1(t), \dots, Z_n(t))$ is referred to as the behaviour variable or behaviour process.

At a finite number (strictly larger than one) of observation times $t_1 < \dots < t_M$, the network and behaviour variables are observed, giving data $(X(t_1), Z(t_1)) = (x(t_1), z(t_1)), \dots, (X(t_M), Z(t_M)) = (x(t_M), z(t_M))$. As before, covariates can also be measured.

The network and behaviour variables, and any individual or dyadic covariates, are jointly modelled as one process $Y(t) = (X(t), Z(t))$ (with the covariates included implicitly). An observation of the process at time t is denoted by $y(t)$. As before, we refer to the multi-dimensional parameter governing the stochastic process as θ .

2.2.2 Overview and assumptions

The network and behaviour processes are likely to be dependent on one another, so we consider them as a joint process $\{Y(t) = (X(t), Z(t)) : t \in \mathcal{T}\}$. Modelling as a joint process allows us to incorporate explanatory variables which account for the dependencies that the network and the behaviours have on one another; examples of these explanatory variables will be detailed later in this section. Note that this formulation means that the homophily mechanisms are assumed to be dependent only on the process $Y(t)$ and any measured covariates; latent homophily is not considered

here (Shalizi and Thomas, 2011).

The assumptions about this process following a continuous time Markov chain continue for this model: we assume that between two observation times t_{m-1} and t_m , the process $\{Y(t) : t_{m-1} \leq t < t_m\}$, is a continuous time Markov chain with initial state given by the observation $Y(t_{m-1}) = y(t_{m-1})$. The state space of this process is $\mathcal{X} \times \mathcal{B}^n$, where \mathcal{X} is the set of binary adjacency matrices and \mathcal{B}^n is the set of all n -tuples of the set \mathcal{B} (i.e. \mathcal{B}^n is the set of all values that the vector of behaviours can take). Given two distinct states $y, \hat{y} \in \mathcal{X} \times \mathcal{B}^n$, we denote the transition intensity for moving from y to \hat{y} at time t by $Q_\theta(y, \hat{y}, t)$.

As with the network model, we assume that there can be only one change to the process, either to the network or the behaviour variable, at any one point in time. We also assume that at any point in time, an actor's behaviour can change by at most one, either by increasing or decreasing.

Given the current state of the process $Y(t) = y = (x, z)$ (we suppress the dependence of y on t to simplify the notation), the process can change in either the network x or the behaviour variable z . If it is the network that updates then a single tie can change; in other words, the network moves from x to a member of the set $\mathcal{A}(x)$; see (2.1).

If it is the behaviour variable which changes, the only change that can occur is that a single actor may increase or decrease their behaviour by 1; in other words the behaviour variable changes from z to a member of the

set

$$\mathcal{R}(z) = \left[\bigcup_{i: z_i < \max(\mathcal{B})} \{z(i \uparrow 1)\} \right] \cup \left[\bigcup_{i: z_i > \min(\mathcal{B})} \{z(i \downarrow 1)\} \right],$$

where $z(i \uparrow 1), z(i \downarrow 1) \in \mathcal{B}^n$ and

$$z(i \uparrow 1)_j = \begin{cases} z_j + 1 & \text{for } j = i, \\ z_j & \text{for } j \neq i, \end{cases}$$

and

$$z(i \downarrow 1)_j = \begin{cases} z_j - 1 & \text{for } j = i, \\ z_j & \text{for } j \neq i. \end{cases}$$

Combining these two scenarios, given the current state $Y(t) = y = (x, z)$, a mini-step can change the process to a member of the set

$$\mathcal{M}(y) = \{\mathcal{A}(x) \times \{z\}\} \cup \{\{x\} \times \mathcal{R}(z)\}.$$

If $\hat{y} \in \mathcal{X} \times \mathcal{B}^n$ is not a member of the set $\mathcal{M}(y) \cup \{y\}$ then the transition intensity $Q_\theta(y, \hat{y}, t)$ is zero.

2.2.3 Model specification

Given the current state $Y(t) = y$, the model for the transition intensity $Q_\theta(y, \hat{y}, t)$ where $\hat{y} \in \mathcal{A}(x) \times \{z\}$ (i.e. where is the network that changes) is the same as described earlier, in Section 2.1.3. In this section, we describe the most common model used for the the transition intensity $Q_\theta(y, \hat{y}, t)$

where $\hat{y} \in \mathcal{A}(x) \times \{z\}$, i.e. where is the behaviour that changes (in Chapter 3 we will propose an alternative model for a special case of the behaviour variable).

Similarly to the model for a network change, for a change in behaviours, we model the transition intensity by considering the decompositions

$$Q_\theta(y, (x, z(i \uparrow 1)), t) = \lambda_i^{[Z]}(y, t; \theta) p_{i,1}^{[Z]}(y; \theta), \quad (2.9)$$

and

$$Q_\theta(y, (x, z(i \downarrow 1)), t) = \lambda_i^{[Z]}(y, t; \theta) p_{i,-1}^{[Z]}(y; \theta). \quad (2.10)$$

As with a network change, we can interpret a behaviour change as a two stage process: firstly, an actor obtains an opportunity to alter their behaviour, and secondly, they choose, whether to increase or decrease their behaviour, or to leave in unchanged. We might assume that the time taken for the opportunity to arise is governed by the rate function $\lambda_i^{[Z]}(y, t; \theta)$, and the choice of what change (if any) to make, by the conditional choice probabilities $(p_{i,-1}^{[Z]}(y; \theta), p_{i,0}^{[Z]}(y; \theta), p_{i,1}^{[Z]}(y; \theta))$

Rate function

Often, the behaviour rate function is modelled as the same for all actors, and not dependent on the current state of the network (although in Chap-

ter 3 this will not be the case); assuming this implies that

$$\lambda_i^{[Z]}(y, t; \theta) = \rho^{[Z]}(t),$$

for some time dependent function $\rho^{[Z]}(t)$.

When we model the network and behaviours jointly, the function $\rho^{[Z]}(t)$ has an important restriction required for the computational tractability of the estimation procedure: when we simulate from these processes, we must know the relative frequencies with which to change the network and the behaviours. To achieve this, we can restrict the function so that, between observations, the rate of change to the behaviours is proportional to the rate of change of the network, with the constant of proportionality being estimated from the data. Specifically, recall from Section 2.1.3 that the rate function for a change to the network is denoted by $\rho^{[X]}(t)$; then between observations at times t_{m-1} and t_m , the ratio between the network and behaviour rate functions must be constant: for $m = 2, \dots, M$,

$$\frac{\rho^{[Z]}(t)}{\rho^{[X]}(t)} = c_m,$$

for some constant c_m . As with the network variable, between observations at times t_{m-1} and t_m , the number of opportunities that each actor has to change their behaviour is distributed as a Poisson random variable with mean

$$\rho_m^{[Z]} := \int_{t_{m-1}}^{t_m} \rho^{[Z]}(t) dt. \quad (2.11)$$

Conditional choice probabilities

Given the current state $Y(t) = y$, and given that actor i has an opportunity to change their behaviour, they must choose whether to increase or decrease its value, or to leave it unchanged. The choice is governed by the conditional choice probabilities $(p_{i,-1}^{[Z]}(y; \theta), p_{i,0}^{[Z]}(y; \theta), p_{i,1}^{[Z]}(y; \theta))$. We define $p_{i,0}^{[Z]}(y; \theta) = 1 - p_{i,-1}^{[Z]}(y; \theta) - p_{i,1}^{[Z]}(y; \theta)$. Similarly to the conditional choice probabilities for network changes, the inclusion of a potentially non-zero probability $p_{i,0}^{[Z]}(y; \theta)$ that the behaviour remains unchanged can be interpreted as accounting for the fact that actors satisfied with the state of behaviour will prefer to leave it unaltered.

We model the conditional choice probabilities in a multinomial logit form: given a change opportunity and the current state of the process $Y(t) = y$, actor i decreases their behaviour with 1 with probability

$$p_{i,-1}^{[Z]}(y; \theta) = \frac{\exp\{f_i^{[Z]}[(x, z(i \downarrow 1)); \theta]\}}{1 + \exp\{f_i^{[Z]}[(x, z(i \downarrow 1)); \theta]\} + \exp\{f_i^{[Z]}[(x, z(i \uparrow 1)); \theta]\}}, \quad (2.12)$$

and increases it with probability

$$p_{i,1}^{[Z]}(y; \theta) = \frac{\exp\{f_i^{[Z]}[(x, z(i \uparrow 1)); \theta]\}}{1 + \exp\{f_i^{[Z]}[(x, z(i \downarrow 1)); \theta]\} + \exp\{f_i^{[Z]}[(x, z(i \uparrow 1)); \theta]\}}, \quad (2.13)$$

where $f_i^{[Z]}(y; \theta)$ is referred to as the *behaviour objective function*. As with the network objective function, one interpretation of this is that actor i

prefers behaviours for which $f_i^{[Z]}(y; \theta)$ is high, and it is modelled as a weighted linear combination of effects. The objective function is modelled as a weighted linear combination of effects $s_{i1}^{[Z]}(y), \dots, s_{iK^{[Z]}}^{[Z]}(y)$, for some known $K^{[Z]}$, and has the form

$$f_i^{[X]}(y; \theta) = \sum_{k=1}^{K^{[Z]}} \beta_k^{[Z]} s_{ik}^{[Z]}(y). \quad (2.14)$$

The weights $\beta_1^{[Z]}, \dots, \beta_{K^{[Z]}}^{[Z]}$ are parameters to be estimated. Similarly to network effects, *behaviour effects* are functions of the current state of the process which are intended to encapsulate the different reasons an actor considers when making their behaviour change. Some example behaviour effects are (more are detailed in the Appendix of Snijders et al. (2010)):

1. *Behaviour shape*: the current value of the behaviour,

$$z_i.$$

2. *Quadratic shape*: the square of the current value of the behaviour,

$$z_i^2.$$

3. *Average similarity*: the average value of a measure of behaviour similarity between the actor and those to whom they have a tie,

$$\frac{\sum_{j=1}^n x_{ij} \left(1 - \frac{|z_i - z_j|}{\max_{i,j} \{|z_i - z_j|\}} - c \right)}{\sum_{j=1}^n x_{ij}},$$

where a centering constant c depending on the observed data (and given by Ripley et al. (2011)) is included to reduce correlations with the rate parameters.

Note that average similarity is an example of an effect which can be used for account for *influence*, because the more similar an actor's behaviour is to its peers, the higher the value of this effect is.

Selection network effects

In Section 2.1.3 we saw examples of network effects; now that we have a co-evolving behaviour variable, we may wish to incorporate network effects that also depend on the current state of the behaviour; this can allow us to account for the effect of the behaviour on the network. Some examples are:

1. *Behaviour ego*: the outdegree of actor i multiplied by the value of their behaviour,

$$z_i \sum_{j=1}^n x_{ij}.$$

2. *Behaviour alter*: the sum of the behaviours of actor i 's ties.

$$\sum_{j=1}^n x_{ij} z_j.$$

3. *Behaviour similarity*: the sum of the similarities between the behaviour

for actor i and the behaviour for each of actor i 's ties,

$$\sum_{j=1}^n x_{ij} \left(1 - \frac{|z_j - z_i|}{\max_{i,j} \{|z_j - z_i|\}} - c \right)$$

A centering constant c depending on the observed data is included to reduce correlations with the outdegree effect. The value of c is given by Ripley et al. (2011).

Note that behaviour similarity can be used to account for *homophilous selection*, because it is higher in value when an actor has more ties to others with behaviour that is similar to theirs.

2.2.4 Discussion of the model

In summary, we refer to the collection of all the unknown parameters as $\theta = (\rho, \beta)$, where $\rho = (\rho_2^{[X]}, \dots, \rho_M^{[X]}, \rho_2^{[Z]}, \dots, \rho_M^{[Z]})$ are those parameters associated with the rate functions, whilst $\beta = (\beta_1^{[X]}, \dots, \beta_{K[X]}^{[X]}, \beta_1^{[Z]}, \dots, \beta_{K[Z]}^{[Z]})$ are those associated with the network and behaviour effects. Then between observations at t_{m-1} and t_m , the continuous time Markov process $\{Y(t) : t_{m-1} \leq t < t_m\}$ on state space $\mathcal{X} \times \mathcal{B}^n$ has initial state $Y(t_{m-1}) = y(t_{m-1})$, where $y(t_{m-1})$ is the observation of the process at t_{m-1} , and then evolves according to transition intensity $Q_\theta(y, \hat{y}, t) = \tilde{Q}_\theta(y, \hat{y}) \left(\rho^{[X]}(t) / \rho_m^{[X]} \right)$,

for $\hat{y} \neq y$, where

$$\tilde{Q}(y, \hat{y}) = \begin{cases} \rho_m^{[X]} p_{ij}(y; \theta), & \text{if } \hat{y} = (x(i \rightsquigarrow j), z) \in \mathcal{A}(x) \times \{z\}, \\ \rho_m^{[Z]} p_{i,-1}^{[Z]}(y; \theta), & \text{if } \hat{y} = (x, z(i \downarrow 1)) \in \{x\} \times \mathcal{R}(z), \\ \rho_m^{[Z]} p_{i,1}^{[Z]}(y; \theta), & \text{if } \hat{y} = (x, z(i \uparrow 1)) \in \{x\} \times \mathcal{R}(z), \\ 0, & \text{otherwise,} \end{cases} \quad (2.15)$$

where $p_{ij}(y; \theta)$, $p_{i,-1}^{[Z]}(y; \theta)$ and $p_{i,1}^{[Z]}(y; \theta)$ are given by (2.5), (2.12) and (2.13), respectively.

As is discussed in detail by Steglich et al. (2010), modelling the dynamics of the network by assuming an explicit statistical model where the network evolves with an actor-level variable (here, the behaviour variable) gives a reasonably realistic model of the unobserved underlying continuous-time process, given incomplete, longitudinal data (which for practical purposes, is the most likely available kind). Assuming a static network may wrongly attribute too much importance to social influence, while, as explained by Steglich et al. (2010), modelling a dynamic network but failing to control for the dependencies between network dyads by neglecting mechanisms such as reciprocity and transitivity will lead to inaccurate conclusions. After controlling for these important actor tendencies, we can estimate the effects of selection and influence, which may both be of interest. In our model, we simultaneously control for these distinct mechanisms by including a network effect which can account for homophilous peer selection, such as the behaviour similarity effect, and also including

behaviour effects which are interpretable as influence effects, such as the average similarity effect.

2.3 Parameter estimation

We do not observe the times of ministeps, instead only observing the state of the network and behaviour variables at a few time points $t_1 < \dots < t_M$. As a result of this, the likelihood for this model cannot in general be written in a closed form expression, which makes maximum likelihood estimation of θ complicated and often computationally very expensive. We will consider maximum likelihood estimation in Parts III and IV, but the fastest, and hence most commonly used, estimation technique is to employ the Method of Moments (reviewed by Bowman and Shenton (1985)). This method involves choosing statistics of the process and finding the parameter value such that the expected values of these statistics is equal to the observed values; this parameter value (which we hope both exists and is unique) is then our method of moments estimate, which we denote by $\hat{\theta}^{\text{MoM}}$.

For $m \in \{2, \dots, M\}$, we choose an appropriate vector statistic

$$G_m(Y(t_{m-1}), Y(t_m))$$

with the same dimension as the vector of parameters θ . We then estimate

θ by solving

$$\sum_{m=2}^M \mathbb{E}_{\theta} (G_m(Y(t_{m-1}), Y(t_m)) | Y_{m-1} = y_{m-1}) = \sum_{m=2}^M G_m(y(t_{m-1}), y(t_m)). \quad (2.16)$$

We choose the statistics so that the j th component is primarily associated with estimating the j th component of θ , for $j \in \{1, \dots, \dim(\theta)\}$.

Considering the components of θ associated with the rate functions, for $m \in \{2, \dots, M\}$, we use $\sum_{i,j} |x_{ij}(t_m) - x_{ij}(t_{m-1})|$ as the statistic associated with $\rho_m^{[X]}$, and $\sum_i |z_i(t_m) - z_i(t_{m-1})|$ as the statistic associated with $\rho_m^{[Z]}$.

For any component of θ associated with an effect, for $m \in \{2, \dots, M\}$, we simply use the value of that effect at time t_m summed over all actors.

Given a parameter estimate $\tilde{\theta}$, we can simulate data $\tilde{y} = (\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_M)$, by setting $\tilde{y}_1 = y(t_1)$, and for $m \in \{2, \dots, M\}$, starting at time t_{m-1} , and in initial state \tilde{y}_{m-1} , and simulating forward according to the specified model with parameter $\theta = \tilde{\theta}$ until time t_m , at which point we set \tilde{y}_m as the current state of the process. We denote that \tilde{y} is simulated using this method by $\tilde{y} \sim \tilde{\theta}$.

2.3.1 Robbins-Monro stochastic approximation algorithm

We approximate the solution to (2.16) using a variation of the Robbins-Monro algorithm (Robbins and Monro, 1951; Snijders, 2001). The algorithm has three phases: Phase 1 is a short preliminary phase, used to gen-

erate a matrix used in the iterative formula used in Phase 2; Phase 2 contains the iterations used to actually estimate the solution; Phase 3 is used to estimate standard errors and check convergence of the algorithm.

Phase 1

In this phase, we simulate from an initial guess for the parameter estimate and use the simulations to construct an estimate \hat{D} of

$$D_{\hat{\theta}^{\text{MoM}}} = \frac{\partial}{\partial \theta} \mathbb{E}_{\theta} (G) \Big|_{\theta = \hat{\theta}^{\text{MoM}}}, \quad (2.17)$$

where $G = \sum_{m=2}^M G_m(\tilde{Y}_{m-1}, \tilde{Y}_m)$, and $\tilde{Y} \sim \theta$. We will use \hat{D} in the next phase. We also use \hat{D} to make one update to our parameter estimate, which will then be used as our initial value in the next phase.

Phase 2

This phase contains several subphases, the default number of which is 4.

In Subphase L , we start with an initial parameter value $\theta^{(1)}$ (obtained from the previous subphase if $L > 1$, or Phase 1, otherwise), and update using the iterative formula

$$\theta^{(k+1)} = \theta^{(k)} - a_L \hat{D}^{-1} \sum_{m=2}^M \left[G_m \left(\tilde{y}_{m-1}^{(k)}, \tilde{y}_m^{(k)} \right) - G_m \left(y(t_{m-1}), y(t_m) \right) \right],$$

where $\tilde{y}^{(k)} \sim \theta^{(k)}$, for $k = 1, \dots, K$, for some integer K , and a_L decreases with each subsequent subphase. We then use an average of the sequence

$\theta^{(1)}, \dots, \theta^{(k+1)}$ as our final parameter estimate for this subphase. If we are not in the final subphase, this estimate is then the initial value in the next; otherwise this is our final parameter estimate.

Phase 3

In Phase 3, we simulate using our final parameter estimate, in order to estimate standard errors (as well as checking the convergence of the algorithm). As described by Snijders (2001), we use the delta method (Bishop et al., 1975) and the implicit function theorem to estimate the covariance matrix by

$$\text{cov}(\hat{\theta}^{\text{MoM}}) \approx \left(D_{\hat{\theta}^{\text{MoM}}}^{-1} \right)^T \Sigma_{\hat{\theta}^{\text{MoM}}} D_{\hat{\theta}^{\text{MoM}}}^{-1},$$

where $D_{\hat{\theta}^{\text{MoM}}}$ is given by (2.17), and

$$\Sigma_{\hat{\theta}^{\text{MoM}}} = \text{Cov}_{\hat{\theta}^{\text{MoM}}}(G),$$

where $G = \sum_{m=2}^M G_m(\tilde{Y}_{m-1}, \tilde{Y}_m)$, and $\tilde{Y} \sim \hat{\theta}^{\text{MoM}}$.

We construct Monte Carlo estimates of $D_{\hat{\theta}^{\text{MoM}}}$ and $\Sigma_{\hat{\theta}^{\text{MoM}}}$ by simulating evolutions of the process according to $\hat{\theta}^{\text{MoM}}$.

The entire estimation procedure is implemented in the R package `RSiena` (Ripley et al., 2011).

Part II

Diffusion of innovations

Chapter 3

Diffusion of innovations in dynamic networks

This Chapter is based on work by Greenan (2015).

3.1 Introduction

The spread of a new idea, belief or practice through a network of social actors has long been a topic of interest in the social sciences. Examples range from the initiation of recycling amongst Californian households to the use of hybrid corn by Iowa farmers (Rogers, 1995). Early diffusion models assume social actor homogeneity, where the chance of adopting the innovation is the same for each individual in the network, and the interest lies in how the proportion of those who have adopted the new behaviour varies over time (Bartholomew, 1967); however, these population

models are inadequate for many research questions, ignoring the different tendencies of and influences on the social actors, and so models specifying individual rates of adoption are often required. This amounts to an event history analysis (Allison, 1984), where the hazard of adopting the innovation varies between actors, according to specified explanatory variables.

Sociologists and statisticians alike have asserted the importance of social contagion in such analyses, regarding the time taken to adopt as depending not only on individual characteristics, such as age or gender, but also on the social network to which an actor belongs, and their place within it (Coleman et al. (1966), Valente (1995)). For example, those with many connections in the network may be more likely to adopt the innovation than those who are isolated, due to increased interactions with those who have already adopted. Furthermore, once someone with many connections has adopted, many others may quickly follow, guided by this popular individual.

In existing diffusion studies (such as those by Strang and Tuma (1993), Myers (2000) and Behrman et al. (2002)) it is assumed that the social network in question is static, and does not change over time; however, this is often unrealistic, especially given the time-scale of the process that will usually be required to allow the innovation to diffuse adequately. Over such a time, friendships form and dissolve, trust can be built and broken, and those who were at one time popular and influential may not always be so. For many applications, it is an empirical fact that networks are not constant. In order to encapsulate network changes, we model the dynam-

ics of the social network as evolving simultaneously with the diffusion of the innovation.

In this chapter, we describe a modification of the model for the co-evolution of a network and an individual-level variable described in Chapter 2. Our model inherits the formulation that means that at every point in time the network and adoption processes depend on one another, which allows us to simultaneously account for influence and selection. The new parametric form proposed in this paper provides us with a proportional hazards model for the adoption process. As will be described later in more detail, this means that at any given point in time, conditional on the current state of the dynamic network, the model for the adoption process follows a standard and well-established survival analysis model. It also means that we can easily quantitatively interpret the model parameters.

In Section 3.2, we describe the notation that will be employed, before explaining the model in detail in Section 3.3; we also detail possible explanatory variables representing various forms of social contagion. The method of parameter estimation is described in Section 3.4, along with a theorem about its asymptotic performance. In Section 3.5, the method is applied to a real dataset which documents the initiation of cannabis smoking amongst adolescents in a Glaswegian school in the 1990s. The presence of social contagion is shown in this case. In Section 3.6, the performance of the estimator is examined with a simulation study.

3.2 Notation and data structure

An adoption of an innovation is modelled as a time dependent binary random variable for which $\{1\}$ is an absorbing state indicating adoption: for actor i , the *adoption indicator* is a random variable $Z_i(t)$ defined so that if T_i is the time of adoption of the innovation, then $Z_i(t) = 1$ if and only if $t > T_i$. This is a special case of a *behaviour variable* described in Chapter 2, where $Z_i(t)$ can take on any predetermined finite number of integer values and can both increase and decrease over time. Collectively, the vector of adoption indicators $Z(t) = (Z_1(t), \dots, Z_n(t))$ is referred to as the *adoption variable* or *adoption process*, and, as with the behaviour variable, its observation at time t is denoted by $z(t)$.

3.3 Model for the simultaneous diffusion of an innovation and evolution of a dynamic network

In this section, we describe the model that we will use for the simultaneous diffusion of an innovation and evolution of a dynamic network. After describing the model in detail, we will explain the benefits of this model for the diffusion process over that described in Chapter 2.

We make many of the same model assumptions as in Chapter 2; between two observation times t_{m-1} and t_m , the process $\{Y(t) : t_{m-1} \leq t < t_m\}$, is a

continuous time Markov chain with initial state given by the observation $Y(t_{m-1}) = y(t_{m-1})$. The state space of this process is $\mathcal{X} \times \{0, 1\}^n$, where \mathcal{X} is the set of binary adjacency matrices and $\{0, 1\}^n$ is the set of binary n -tuples. Given two distinct states $y, \hat{y} \in \mathcal{X} \times \{0, 1\}^n$, we denote the transition intensity for moving from y to \hat{y} at time t by $Q_\theta(y, \hat{y}, t)$.

As in Chapter 2, given the current state $Y(t) = y = (x, z)$, we assume that the process updates via mini-steps, where the process can change only to a member of the set

$$\mathcal{M}(y) = \{\mathcal{A}(x) \times \{z\}\} \cup \{\{x\} \times \mathcal{R}(z)\},$$

where $\mathcal{A}(x)$ is defined in (2.1). Now the only change that can be made to the adoption variable is that a single actor may adopt the innovations, and so now

$$\mathcal{R}(z) = \bigcup_{i: z_i=0} \{z(i \uparrow 1)\},$$

where $z(i \uparrow 1) \in \{0, 1\}^n$ and

$$z(i \uparrow 1)_j = \begin{cases} 1 & \text{for } j = i, \\ z_j & \text{for } j \neq i. \end{cases}$$

As before, if $\hat{y} \in \mathcal{X} \times \{0, 1\}^n$ is not a member of the set $\mathcal{M}(y) \cup \{y\}$ then the transition intensity $Q_\theta(y, \hat{y}, t)$ is zero.

Given the current state $Y(t) = y$, the model for the transition intensity $Q_\theta(y, \hat{y}, t)$ where $\hat{y} \in \mathcal{A}(x) \times \{z\}$ (i.e. where is the network that changes) is

the same as described earlier, in Section 2.1.3. In Section 3.3.1, we describe the model for the components of the transition intensity matrix where the change is to the adoption process.

3.3.1 Adoption process

We consider modelling the transition intensity of a single actor adopting the innovation. Examining diffusion where the risk of adopting the innovation has actor heterogeneity is most easily done by assuming an event history model; this is done, amongst others, by Strang (1991), Strang and Tuma (1993), and Iyengar et al. (2011).

Denote by $\mathcal{R}(t) \subseteq \{1, \dots, n\}$ the *risk set* at time $t \in \mathcal{T}$; this is the set of actors who are at risk of adopting the innovation. In other words, it is those $i \in \{1, \dots, n\}$ for which $Z_i(t) = 0$. The *hazard function* for actor $i \in \mathcal{R}(t)$ is defined as the event rate at time t conditional on survival until time t :

$$h_i(t; \theta) = \lim_{\Delta t \downarrow 0} \frac{\mathbb{P}_\theta(t < T_i \leq t + \Delta t \mid T_i > t, \mathcal{H}(t))}{\Delta t}, \quad (3.1)$$

where T_i is the time of adoption. The hazard is defined as zero for those $i \notin \mathcal{R}(t)$. The hazard is conditional upon $\mathcal{H}(t) = \{Y(s) : t_1 \leq s \leq t\}$, the collection of all information about the process up until time t , which is referred to as the *history* of the process until time t (Andersen, 1993). Our models use a Markov assumption and so the only relevant part of the history is the state of the process at time t ; to make this explicit, we add

the current state of the process $y = y(t)$ as an argument of the hazard, so that it is defined as

$$h_i(t; y, \theta) = \lim_{\Delta t \downarrow 0} \frac{\mathbb{P}_\theta(t < T_i \leq t + \Delta t \mid T_i > t, Y(t) = y)}{\Delta t}. \quad (3.2)$$

The hazard function can equivalently be expressed in terms of the adoption process as

$$h_i(t; y, \theta) = \lim_{\Delta t \downarrow 0} \frac{\mathbb{P}_\theta(Z_i(t + \Delta t) = 1 \mid Z_i(t) = 0, Y(t) = y)}{\Delta t}. \quad (3.3)$$

The hazard is the instantaneous transition intensity, and is a fundamental function in event history data analysis, since it measures the risk of adopting the behaviour at each point in time. The hazard is related to our Markov model in the following way: given the current state of the process $Y(t) = y$, the process updates to $\hat{y} = (x, z(i \uparrow 1))$, where $z(i \uparrow 1) \in \mathcal{R}(z)$, by actor i adopting the innovation, with transition intensity

$$Q_\theta(y, (x, z(i \uparrow 1)), t) = h_i(t; y, \theta). \quad (3.4)$$

Model for the hazard

The most popular event history model is the Cox regression model (Cox, 1972), where the hazard function for actor $i \in \mathcal{R}(t)$ is of the form

$$h_i(t; y, \theta) = h_0(t) \exp \left\{ \sum_{k=1}^K \alpha_k a_{ik}(y) \right\}, \quad (3.5)$$

where $a_{i1}(y), \dots, a_{iK}(y)$ are time-dependent explanatory variables (although time is suppressed in the notation, as these variables depend only the current state of the process $y = y(t)$), and the parameters $(\alpha_1, \dots, \alpha_K)$ are to be estimated. For our model the explanatory variables are functions of the adoption process, the network, or any other measured actor attribute, and we refer to them as *effects*. It is a semi-parametric model, because no assumptions are made about the form of the baseline hazard $h_0(t)$. As is explained in detail by Tuma and Hannan (1984), it is conventional to assume that explanatory variables are incorporated in the hazard as an exponentiated weighted linear combination, as in equation (3.5). As long as the baseline hazard is non-negative, this ensures that the hazard is kept non-negative for all possible values that the variables or parameters might take. It also means that the model belongs to a class commonly referred to as the *proportional hazards models* (O'Quigley, 2008). These models have the property that if a covariate is increased by some predetermined unit, then the effect on the hazard is multiplicative; for example, considering equation (3.5), increasing a_{i1} by 1 causes the hazard to increase by a factor of $\exp(\alpha_1)$. This means that if two actors differ only by some time-independent covariate, for example gender, then their two hazards will be proportional, and the ratio of the two will be constant over time. This is attractive because it allows us to easily compare the two actors' hazards, consequently enabling us to evaluate the effect of a particular explanatory variable, in this example gender.

We will employ the Cox regression model, and can estimate the param-

eters associated with the explanatory variables whilst leaving the baseline hazard unspecified, apart from its relationship with the network rate function, which we describe below. The use of an exponential hazard model for event history analysis in the social sciences is well established; Blossfeld (2002) describes it as “one of the most useful models for empirical research.” Later, Blossfeld (2005) uses this model for a variety of applications for event history analyses relating to youth behaviour, such as transition into cohabitation or parenthood.

Moreover, proportional hazards models have previously been employed to analyse diffusions through networks, in a very similar application to ours (albeit with static networks): Strang and Tuma (1993) and Strang (1991) examine actor heterogeneity in diffusion with the use of an individual-level exponential model, where the baseline hazard is constant; the latter refers to this as the *epidemic model of diffusion*.

Baseline hazard

We describe our hazard model as semi-parametric because the baseline hazard $h_0(t)$ is an unspecified function of time; however, due to the method of estimation we use, we require a restriction on the relationship between this and the network rate function: between observations, the ratio between them must be constant: for $m = 2, \dots, M$, if we define $\rho_m^{[Z]} = \int_{t_{m-1}}^{t_m} h_0(t) dt$, and $h_{0m}(t) = h_0(t)/\rho_m^{[Z]}$, then the network rate function is given by

$$\rho^{[X]}(t) = \rho_m^{[X]} h_{0m}(t).$$

An important special case is obtained if $h_{0m}(t) = 1/(t_m - t_{m-1})$ for all $m = 2, \dots, M$; the rates of network changes and adoptions are constant between observations, and the adoption process follows the piecewise constant exponential model (Blossfeld, 2002).

Adoption effects

We now present a description of possible adoption effects, which have been inspired by those used in analyses by Strang and Tuma (1993), Myers (2000) and Valente (2005). There are four types of effects that we will detail, and apart from the first, *intrinsic characteristics*, they can all be described as contagion effects, as they are intended to account for the impact of the prior adoptions by actors in the network on an individual's propensity to adopt. All of the contagion effects we include are what Strang and Tuma (1993) refer to as *social proximity measures*, which means they can be written in the form

$$a_{ik}(y) = \sum_{j=1}^n z_j d_{ijk}(y), \quad (3.6)$$

where $d_{ijk}(y)$ is a measure of closeness between actors i and j and the indicators $\{z_1, \dots, z_n\}$ mean that the effect sums only over those j which have adopted the innovation. If we interpret this 'closeness' as 'potential influence', then the effect measures the total amount of potential influence from current adopters that actor i is subjected to. Strang and Soule (1998) give a qualitative review of various mechanisms which potentially underlie the

process of a diffusion in a social network; several more contagion effects other than the ones we will now present could be constructed from these. Note that we define $0/0 = 0$.

1. *Intrinsic characteristics*: This type of effect can be described as an attribute which affects an actor's propensity to adopt the innovation, irrespective of the adoptions of others in the network. These can be covariates, such as age or gender, in which case the effect is simply the value of the covariate; for example

$$a_{i1}(y) = w_i, \quad (3.7)$$

where w_i is the value of the covariate for actor i . They could also be statistics of the network, such as indegree or outdegree (the number of incoming or outgoing ties, respectively, that an actor has with others in the network).

2. *Infection*: Myers (2000) describes this type of effect as a measure of "how influential the individual actor's adoption act is on everyone else in the system", and defines it formally as

$$\sum_{j=1}^n z_j s_j(y),$$

where $s_j(y)$ is a measure of the importance of actor j 's adoption (Myers refers to this as a measure of the adoption's "severity"). In this formulation, an adoption by an actor affects all others in the network

equally, which for many networks may not be a reasonable assumption. Moreover, at any given time, this effect has equal value for all actors, which may lead to high correlation with the baseline hazard. Another type of effect can be constructed so that an adoption by an alter affects the ego actor only if there is tie from ego to alter. For this effect, we sum only over those actors to whom actor i has an outgoing tie, giving the infection effect the formula

$$a_{i2}(y) = \sum_{j=1}^n z_j x_{ij} s_j(y). \quad (3.8)$$

It is this effect that we consider later in our analysis of real data. Options for the importance measure $s_j(y)$ could be an actor's indegree or outdegree, or an actor covariate. Strang and Tuma (1993) choose to measure importance by the adopter's indegree (although, like Myers, they assume adoptions affect all actors in the network equally). Using indegree as the measure of importance with the formula given in equation (3.8), the infection effect measures the sum of the indegrees of the adopters to whom an actor is tied. A positive coefficient associated with this effect implies that individuals with high indegrees influence their alters' behaviour more than those with low indegrees; in other words, their adoptions are more highly infectious.

3. *Exposure*: According to Valente (2005), exposure "captures social influence conveyed through overt transmission of information, persuasion, or direct pressure," and can be measured either by the proportion of contacts that have adopted, or the total number. We refer

to the former definition as the *average exposure* effect, which has the form

$$a_{i3}(y) = \frac{\sum_{j=1}^n z_j x_{ij}}{\sum_{j=1}^n x_{ij}}, \quad (3.9)$$

and to the latter as the *total exposure* effect, which can be written as

$$a_{i4}(y) = \sum_{j=1}^n z_j x_{ij}. \quad (3.10)$$

A positive coefficient associated with these effects means that being tied to adopters increases the chance of adopting.

4. *Susceptibility*: This type of effect conveys “how responsive an individual actor is when an adoption act occurs,” according to Myers (2000), and is intended to address the question of ‘whether some [actors] are more likely to react to the behavior of others, irrespective of their inherent propensity to [adopt]’. For actor i , Myers (2000) operationalises this as a measure of ‘responsiveness’ $p_i(y)$ multiplied by the total number of adopters currently in the network:

$$p_i(y) \sum_{j=1}^n z_j.$$

However, as with the infection effect, we argue that often it may be more reasonable to assume that adoptions only affect those from

whom the adopter has a tie, and so propose instead the formula

$$a_{i5}(y) = p_i(y) \sum_{j=1}^n z_j x_{ij}. \quad (3.11)$$

This effect can be considered an interaction between the responsiveness measure $p_i(y)$ and the total exposure (see equation (3.10)); we can then think of this effect as measuring an actor's *susceptibility to total exposure*. This suggests a further effect, *susceptibility to average exposure*, which has the formula

$$a_{i6}(y) = p_i(y) \frac{\sum_{j=1}^n z_j x_{ij}}{\sum_{j=1}^n x_{ij}}. \quad (3.12)$$

Strang and Tuma (1993) choose to measure 'responsiveness' $p_i(y)$ by an actor's indegree, reasoning that an actor with a high indegree "may be more susceptible to the adoptions of others, due to their wide circle of contacts." Alternatively, we may use outdegree (but not for susceptibility to average exposure, as this just gives the total exposure effect), or an actor covariate.

3.3.2 Discussion of the model

In summary, we refer to the collection of all the unknown parameters as $\theta = (\rho, \beta, \alpha)$, where $\rho = (\rho_2^{[X]}, \dots, \rho_M^{[X]}, \rho_2^{[Z]}, \dots, \rho_M^{[Z]})$ are those parameters associated with the baseline rates, whilst $\beta = (\beta_1^{[X]}, \dots, \beta_{K^{[X]}}^{[X]})$ are those associated with the network effects, and $\alpha = (\alpha_1, \dots, \alpha_K)$ are those re-

lated to the adoption effects. Between observations at t_{m-1} and t_m , the continuous time Markov process $\{Y(t) : t_{m-1} \leq t < t_m\}$ on state space $\mathcal{X} \times \{0, 1\}^n$ has initial state $Y(t_{m-1}) = y(t_{m-1})$, where $y(t_{m-1})$ is the observation of the process at t_{m-1} , and then evolves according to transition intensity $Q_\theta(y, \hat{y}, t) = h_{0m}(t)\tilde{Q}_\theta(y, \hat{y})$, for $\hat{y} \neq y$, where

$$\tilde{Q}(y, \hat{y}) = \begin{cases} \rho_m^{[X]} p_{ij}(y; \theta), & \text{if } \hat{y} = (x(i \rightsquigarrow j), z) \in \mathcal{A}(x) \times \{z\}, \\ \rho_m^{[Z]} \exp(\alpha^T a_i(y)), & \text{if } \hat{y} = (x, z(i \uparrow 1)) \in \{x\} \times \mathcal{R}(z), \\ 0, & \text{otherwise,} \end{cases} \quad (3.13)$$

where $p_{ij}(y; \theta)$ is given by (2.5).

We have assumed that the diffusion follows a Cox regression model, and have modelled the process by considering the hazard function. The model by Snijders et al. (2007) described in Chapter 2 for the co-evolution of a dynamic network with an actor integer-valued variable could be applied to the network and adoption variables described in this chapter; however, the model utilizing hazard functions proposed here is an improvement for our application in two ways. Firstly, the use of Cox regression models is well established in event history analysis, while the model proposed by Snijders et al. (2007) (where the actor variable changes with transition intensities modelled in a multinomial logit form) does not correspond to any known survival model. Secondly, the use of a proportional hazards model in this paper means that the parameters are easy to interpret; this is more difficult using the model described in Chapter 2.

3.4 Parameter estimation

In Section 2.3, we described how we estimate θ , the vector of parameters of the model. Our method of moments estimation procedure requires choosing vector valued moment statistics

$$G_2(Y(t_1), Y(t_2)), \dots, G_M(Y(t_{M-1}), Y(t_M)),$$

with dimension equal to the dimension of θ , such that each component is primarily associated with a component of θ . In Section 2.3, we describe choices of the components associated with rate parameter and network effect parameters. Now we have additional parameters $\alpha = (\alpha_1, \dots, \alpha_K)$, and for $k \in \{1, \dots, K\}$, and $m = \{2, \dots, M\}$, we choose the component of $G_m(Y(t_{m-1}), Y(t_m))$ associated with α_k to be

$$\sum_{i \in \mathcal{R}(t_{m-1})} a_{ik}(y(t_{m-1})) z_i(t_m). \quad (3.14)$$

3.4.1 Asymptotic behaviour of estimates of α

In this section, we consider the behaviour of the components of the moment statistics associated with estimating α . We assume that it is the first K components of the moments statistics that are associated with α , and

define $(g_1(\theta; Y), \dots, g_K(\theta; Y))$ such that, for $k = 1, \dots, K$,

$$g_k(\theta; Y) = \sum_{m=2}^M G_m(y(t_{m-1}), y(t_m)) - \mathbb{E}_\theta (G_m(Y(t_{m-1}), Y(t_m)) | Y_{m-1} = y_{m-1}). \quad (3.15)$$

Assessing the performance of the method of moments estimator suggested by Snijders (2001) is difficult. However, conditional on the outcome of the process, if we increase the frequency of the observations of the process, then the method of moments estimate of α converges to the maximum likelihood estimate obtained by observing the process continuously in time. To see this, consider a process $Y = \{Y(t) = (X(t), Z(t)) : 0 \leq t \leq 1\}$, which evolves according to the model described in Section 3.3. Suppose that we can observe the process continuously in time, giving the observation $Y = y$. Denote by α the vector of parameters associated with the adoption effects. Then we may obtain a maximum likelihood estimate $\hat{\alpha}_{ML}$ for α based on this complete data. Suppose we instead observe the process only at $M + 1$ equally spaced timepoints $\{t_m^M = (m - 1)/M : m = 1, \dots, M + 1\}$, (the equal spacing is not necessary, but simplifies notation; it is just necessary the intervals between observations are all $O(1/M)$). We denote the observed data by $y^M = (y_1^M, \dots, y_m^M, \dots, y_{M+1}^M)$, where $y_m^M = y(t_m^M)$ is the observation of the state of the process at time t_m^M . We may then obtain a method of moments estimate $\hat{\alpha}_{MoM}^M$ for α based on the data y^M .

We compare the sequence of method of moments estimates $\{\hat{\alpha}_{MoM}^M\}_{M=1}^\infty$

of α with the maximum likelihood estimate $\hat{\alpha}_{ML}$ of α . We consider an outcome $Y = y$ for which

1. the baseline hazard $h_0(t)$ is a known function, with all derivatives bounded for $0 \leq t < 1$;
2. the remaining parameters $\{\theta \setminus \alpha\}$ are known;
3. the maximum likelihood estimate $\hat{\alpha}_{ML}$ of α based on continuous observation $Y = y$ is unique and finite.

Theorem 3.16. *Given the outcome $Y = y$, suppose that there is a increasing integer sequence M_1, M_2, \dots such that the subsequence of method of moments estimates $\{\hat{\alpha}_{MoM}^{M_N}\}_{N=1}^{\infty}$ of α is bounded with each estimate in the subsequence the unique method of moments estimate for the corresponding data y^M . Then the corresponding subsequence of distances between method of moments estimates and $\hat{\alpha}_{ML}$ converges to zero: as $N \rightarrow \infty$,*

$$\|\hat{\alpha}_{MoM}^{M_N} - \hat{\alpha}_{ML}\| \rightarrow 0.$$

Proof. At time t , the transition intensity matrix of the process is given by $Q(t) = h_0(t)\tilde{Q}$, where $h_0(t)$ is a scalar function and the matrix \tilde{Q} does not vary with time (its entries are given in Section 3.3.2). The start and end of the process at 0 and 1 respectively are chosen without loss of generality.

Firstly, consider the score when the process is observed continuously, giving data y . Denote the total number of changes to the process by C

and the times of these changes by s_1, s_2, \dots, s_C . Define $s_{C+1} = 1$, denote $y_c = y(s_c)$ for $c = 1, \dots, C+1$, and let y_0 be the initial state of the process.

Then the likelihood for α is given by

$$L_\theta(\alpha, y) = \left[\prod_{c=1}^C Q(s_{c-1})_{y_{c-1}, y_c} \exp \left\{ \int_{s_{c-1}}^{s_c} Q(t)_{y_{c-1}, y_{c-1}} dt \right\} \right] \exp \left\{ \int_{s_C}^1 Q(t)_{y_C, y_C} dt \right\}.$$

Therefore, for $k = 1, \dots, K$, the k th component of the score is given by

$$U_k(\alpha; y) = \left(\sum_{c=1}^C \frac{1}{\tilde{Q}_{y_{c-1}, y_c}} \frac{\partial \tilde{Q}_{y_{c-1}, y_c}}{\partial \alpha_k} \right) + \left(\sum_{c=1}^{C+1} \int_{s_{c-1}}^{s_c} \frac{\partial Q(t)_{y_{c-1}, y_{c-1}}}{\partial \alpha_k} dt \right).$$

For $c = 1, \dots, C$, if the difference between y_c and y_{c-1} is a network change, then \tilde{Q}_{y_{c-1}, y_c} is independent of α and

$$\frac{\partial \tilde{Q}_{y_{c-1}, y_c}}{\partial \alpha_k} = 0. \quad (3.17)$$

If instead the difference is an adoption of the innovation, by actor i , then

$\tilde{Q}_{y_{c-1}, y_c} \propto \exp(\alpha^T a_i(y_{c-1}))$ and

$$\frac{1}{\tilde{Q}_{y_{c-1}, y_c}} \frac{\partial \tilde{Q}_{y_{c-1}, y_c}}{\partial \alpha_k} = a_{ik}(y(s_{c-1})). \quad (3.18)$$

Note that, for $c = 1, \dots, C+1$,

$$Q(t)_{y_{c-1}, y_{c-1}} = - \sum_{\hat{y} \in \mathcal{M}(y_{c-1})} Q(t)_{y_{c-1}, \hat{y}},$$

and so

$$\begin{aligned}
\frac{\partial Q(t)_{y_{c-1}, y_{c-1}}}{\partial \alpha_k} &= - \sum_{\hat{y} \in \mathcal{M}(y_{c-1})} \frac{\partial Q(t)_{y_{c-1}, \hat{y}}}{\partial \alpha_k}, \\
&= - \sum_{i: z_{c-1, i} = 0} \frac{\partial h_i(t)}{\partial \alpha_k}, \\
&= - \sum_{i: z_{c-1, i} = 0} a_{ik}(y(t)) h_i(t), \\
&= - \sum_{i: z_{c-1, i} = 0} a_{ik}(y(t)) h_i(t), \tag{3.19}
\end{aligned}$$

where $h_i(t) = h_0(t) \exp(\alpha^T a_i(y(t)))$; for simplicity, we suppress the dependence on α in the notation. Therefore, equations (3.17), (3.18) and (3.19) imply that

$$U_k(\alpha; y) = \sum_{i \in I} a_{ik}(y(T_i)) - \sum_{i=1}^n \int_0^1 (1 - z_i(t)) a_{ik}(y(t)) h_i(t) dt. \tag{3.20}$$

where I is the set of actors that adopt in $(0, 1]$, and T_i is the adoption time for actor i .

Next, we consider the moment statistic that we use to estimate α . Let A be a compact set containing $\hat{\alpha}_{ML}$ and the bounded subsequence of method of moments estimates. Recall (Norris, 1997) that, for small Δt , uniformly for $\alpha \in A$,

$$\mathbb{P}_\theta(Y(t + \Delta t) = \hat{y} | Y(t) = y') = 1\{\hat{y} = y'\} + Q(t)_{y', \hat{y}} \Delta t + O(\Delta t^2). \tag{3.21}$$

This approximation is uniform on A as $Q(t)$ is bounded and has

bounded derivative with respect to each component of α (the latter is true as each element of $Q(t)$ is of the form $Ae^{\alpha_k B}$ for bounded constants A and B , for $k = 1, \dots, K$).

Let $R_m^M = \mathcal{R}(t_{m-1}^M)$ be the set of actors at risk of adopting the innovation at time t_{m-1}^M . To simplify notation, let $a_{ikm}^M = a_{ik}(y_{m-1}^M)$, $z_{mi}^M = z_i(y_{m-1}^M)$ and $Z_{mi}^M = Z_i(y_{m-1}^M)$. Recall that, for $k = 1, \dots, K$, the component of the moment statistics used for method of moments associated with α_k is given by

$$\begin{aligned}
g_k^M(\alpha; y^M) &= \sum_{m=2}^{M+1} \sum_{i \in R_m^M} a_{ikm}^M (z_{mi}^M - \mathbb{E}_\theta(Z_{mi}^M | Y_{m-1}^M = y_{m-1}^M)), \\
&= \sum_{m=2}^{M+1} \sum_{i \in R_m^M} a_{ikm}^M \left\{ z_{mi}^M - \sum_{\hat{y} \in \mathcal{Y}_i} \mathbb{P}_\theta(Y_m^M = \hat{y} | Y_{m-1}^M = y_{m-1}^M) \right\}, \\
&= \left[\sum_{m=2}^{M+1} \sum_{i \in R_m^M} a_{ikm}^M \left\{ z_{mi}^M - \mathbb{P}_\theta(Y_m^M = y_{m-1}^M(i \uparrow 1) | Y_{m-1}^M = y_{m-1}^M) \right\} \right] + \\
&\hspace{20em} O(1/M), \\
&= \left[\sum_{m=2}^{M+1} \sum_{i \in R_m^M} a_{ikm}^M \left\{ z_{mi}^M - h_i(t_{m-1})/M \right\} \right] + O(1/M), \quad (3.22)
\end{aligned}$$

where \mathcal{Y}_i is the set of states such that $z_i = 1$, and where $y_{m-1}^M(i \uparrow 1)$ is the same as y_{m-1}^M but with actor i having adopted the innovation. Equation (3.22) is obtained using the approximation given by (3.21). Then, by the continuity and boundedness of the transition intensity matrix Q and its

derivatives, uniformly on A ,

$$g_k^M(\alpha; y^M) \longrightarrow \sum_{i \in I} a_{ik}(y(T_i)) - \sum_{i=1}^n \int_0^1 (1 - z_i(t)) a_{ik}(y(t)) h_i(t) dt = U_k(\alpha, y).$$

Therefore, for all $k = 1, \dots, K$, the difference $|g_k^M(\alpha, y^M) - U_k(\alpha, y)|$ between the method of moments statistic and the score based on y converges to zero uniformly on A , as $M \rightarrow \infty$. Therefore

$$\|g^M(\alpha, y^M) - U(\alpha, y)\| \longrightarrow 0, \quad (3.23)$$

as $M \rightarrow \infty$, uniformly on A .

Let $\epsilon > 0$ be sufficiently small that there exists $\alpha \in A$ with $\|\alpha - \hat{\alpha}_{ML}\| \geq \epsilon$. Then, by the uniqueness of the root of the score U and the compactness of A , there exists $\delta_\epsilon > 0$ such that if $\alpha \in A$ and $\|\alpha - \hat{\alpha}_{ML}\| \geq \epsilon$ then $\|U(\alpha, y)\| \geq \delta_\epsilon$. By the uniform convergence in (3.23), there exists N_ϵ such that $\|U(\hat{\alpha}_{MoM}^{M_N}, y)\| < \delta_\epsilon$ for all $N > N_\epsilon$. Therefore $\|\hat{\alpha}_{MoM}^{M_N} - \hat{\alpha}_{ML}\| < \epsilon$ for all $N > N_\epsilon$. So $\|\hat{\alpha}_{MoM}^{M_N} - \hat{\alpha}_{ML}\| \rightarrow 0$ as $N \rightarrow \infty$. \square

Consequences of theorem

In practice, neither the network parameters nor the baseline hazard are known; these (or in the case of the baseline hazard, its integrals between observations times) must be estimated simultaneously to the adoption parameters. However, if we assume that we can estimate these unknown quantities well, then the theorem indicates that taking frequent observa-

tions of the process should give an estimate of the adoption parameters that is close to the maximum likelihood estimate which could be obtained if the process were observed continuously.

The convergence shown in the theorem is conditional upon the particular outcome of the process, and so we cannot infer that the method of moments estimator is asymptotically optimal amongst those that could be obtained using any method of moment equations (e.g. according to the definition of optimality given by Godambe (1960)); however, the theorem suggests that, for a real dataset that features frequent observations of the process, the parameter estimate obtained using these estimating equations will be a good choice, by virtue of the properties of the maximum likelihood estimator. In this way, the theorem supports the choice of the moment statistics that are associated with the estimation of the adoption effects.

3.5 Example: Glasgow school data

In this section, we apply our model to a real dataset. We analyse data collected as part of the *Teenage Friends and Lifestyle Study* (West and Sweeting, 1996): the friendship ties between 129 pupils at a school in Glasgow are recorded at three yearly observations, beginning when the students were between 12 and 13 years old. The innovation of interest is cannabis smoking, and so an actor becomes an adopter when they first smoke cannabis; for simplicity, we often refer to ‘smoking’ and ‘smokers’, omitting the

specification of cannabis, and also that someone may have only smoked cannabis once, and are not current users. Initially there were 35 pupils who had smoked cannabis; by the second observation this had risen to 47, and by the third, to 71. Covariates that are included in the analyses are whether they are female (which is encoded as 1 if they are, and 0 otherwise) and frequency of alcohol consumption (which was self-reported each year on a scale from 1 to 5, where 1 corresponds to ‘never’ and 5 to ‘more than once a week’). The network effects that are included were described in Section 2.1.3 with the addition of a gender similarity effect (which is defined like the smoking similarity effect); this selection has been inspired by the choices made by Snijders et al. (2007) and Steglich et al. (2010) in their analyses of the same dataset (where they model tobacco use, and use a different model to that presented here). We experimented with a variety of choices for the adoption process effects. Many of the contagion effects described in Section 3.3 were highly correlated when included jointly in the analysis; those that remain in the final model were chosen as they were not.

Table 3.1 shows the results of estimating the parameters of this model. For each of the adoption effects, the final column indicates which effect type is included, by referencing the equations given in Section 3.3.1. For each parameter estimate, we assess significance by approximately testing the null hypothesis that it is zero (the form of this t -type test, and the assumptions required, are described by Snijders et al. (2010)). Table 3.1 reports a guide to the p -values associated with this test for each parameter (except

Table 3.1: Results for the Glasgow school dataset.

	<i>Estimate</i>	<i>Std. error</i>	<i>Effect eq.</i>
<i>Network Rate Effects</i>			
Mean change opportunities (period 1)	11.29	(1.11)	
Mean change opportunities (period 2)	8.75	(0.81)	
<i>Network Evaluation Effects</i>			
Outdegree	-1.93***	(0.10)	
Reciprocity	2.02***	(0.11)	
Transitive triplets	0.24***	(0.03)	
Number of actors at distance 2	-0.43***	(0.04)	
Gender similarity	0.89***	(0.10)	
Smoking alter	-0.08	(0.09)	
Smoking ego	-0.08	(0.10)	
Smoking similarity	0.21*	(0.09)	
<i>Adoption effects</i>			
Integrated baseline hazard (period 1)	0.07	(0.07)	
Integrated baseline hazard (period 2)	0.14	(0.15)	
Average exposure	3.42*	(1.71)	(3.9)
Female	-1.12	(0.79)	(3.7)
Infection by 'female'	0.51	(0.42)	(3.8)
Alcohol intake	0.57**	(0.21)	(3.7)

† $p < 0.1$; * $p < 0.05$; ** $p < 0.01$; *** $p < 0.001$; all t ratios < 0.06 .

for the parameters associated with the baseline rates, as these are known to be strictly greater than zero, since there are changes in the data between observations). Also given is a maximum of t ratios which assess convergence of the estimation algorithm for all parameter estimates (see Ripley et al. (2011)): this is 0.06, which suggests that the convergence is good.

The smoking similarity effect is positive and significant; this suggests that the students have a predilection for homophily with respect to smoking. This mechanism could not have been discerned if a static network had been assumed. The significance of the exposure effect implies that the initiation of smoking is socially contagious: the higher the proportion of smokers amongst a non-smoking student's friends, the higher their hazard of beginning to smoke. Due to the proportional hazards property of our model, we can infer from our estimates that, for any actor, if their average exposure increases by $\delta \in [0, 1]$ then their hazard increases by a factor of approximately 31^δ (since $\exp(3.42) \approx 30.57$). Similarly, increasing alcohol intake by $\delta \in \{1, 2, 3, 4\}$ increased an actor's hazard by approximately 1.8^δ (since $\exp(0.57) \approx 1.77$). That these parameters can easily be interpreted in this way is an important advantage of using this model.

3.6 Simulation study

In Section 3.4, we showed the theoretical advantage in collecting data frequently; however, in many cases, a large number of observations will be impractical; for example, if the network is large then surveying ties will be

very time consuming. Therefore, in this section we conduct a simulation study using generated data of just three observations, as a very limited exploration of the performance of the method of moments estimator. We generate data and conduct an analysis, and then repeat 1000 times, collating the results to obtain Monte Carlo estimates of some properties of the estimator.

The data are generated to be similar to the real dataset considered in the previous section; this is achieved by taking the initial states of the network and adoption variable directly from the dataset, along with the values of the covariates for all three observations. Taking the covariates for all observations does not violate the Markov assumption, since these are assumed to be exogenous to the stochastic process. Two further observations of the network and the adoption variable are obtained by simulating the network according to a model similar to that estimated in the previous section: the effects and associated parameters are chosen to be similar to those obtained by the analysis in the previous section. Fewer effects are included than in the analysis in the previous section to reduce computation time and to provide a simpler model to estimate.

Table 3.2 shows the mean parameter estimate and the root mean squared error (RMSE) for the estimator. We also assess the estimator when used in conjunction with the hypothesis test described in the previous section. We assess the appropriateness of the test by calculating, for each analysis and each parameter, the deviation of the parameter estimate from the true value divided by the standard error estimate. Under our assumption re-

Table 3.2: Results for the simulation study with 129 actors.

	H_0	Mean	RMSE	α	$1 - \beta$
<i>Network Rate Effects</i>					
Mean change opportunities (period 1)	11.0	10.86	0.92	0.05	-
Mean change opportunities (period 2)	9.0	9.10	0.85	0.03	-
<i>Network Evaluation Effects</i>					
Outdegree	-2.8	-	-	-	-
Reciprocity	2.0	2.01	0.09	0.03	0.98
Transitive triplets	0.4	0.39	0.03	0.03	0.97
Gender similarity	0.9	0.91	0.10	0.03	0.96
Smoke similarity	0.2	0.21	0.10	0.02	0.38
<i>Adoption effects</i>					
Integrated baseline hazard (period 1)	0.05	0.06	0.08	0.09	-
Integrated baseline hazard (period 2)	0.05	0.05	0.07	0.08	-
Average Exposure	3.8	3.86	1.63	0.08	0.48
Alcohol intake	0.6	0.68	0.27	0.06	0.68

quired for the test, these test statistics should be approximately standard normally distributed, and so we compute the proportion of these statistics that exceed in absolute value the upper 97.5% quantile of the standard normal distribution. If our assumption of normality is reasonable, then we expect rejection rates (denoted by α) close to 5%. For the integrated baseline hazards $\{\rho_m^{[Z]} : m = 2, \dots, M\}$, we consider the logarithm of the parameter, as by inspection of the simulation results, this is less skewed and more normally distributed. We then estimate the power of the test (denoted by $1 - \beta$) when applied to each of the parameter estimators, by finding the proportion of simulations for which a incorrect null hypothesis that each parameter coordinate is zero is rejected.

Recall that to reduce correlations the similarity effect has a centering term which depends on each generated dataset. Note that each time this affects the estimate of the outdegree parameter (by translating it by the centering term) so the collated results for the outdegree effects are not meaningful and are hence excluded; we may think of the outdegree effect like a control variable.

Of the 1000 generated datasets, 834 are included in the analysis; the remainder were removed due to inadequate convergence of the algorithm (the t ratio that can be calculated to assess convergence of the algorithm is described by Ripley et al. (2011)); we shall discuss the excluded data later. As can be seen in Table 3.2, the parameters associated with network dynamics are estimated very well; the bias and root mean squared are both low in all cases, and for parameters except the ‘smoke similarity,’ the

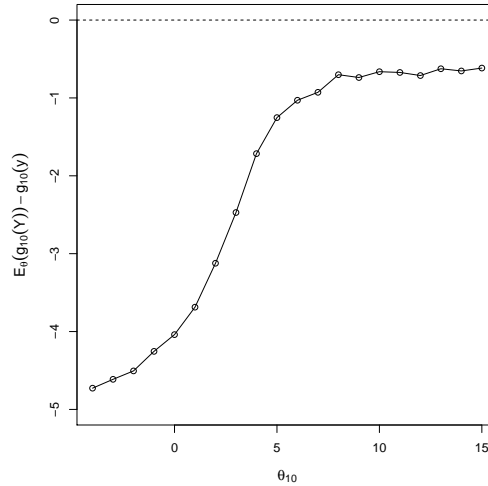


Figure 3.1: Deviation between observed and expected value of the moment statistic associated with the average exposure effect.

power is very high. The parameters associated with the adoption process also have quite small bias, and although the root mean squared error is quite high, the power is reasonable. It is inevitable that the adoption parameters will be more difficult to estimate than those associated with the network; the moment statistic associated with the latter are made up of a sum of $(M - 1)n(n - 1)$ statistics (Snijders, 2001), whilst the former has just $\sum_{m=2}^M R_m < (M - 1)n$ components, where R_m is the number of actors at risk of adoption at time t_{m-1} . This means that there is less information that can be used to estimate the adoption process parameters, and this is likely to be why the results show higher RMSE, and a lower power, for these.

A considerable number of sets of results (16.6%) were excluded due to unsatisfactory convergence of the algorithm; however, when this is investigated further, it does not appear to simply be due to limitations in the

stochastic approximation estimation procedure (which potentially could have been solved by running the algorithm for further iterations, to obtain more accurate Monte Carlo estimates). Instead, it appears likely that in these cases the method of moments equations do not have a solution. This is illustrated for one set of unconverged results in Figure 3.1, which shows the deviation between the observed and expected amounts for the moment statistics associated with the average exposure parameter; for a range of points around the true parameter ($\theta_{10} = 3.8$) the deviation is not zero (the other parameters are chosen so that the other components of the method of moments equations are zero). This suggests that the method of moments equation does not have a solution. The non-existence of a solution, and therefore the non-existence of the method of moments estimator, for some data sets is no surprise, as for regular Cox regression this occurs also for the maximum likelihood estimator; see Silvapulle and Burrige (1986).

3.7 Discussion

We have presented a method for modelling the simultaneous evolution of the diffusion of an innovation with a dynamic social network. Incorporating both of these processes allows us to create a more realistic model, and to include as mechanisms of the process both social influence and homophilous peer selection. The model has been constructed so that the adoption times follow a proportional hazards model, whilst the dynamics

of the network are assumed to follow the stochastic actor oriented model developed by Snijders (2001). In many applications, this model may be preferable to that which would be obtained using the model given by Snijders et al. (2007), as the latter does not correspond to any known survival model. The proportional hazards property of the model also enables easy interpretation of the model parameters.

To estimate the parameters in the model, we have used the method of moments estimator suggested by Snijders et al. (2007), and shown that for a given outcome, the estimate obtained is asymptotically equivalent to the maximum likelihood estimate, as the frequency of observations tends to infinity. We have also examined its performance when applied to data with only 3 observations, and shown that whilst the estimator appears approximately unbiased, there is sometimes the problematic lack of a solution to the method of moments equations. We suggest that as high a number of observations as possible is desirable.

We have also demonstrated the use of the model on a real dataset, involving a Glaswegian school and the initiation of cannabis smoking amongst its students, and using the model have identified both peer influence and selection based on similarity of cannabis use.

The model could be extended and altered in a variety of ways. The most obvious of these is the addition of a wider variety of adoption effects; the implementation of the estimation of the model allows a variety of functions of the network and the adoption variable to be added as effects.

A possible extension of the model could be achieved by increasing the size of the state space of the adoption variable: instead of being a binary random variable, it could be defined to take three values; this could then be used to construct a Knowledge Attitude Practice (KAP) model (Rogers (1995), Valente et al. (1998)), where the actors must first gain knowledge of the innovation before they can adopt it. The first state could correspond to unawareness of the innovation, the second awareness but not practice, and then finally, the third state adoption. The model could then include interactions between the value of adoption variable and the contagion effects, in order to have different dynamics governing the rates of change between the first and second states, and the second and third. Apart from the social sciences, this model could be applied to medical statistics; for example, we could formulate it as an SIR model (Kermack and McKendrick, 1927) with an endogenous dynamic network.

We could extend the state space even further to use the model to approximate an inhomogeneous Poisson process (Suhov and Kelbert, 2008). (Having an infinite state-space would give this process exactly, but the current implementation will not allow this; instead it can be made as large as is computationally feasible). This could be used to model situations where the behaviour of the actors does not consist of a single innovation, but a number of events. For example, instead of being interested in the time at which an actor first initiates a practice, the Poisson process models how many times they actually perform it.

Part III

Local network models

Chapter 4

Local network models

4.1 Introduction

We often interpret that changes to the dynamic social network are driven by the actors, with each actor in control of their own outgoing ties. It is therefore natural to assume that the local properties of the ego in the network will be important to this ‘driving’ actor. In this part, we will consider a class of models where the state of a subnetwork of which the ego actor is the focal node is actually their *only* consideration.

When constructing a Siena network model, we select effects which we hope encapsulate the tendencies of the ego for extending, maintaining and breaking network ties; the ego can then stochastically maximise these features of the network, by altering their outgoing ties. In this part, we consider the situation where these tendencies depend only on the config-

urations of ties to and from actors who are already local to the ego, where 'local' is defined with respect to geodesic distance. This essentially will just place a restriction on which network effects we may include in our model; however, as we will see, doing this will have computational and theoretical implications.

A sufficient condition for the plausibility of this model is that an actor has knowledge of the ties only to and from actors that are 'local' to them. For a large network, it may be reasonable to assume that each actor has only partial knowledge about the state of the network at any given point in time, and it may be reasonable to further their knowledge will extend only to those actors to whom they are closest, with respect to geodesic distance. However, for a small network this may be quite a strong assumption, and a relaxed alternative would be to say that although an actor may be aware of further network connections, involving non-local actors, they are unimportant to them when they consider which network ties to extend or break.

As we will see, the size of the subgroup of actors deemed local can be kept relatively small, irrespective of the size of the network, while still enabling us to include many important mechanisms, such as transitivity, in our model.

One consequence of presuming this class of model is that there is no place for a global hierarchy; ego actors will not be aware of the global structure of the network (or, at least, will not find it important), and so, since the ego drives the mechanisms, they cannot be incorporated in the model. Inevitably, this means that this type of model will be unsuitable for some

research questions; but many, about local structures and hierarchies, can still be answered.

4.2 Local network models

In this part, we define *local network models*, a class of Siena models where it is only the structure of ties for actors that are geodesically ‘close’ to the ego actor that is important when they make network tie changes. We require some definitions:

Definition 4.2.1 (Outties). For a network x and an ego i , let $\tilde{o}(i|x) = \{j \in \{1, \dots, n\} : x_{ij} = 1\}$ be the set of nodes which have an incoming tie from i .

Definition 4.2.2 (Inties). For a network x and an ego i , let $\tilde{i}(i|x) = \{j \in \{1, \dots, n\} : x_{ji} = 1\}$ be the set of nodes which send an outgoing tie to i .

Definition 4.2.3 (Neighbourhood). Given a network x , and an ego i , let the neighbourhood of i , denoted by $\mathcal{N}(i|x)$, be the union of $\{i\}$ and the set of actors that are a(n undirected) geodesic distance of at most one from i :

$$\mathcal{N}(i|x) = \{i\} \cup \tilde{o}(i|x) \cup \tilde{i}(i|x).$$

Hanneman and Riddle (2011) refer to this as a one-step neighbourhood (although they include in their definition not only the set of actors, but the ties between them).

For a matrix A of dimension $n \times n$ and a vector v such that each component of v is in the set $\{1, \dots, n\}$, let $A_{[v]}$ be the matrix of dimension $\dim(v) \times n$ such that, for all $i \in \{1, \dots, \dim(v)\}$, the i th row is given by the v_i th row of A . This means that, for $j = 1, \dots, n$, the (i, j) th component of $A_{[v]}$ is given by

$$A_{[v]i,j} = A_{v_i,j}.$$

Definition 4.2.4 (Neighbourhood statistics). For a network x and an ego i , the neighbourhood statistics are given by the $p \times n$ matrix, where $p = |\mathcal{N}(i|x)| \leq 2(1 + x_{i+} + x_{+i})$, given by

$$N(i|x) = \begin{pmatrix} x_{[\mathcal{N}(i|x)]} \\ x_{[\mathcal{N}(i|x)]}^T \end{pmatrix}$$

The neighbourhood statistics give the incoming and outgoing ties of both i and i 's (incoming and outgoing) ties.

Definition 4.2.5 (Change contribution for an effect). For all $i, j \in \{1, \dots, n\}$, given a network x , the change contribution for an effect is the difference in the value of the effect evaluated at $x(i \rightsquigarrow j)$ and x :

$$\Delta_{ij}(x) = s_i(x(i \rightsquigarrow j)) - s_i(x).$$

This gives the increase in the effect were ego i to change their relationship to alter j .

Definition 4.2.6 (Local network effect). An effect is a local network effect if, for all $i \in \{1, \dots, n\}$, and for any two possible states of the network, x, \tilde{x} , such that $N(i|x) = N(i|\tilde{x})$,

$$\Delta_{ij}(x) = \Delta_{ij}(\tilde{x}),$$

for all $j \in \{1, \dots, n\}$.

Examples of local network effects include all transitive effects and almost all covariate effects. Example of non-local network effects include popularity and assortativity effects.

Definition 4.2.7 (Local network models). A local network model is a Siena network model such that all the effects are local network effects.

Note that we can write the conditional choice probabilities (see (2.5)) in terms of the change contributions; for a network x , and an ego i , for $j \in \{1, \dots, n\}$,

$$p_{ij}(x; \theta) = \frac{\exp(\beta^T \Delta_{ij}(x; \theta))}{\sum_{k=1}^n \exp(\beta^T \Delta_{ik}(x; \theta))}, \quad (4.1)$$

where $\theta = (\rho, \beta)$.

Looking at (4.1), it follows from the definition of a local network model that, given an ego i and a parameter θ , if two networks have the same neighbourhoods statistics, the set of choice probabilities for i will be the

same. Therefore, for a local network model, an actor's decision of which tie to extend or break only depends on the configuration of the ties in and around their neighbourhood; they are unaffected by changes elsewhere in the network.

We now derive an alternative, equivalent, definition of a local effect, which will be useful in determining whether an effect is local or not.

Condition 4.2.1. Given a network x and an ego i , let $(l, m) \in \{1, \dots, n\}^2$ such that $l \neq m$. Then

1. $l \neq i$;
 2. $m \neq i$;
 3. $\max\{x_{il}, x_{im}, x_{li}, x_{mi}\} = 0$.
-

Lemma 4.2. Given a network x and any ego $i \in \{1, \dots, n\}$, let (l, m) be any pair in $\{1, \dots, n\}^2$ such that $l \neq m$. Then

$$N(i|x) = N(i|x(l \rightsquigarrow m))$$

if and only if Condition 4.2.1 holds.

Proof. If we change the network from x to $x(l \rightsquigarrow m)$, then the l th row of x and the m th row of x^T are the only rows which change. Therefore, by definition, the neighbourhood statistics change if and only if at least

one of l or m are in $\mathcal{N}(i|x)$. This occurs if and only if Condition 4.2.1 does not hold. \square

Theorem 4.3. *Given a network x and any ego $i \in \{1, \dots, n\}$, let (l, m) be any pair in $\{1, \dots, n\}^2$ such that $l \neq m$. Suppose that Condition 4.2.1 implies that $\Delta_{ij}(x; \theta) = \Delta_{ij}(x(l \rightsquigarrow m); \theta)$ for all $j = 1 \dots, n$. Then the effect is a local effect.*

Proof. Fix an ego $i \in \{1, \dots, n\}$. Let x, x' be networks such that $N(i|x) = N(i|x')$. Then there exists a sequence of networks $x^{(1)}, \dots, x^{(P)}$, for some P , such that $x^{(1)} = x, x^{(P)} = x'$, and for all $p = 2, \dots, P$,

$$x^{(p)} = x^{(p-1)}(l^{(p)} \rightsquigarrow m^{(p)}),$$

for some $(l^{(p)}, m^{(p)}) \in \{1, \dots, n\}^2$. Furthermore, because none of incoming or outgoing ties for actors in $\mathcal{N}(i|x)$ need to change, this can be chosen so that, for all $p = 2, \dots, P$,

$$N(i|x^{(p)}) = N(i|x^{(p-1)}).$$

Then Lemma 4.2 implies that, for all $p = 2, \dots, P$, Condition 4.2.1 holds for $(l^{(p)}, m^{(p)})$ and so, by the statement of the theorem, for all $j = 1, \dots, n$,

$$\Delta_{ij}(x^{(p)}; \theta) = \Delta_{ij}(x^{(p-1)}; \theta),$$

and so

$$\Delta_{ij}(x'; \theta) = \Delta_{ij}(x; \theta).$$

Therefore the effect is a local effect. □

4.3 Categorising polynomial effects

Most network effects can be expressed as polynomials in the elements of x , where, because the values are all binary, the highest power of any component is one. In this section, we describe an algorithm for determining whether or not a polynomial network effect is a local network effect.

Given an ego i , let $j, l, m \in \{1, \dots, n\}$ such that $i \neq j, l \neq m$ and Condition 4.2.1 applies to (l, m) . Then a polynomial effect can be written in the form

$$s_i(x) = x_{ij}x_{lm}d_{1j}(x) + x_{ij}d_{2j}(x) + x_{lm}d_{3j}(x) + d_{4j}(x), \quad (4.4)$$

where d_{1j}, d_{2j}, d_{3j} and d_{4j} are polynomials that are independent of x_{ij} and x_{lm} (each polynomial has an implicit dependence on i, l and m , but this is not included in the notation for simplicity). Then

$$\Delta_{ij}(x; \theta) - \Delta_{ij}(x(l \rightsquigarrow m); \theta) = d_{1j}(x)(1 - 2x_{ij})(1 - 2x_{lm}).$$

Then, by Theorem 4.3, the effect is a local effect if $\Delta_{ij}(x; \theta) - \Delta_{ij}(x(l \rightsquigarrow m); \theta) = 0$ for all $j \in \{1, \dots, n\}$, and so for a polynomial effect, if $d_{1j}(x) = 0$ for all j .

Example 4.3.1 (Transitive triplets). $s_i(x) = \sum_{k,h} x_{ik}x_{ih}x_{hk}$. Then the only summand involving x_{lm} is when $h = l$ and $k = m$, and so

$$d_{1j}(x) = \begin{cases} x_{il} & j = m, \\ x_{im} & j = l, \\ 0 & \text{otherwise,} \end{cases}$$

and so $d_{1j}(x) = 0$ by Condition 4.2.1. Therefore it is a local effect.

Example 4.3.2 (Indegree popularity). $s_i(x) = \sum_{j,h} x_{ij}x_{hj}$. Then

$$d_{1j}(x) = \begin{cases} 1 & j = m, \\ 0 & \text{otherwise,} \end{cases}$$

which is not zero for all j , and so this is not a local effect.

Chapter 5

Maximum likelihood

5.1 Introduction

We review the algorithm described by Snijders et al. (2010) used for maximum likelihood estimation of a Siena model, and propose changes in order to improve efficiency. Changes will attempt to either decrease computation times, or to improve mixing. In Section 5.2 we review the existing Metropolis-Hastings algorithm that is employed in `RSiena` (Ripley et al., 2011). In Section 5.3, we suggest a method for speeding the calculations required by the algorithm. In Section 5.4, we suggest a method intended to improve the proposal distribution required by the Metropolis-Hastings algorithm.

Most changes in this chapter will apply only to *local models*, as described in Chapter 4; however, in Section 5.6 we will see how the changes, where

applicable, can be used to improve estimation for all models.

In this chapter, so that notation and descriptions can be easier to understand and simpler, we restrict our attention to network models, and not the co-evolution network and behaviour model; however, the methods can all be extended in natural ways for the latter case. We also only consider network models with constant rate functions; the maximum likelihood estimation algorithm for non-constant rates is detailed by Snijders et al. (2010) but is not yet implemented in `RSiena` (Ripley et al., 2011). We consider the case when there are only two observations of the network; again, this is done for simplicity in explanations, but due to the Markov properties of the assumed model, the methods extend easily to cases with more observations. Finally, probabilities and expectations will often have their dependence on θ suppressed, to simplify notation.

5.2 Metropolis-Hasting algorithm

In this section we review the existing algorithm detailed by Snijders et al. (2010) used to perform maximum likelihood estimation for Siena models.

As described earlier, we assume that the network process moves between states via ministeps. We refer to a sequence of consecutive ministeps taking the process from one state to another as a *chain*. Then, given data $(X(t_0), X(t_1)) = (x(t_0), x(t_1))$, to find the maximum likelihood estimate, we want to find θ such that the score function for the observed data is

zero. This is the same (Gu and Kong, 1998) as choosing θ to solve

$$\mathbb{E}_\theta[S_{XV}(\theta; X, V)|(X(t_0), X(t_1)) = (x(t_0), x(t_1))] = 0, \quad (5.1)$$

where S_{XV} is the score function for the chain of ministeps V that takes the process from $X(t_0)$ to $X(t_1)$. We solve this equation using the Robbins-Monro algorithm; to do this, we are required to draw samples of V , given a value for θ . Before explaining how we do this, we require a definition:

Definition 5.2.1 (Toggling). For any $i, j \in \{1, \dots, n\}$, we say that there is a *toggle to the dyad* from one actor i to another actor j to mean that there is a change in the the relationship from i to j : if a tie currently exists, then after toggling it no longer does, and vice versa. Note that we assume the relationship is always non-reflexive, and so toggling a dyad from an actor to themselves leaves the network unchanged.

To draw samples of the chain of ministeps V , we use the Metropolis-Hastings algorithm. We create a chain v of ministeps to use as our initial state, which is constructed so that the first and last states match what is observed, so that $x_0 = x(t_0)$ and $x_R = x(t_1)$, where $x(t_0)$ and $x(t_1)$ are our two observations of the network. Let \mathcal{N}^2 denote the pairs of all actors; then, given our data, we know, for all pairs of actors in \mathcal{N}^2 , the parity of the number of changes to their relationship: for some pair $(i, j) \in \mathcal{N}^2$, if $x_{ij}(t_0) = x_{ij}(t_1)$, then there is an even number of toggles to the dyad, and conversely, if $x_{ij}(t_0) \neq x_{ij}(t_1)$, then the number is odd. We use this

information to choose a suitable v .

We then propose a new chain \tilde{v} using proposal distribution $u(\cdot|v)$; we accept the proposed change with probability

$$\alpha = \min \left\{ 1, \frac{p(\tilde{v})u(v|\tilde{v})}{p(v)u(\tilde{v}|v)} \right\}, \quad (5.2)$$

5.2.1 Chain notation

We often denote a chain of ministeps by v . We denote by m_i the i th minstep in the chain, and by R the number of ministeps in the chain, so that $v = \{m_1, \dots, m_R\}$.

For $r \in \{1, \dots, R\}$, we denote by (i_r, j_r) the tie that is toggled in minstep m_r . For the chain v , there is a corresponding sequence of $R + 1$ networks, which we denote x_0, \dots, x_R . Given x_{r-1} , the network x_r is equal to $x_{r-1}(i_r \rightsquigarrow j_r)$.

5.2.2 Proposal distribution

The proposed chain \tilde{v} is obtained by making a small change to v , which we refer to as an update. There are five different types of updates, and the type is chosen randomly at each proposed update, using the update type proposal distribution.

Definition 5.2.2 (Update type proposal distribution). The update type proposal distribution is the probability distribution for the type of update to be used at each proposed update. The sample space is $\{1, 2, 3, 4, 5\}$ and we denote the distribution function by p_U .

The update type proposal distribution is predetermined before the estimation begins. Every type of update preserves the parity of the number of changes to every component of x . Before describing the updates, we first require a definition.

Definition 5.2.3 (Diagonal ministep). A ministep is *diagonal* if the ego to make no change to their outgoing ties; i.e. when choosing $j \in \{1, \dots, n\}$ according to the conditional choice probabilities, the ego i chooses $j = i$.

The types of updates are:

1. Add a diagonal: insert a diagonal ministep into the chain;
2. Delete a diagonal: delete a diagonal ministep from the chain;
3. Permutation: permute the ordering of some of the ministeps in the chain;
4. Add a cancelling pair: insert two ministeps which toggle the same dyad;
5. Delete a cancelling pair: delete two ministeps which toggle the same dyad.

Changes of types 1, 2, 4 and 5 are sufficient to ensure that the entire support of \mathcal{V} , the space of chains satisfying the observed data, may be reached, given any starting point of the Markov chain (Snijders et al., 2010). Type 3 is included to achieve better mixing properties. In Section 5.4, we suggest an alternative to permuting ministeps.

Since a single update only changes the chain very slightly, given a sampled chain, we obtain another sample only after making many updates: we describe this in more detail in Algorithm 1.

5.3 Improving efficiency for local models

When we propose to make an update to the chain, for a subsection of the chain, we need to recalculate the probabilities of the ministeps; for example, if we insert a pair, then we need to recalculate the probabilities of the ministeps between these two new steps. These calculations involves products of sums of exponentials, and are therefore computationally expensive. In this section, we consider how these calculations can be simplified.

Given a chain v , Table 5.1 shows how the chain, and the probabilities of the ministeps, are changed by the addition of a pair of cancelling ministeps. The ministeps before the first inserted ministeps are unchanged. The ministeps after the second inserted ministep have their probabilities unchanged but their position moved forward by two: the ministep in position r is now in position $r + 2$, due to the earlier inserted ministeps. We need to calculate the probabilities of the inserted ministeps, and also the

Let $\mathcal{R} = \{1, \dots, R\}$ and $\mathcal{R}_+ = \{1, \dots, R + 1\}$;
 Given a chain v ,
for $k = 1, \dots, K$ **do**

- $u \sim p_U$;
- if** $u=1$ **then**
 - Propose adding a diagonal. For a random $r \in \mathcal{R}_+$, some $l \in \{1, \dots, n\}$ is chosen and a ministep toggling the dyad (l, l) , (it is diagonal, so no change actually occurs), is inserted before position r (if $r = R + 1$, then the ministep is inserted at the end).
- else if** $u=2$ **then**
 - Propose deleting a diagonal. For a random $r \in \mathcal{R}$, with $i_r = j_r$, ministep m_r is deleted.
- else if** $u=3$ **then**
 - Propose permuting a section of the chain. For a random pair $(r_1, r_2) \in \mathcal{R}^2$, with $0 < r_2 - r_1 < K$, where K is chosen so to avoid too lengthy calculations, the subsequence of ministeps m_{r_1}, \dots, m_{r_2} is randomly permuted.
- else if** $u=4$ **then**
 - Propose adding a cancelling pair. For a random $r_1 \in \mathcal{R}$, a dyad (l, m) is chosen according to the usual choice probabilities, given the network x_{r_1-1} . A random $r_2 \in \mathcal{R}$ is chosen so that $r_2 \geq r_1$ and $(i_r, j_r) \neq (l, m)$ for all $r \in \{r_1, \dots, r_2\}$, and ministeps toggling (l, m) are inserted before r_1 and r_2 .
- else** $u = 5$
 - Propose deleting a cancelling pair. For a random pair $(r_1, r_2) \in \mathcal{R}^2$, $r_1 < r_2$, with $(i_{r_1}, j_{r_1}) = (i_{r_2}, j_{r_2})$, $i_{r_1} \neq j_{r_1}$, and $(i_r, j_r) \neq (i_{r_1}, j_{r_1})$ for all $r \in \{r_1 + 1, \dots, r_2 - 1\}$, m_{r_1} and m_{r_2} are deleted.
- end**
- if** *proposal is accepted* **then**
 - $v = \tilde{v}$.
- end**

end
return v

Algorithm 1: Metropolis-Hastings algorithm

Section	r	\tilde{m}_r	\tilde{x}_{r-1}	$\mathbb{P}(\tilde{m}_r \tilde{x}_{r-1})$
1	$1, \dots, r_1 - 1$	m_r	x_{r-1}	$\mathbb{P}(m_r x_{r-1})$
2	r_1	(l, m)	x_{r_1-1}	$\mathbb{P}((l, m) x_{r_1-1})$
3	$r_1 + 1, \dots, r_2$	m_{r-1}	$x_{r-2}(l \rightsquigarrow m)$	$\mathbb{P}(m_{r-1} x_{r-2}(l \rightsquigarrow m))$
4	$r_2 + 1$	(l, m)	$x_{r_2-1}(l \rightsquigarrow m)$	$\mathbb{P}((l, m) x_{r_2-1}(l \rightsquigarrow m))$
5	$r_2 + 2, \dots, R + 2$	m_{r-2}	x_{r-3}	$\mathbb{P}(m_{r-2} x_{r-3})$

Table 5.1: Updated chain of ministeps, and their probabilities, after adding a cancelling pair of ministeps toggling (l, m) .

Section	r	\tilde{m}_r	\tilde{x}_{r-1}	$\mathbb{P}(\tilde{m}_r \tilde{x}_{r-1})$
1	$1, \dots, r_1 - 1$	m_r	x_{r-1}	$\mathbb{P}(m_r x_{r-1})$
2	$r_1, \dots, r_2 - 2$	m_{r+1}	$x_r(l \rightsquigarrow m)$	$\mathbb{P}(m_{r+1} x_r(l \rightsquigarrow m))$
3	$r_2 - 1, \dots, R - 2$	m_{r+2}	x_{r+1}	$\mathbb{P}(m_{r+2} x_{r+1})$

Table 5.2: Updated chain of ministeps, and their probabilities, after deleting a cancelling pair of ministeps m_{r_1} and m_{r_2} which both toggle (l, m) .

probabilities of the ministeps $\tilde{m}_{r_1+1}, \dots, \tilde{m}_{r_2}$, that occur between the two inserted ministeps.

Similarly, Table 5.2 shows the ministeps and probabilities once a cancelling pair of ministeps are deleted. The probabilities for the ministeps before the first deleted ministep, and after the second, are unchanged, but the remaining ministeps, now labelled $\tilde{m}_{r_1}, \dots, \tilde{m}_{r_2-2}$, need to have their probabilities recalculated.

In both cases, for some $r_{\min}, r_{\max} \in \{1, \dots, R\}$ with $r_{\min} \leq r_{\max}$, we need to calculate, for some $(l, m) \in \mathcal{N}^2$,

$$\tilde{\mathcal{P}} = (\mathbb{P}(m_{r_{\min}}|x_{r_{\min}-1}(l \rightsquigarrow m)), \dots, \mathbb{P}(m_{r_{\max}}|x_{r_{\max}-1}(l \rightsquigarrow m))).$$

Note that, at this point in the algorithm, we have already calculated

$$\mathcal{P} = (\mathbb{P}(m_{r_{\min}} | x_{r_{\min}-1}), \dots, \mathbb{P}(m_{r_{\max}} | x_{r_{\max}-1})).$$

We consider local models, and so can use the following theorem:

Theorem 5.3. *For a local model, if*

$$N(i_r | x_{r-1}(l \rightsquigarrow m)) = N(i_r | x_{r-1})$$

then

$$\mathbb{P}(m_r | x_{r-1}(l \rightsquigarrow m)) = \mathbb{P}(m_r | x_{r-1}).$$

Proof. By the definition of a local model, if $N(i_r | x_{r-1}(l \rightsquigarrow m)) = N(i_r | x_{r-1})$, then for all $j = 1, \dots, n$,

$$\Delta_{i_r j}(x_{r-1}(l \rightsquigarrow m); \theta) = \Delta_{i_r j}(x_{r-1}; \theta),$$

and so

$$\begin{aligned} \mathbb{P}(m_r | x_{r-1}(l \rightsquigarrow m)) &= \frac{\exp[\Delta_{i_r j_r}(x_{r-1}(l \rightsquigarrow m); \theta)]}{\sum_{j=1}^n \exp[\Delta_{i_r j}(x_{r-1}(l \rightsquigarrow m); \theta)]}, \\ &= \frac{\exp[\Delta_{i_r j_r}(x_{r-1}; \theta)]}{\sum_{j=1}^n \exp[\Delta_{i_r j}(x_{r-1}; \theta)]}, \\ &= \mathbb{P}(m_r | x_{r-1}). \end{aligned} \tag{5.4}$$

□

By Theorem 5.3, we do not need to calculate every component of $\tilde{\mathcal{P}}$: for $r \in r_{\min}, \dots, r_{\max}$, if $N(i_r|x_{r-1}(l \rightsquigarrow m)) = N(i_r|x_{r-1})$, then $\mathbb{P}(m_r|x_{r-1}(l \rightsquigarrow m)) = \mathbb{P}(m_r|x_{r-1})$ and

$$\tilde{\mathcal{P}}_{r+1-r_{\min}} = \mathcal{P}_{r+1-r_{\min}}.$$

This reduces the number of calculations from $1 + r_{\max} - r_{\min}$ to

$$\#\{r \in \{r_{\min}, \dots, r_{\max}\} : N(i_r|x_{r-1}(l \rightsquigarrow m)) \neq N(i_r|x_{r-1})\}.$$

For each $r \in r_{\min}, \dots, r_{\max}$, whether or not $N(i_r|x_{r-1}(l \rightsquigarrow m)) = N(i_r|x_{r-1})$ can be easily found using Lemma 4.2. In virtually all cases, checking whether the neighbourhood statistics are unchanged rather than calculating every probability should speed up calculations.

5.3.1 Example

In this section, we consider three real datasets to assess the effect of changing the calculations in the way described in this section. We consider three real observed networks, each of different sizes, in order to also see how the computational saving varies with network size. The networks we consider are:

1. A network of size 32 of friendship ties between university freshmen

(van de Bunt et al., 1999).

2. A network of size 50 consisting of friendship ties for a set of female Glaswegian school children (West and Sweeting, 1996), which we will refer to as the *s50* dataset.
3. A network of size 129 of friendship ties between Glaswegian school children (West and Sweeting, 1996), which we will refer to as the Glasgow dataset.

For all three datasets we use two observations, so that there is one basic rate parameter to estimate. The model we choose consists of the network effects *outdegree*, *reciprocity*, *transitive triplets*, *3-cycles* and *smoking similarity*. For the first 2 datasets, we also include a *drinking similarity* effect, and for the third, a *gender similarity* effect.

We perform Siena estimation runs using the default specifications, using first the original algorithm, and then using method described in this section. We repeat this 50 times.

The average computation times are shown in Figure 5.1. There is a reduction in time taken of 17%, 43% and 68%, for the first, second and third datasets, respectively; the standard deviation of the fractions of time taken by the new method compared to the original method is very small in all cases (less than 0.01). Therefore the improvement increases with network size, as we would expect (an actor's neighbourhood is a larger proportion of the total network for smaller network, so a higher proportion of minimesteps will need to be recalculated than for a large network, when many

ministeps will feature egos outside the actor's neighbourhood).

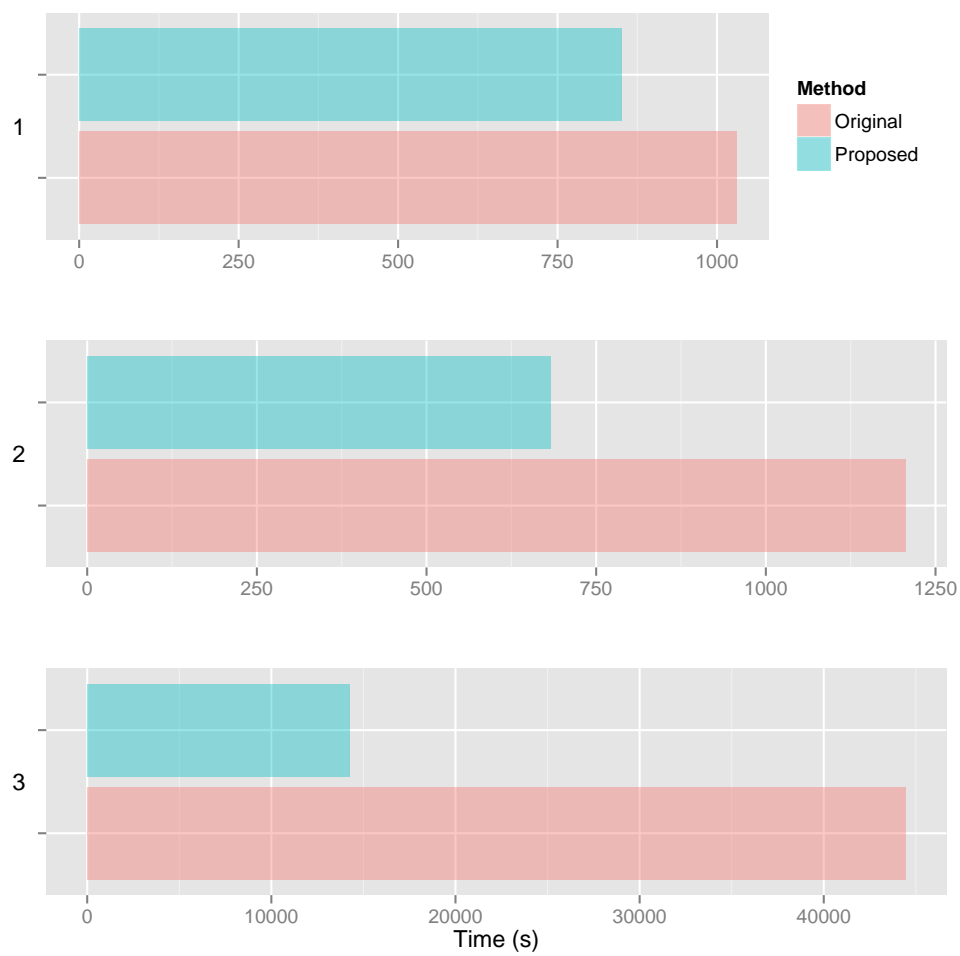
5.4 Improving the proposal distribution

Any proposal distribution (subject to regularity conditions) will eventually deliver samples from the target distribution (Gilks et al., 1996). However, the rate of convergence to the stationary distribution will depend on the proposal distribution. If the proposal distribution gives small acceptance rates, then the Markov chain will remain static for many of the iterations, and so will be slow to converge. If instead the moves proposed are very 'small', and do not make a significant change to the random variable (in our model, the chain of ministeps) then acceptance rates may be high, but the Markov chain will move very slowly around the support of the target distribution.

In the current algorithm by Snijders et al. (2010), a ministep in the chain may be moved only by a permutation of what will usually be a small section of the chain. This typically has high acceptance rates but only achieves a small amount of mixing, since each of the ministeps which moves only does by a small amount (relative to the entire length of the chain). In Section 5.4.1 we propose to move a single ministep to any position in the chain. Only one ministep is moved in this proposal, but by a potentially large distance, and so this move will arguably achieve better mixing.

In this section, we still consider only local models; in Section 5.6, we consider modifying this proposal so it can be used with non-local models.

Figure 5.1: Computation times for a Siena analysis of the three datasets described in Section 5.3.1.



5.4.1 Moving a tie

We suggest making a update by taking a ministep from the existing chain and changing its position in the chain, where the position is chosen depending on the probability of the resulting chain; note the similarity between this type of update and Gibbs sampling (Casella and George, 1992).

We randomly choose $r_1 \in \{1, \dots, R\}$ and remove the tie change $(i, j) := (i_{r_1}, j_{r_1})$ from this position. Then the chain has length $R - 1$, and we denote this new reduced chain v' . We then insert a tie change to (i, j) directly before position $r_2 \in \{1, \dots, R\}$ of v' ($r_2 = R$ means that the tie change is inserted at the end of v'). We want to choose r_2 according the probabilities of the resulting chains.

Let $r_{\min} < r_1$ be the position of the tie change to (i, j) before the one at position r_1 (define $r_{\min} = 0$ if there is no earlier change to (i, j)). Similarly, let $r_{\max} > r_1$ be the position of the next tie change to (i, j) after r_1 (define $r_{\max} = R$ if no such change occurs). Then we choose $r_2 = r$, where $r \in \{r_{\min} + 1, \dots, r_{\max}\}$, with probability proportional to

$$\pi_r = p(v'_r), \quad (5.5)$$

where v'_r denotes the chain of ministeps if we take v' and insert a tie change to (i, j) before position r (e.g. $v'_{r_1} = v$).

Calculating $\pi_{r_{\min}}, \dots, \pi_{r_{\max}}$ can be made more efficient using the following recursive formulae. Considering v' (and the corresponding sequence of

networks x'_0, \dots, x'_{R-1}), we calculate $\{\pi_r : r = r_{\min} + 1, \dots, r_{\max}\}$ using

$$\pi_r = \begin{cases} A_r B_r p_{r-1}, & r < r_1, \\ \frac{B_r}{A_r} p_{r-1}, & r \geq r_1, \end{cases} \quad (5.6)$$

where

$$A_r = \begin{cases} 1 & \text{if } N(i_r | x'_{r-1}) = N(i_r | x'_r(i \rightsquigarrow j)), \\ \frac{p((i_r, j_r) | x'_{r-1})}{p((i_r, j_r) | x'_{r-1}(i \rightsquigarrow j))}, & \text{otherwise,} \end{cases}$$

and

$$B_r = \begin{cases} 1 & \text{if } N(i | x'_{r-1}) = N(i | x'_r), \\ \frac{p((i, j) | x'_r)}{p((i, j) | x'_{r-1})}, & \text{otherwise.} \end{cases}$$

The conditions are easy to check, using Lemma 4.2. We choose r_2 according to the probabilities $(\pi_{r_{\min}+1}, \dots, \pi_{r_{\max}})/\pi_+$; the acceptance probability is 1.

5.5 Optimising the proposal distribution

We now have six types of Metropolis-Hastings steps and must update the update type proposal distribution accordingly, so that it now has sample space $\{1, 2, 3, 4, 5, 6\}$. In this section we will use simulation studies to try to make a good choice for the update type proposal distribution.

Before estimation, we must choose the total number of steps K , and $p_U(1), \dots, p_U(6)$, or equivalently, choose K_1, \dots, K_6 , where for $i = 1, \dots, 6$,

$$K_i := \mathbb{E}(\text{number of steps of type } i) = K p_U(i).$$

The Siena default chooses K based on the observed data, and then uses $K_1 = 0.05K, K_2 = 0.05K, K_3 = 0.2K, K_4 = 0.3K, K_5 = 0.3K$ (0.1 K steps are reserved for those dealing with missing data, which we do not consider in this chapter). In this section, we aim to explore what might be a good choice once we incorporate the new 6th type of step. To simplify the problem, we keep the following default proportions:

$$K_1 = K_2; K_4 = K_5; K_4 = 6K_1.$$

This reduces the problem to three dimensions: we must choose K_3, K_6 , and $L = K_1 + K_2 + K_4 + K_5$.

Ripley et al. (2011) recommends that autocorrelations at lag 1 of the scores of the chains in Phase 3 be kept below 0.4 in order to give sample paths that are not too highly correlated; we will follow this advice, and look for values of (L, K_3, K_6) which, on average, achieve this.

We propose replacing permutations with moves, and so, to simplify the problem further we compare two separate cases:

1. Use permutations and no moves: $K_3 > 0; K_6 = 0$;
2. Use moves and no permutations: $K_3 = 0; K_6 > 0$.

We now have two two-dimensional problems. For case 1, we try to find pairs (L, K_3) which give good autocorrelations, and then for case 2, pairs for (L, K_6) . We will then compare results for the two cases, using optimal pairs. In Section 5.5.1 we consider the *s50* dataset, and then in Section 5.5.2, the Glasgow dataset.

5.5.1 Example 1: *s50* data

Using the first two observations of the *s50* dataset, the current default choice in Siena gives $L = 420$ and $K_3 = 120$. We consider a model with network effects *outdegree*, *reciprocity*, *transitive triplets*, *smoking similarity* and *drinking similarity* and find that these defaults do not give sufficiently small autocorrelations for this choice of model, and so we need to explore how to increase them.

Choosing pairs

We explore pairs for case 1 by considering increments of 420 for L and 120 for K_3 : see Algorithm 2. The algorithm begins by following advice given by Ripley et al. (2011): we begin our search for an optimal pair for (L, K_3) by beginning with the default values, and then increasing the total number of steps K while keeping K_3/L constant, until we achieve small autocorrelations. The results from this are given in the diagonal of Table 5.3: beginning with $(L, K_3) = (420, 120)$, we found that we need to increase to $(L, K_3) = (1680, 480)$ to achieve small autocorrelations. We then

successively increase and decrease values for K_3 and L , filling in the off-diagonals of the table.

We may also finish the process (or proceed to the next step of the algorithm) if computation times become excessively long, or if it seems unlikely that small autocorrelations will be found in the current direction.

We then repeat the algorithm for case 2, varying K_6 instead of K_3 ; in this case, we started with $(L, K_6) = (420, 60)$, using a smaller number of moves than the default number of permutations because moves are more computationally demanding. We also use increments of 60 when increasing K_6 , for the same reason.

Evaluating pairs

The results showing autocorrelations and computation times for various pairs are shown in Tables 5.3 and 5.4; each entry is based on averages obtained from five analyses using the specified pairs for (L, K_3) and (L, K_6) , respectively.

Now that we have found some pairs which achieve sufficiently small autocorrelations (which we have labelled (A), ..., (E) in Tables 5.3 and 5.4), we increase the number of repetitions from 5 to 50 and consider the t convergence statistics; see Figure 5.2. Ripley et al. (2011) says that we can interpret that the algorithm is 'converged' when $t < 0.1$ and 'nearly converged' when $t < 0.2$. Looking at the plot, there is not a large difference between any of these five pairs. We can therefore eliminate pair (B) due to


```

1.
i = 1; (L, K3) = (420, 120);
while AC(L, K3) > 0.4 do
  | i = i + 1;
  | (L, K3) = (420i, 120i);
end
(LC, K3,C) = (L, K3);
i = 1;
while L > 420 do
  | while AC(L, K3) < 0.4 do
  | | L = LC - 420i;
  | | K3 = K3,C;
  | | i = i + 1;
  | end
  | (LC, K3,C) = (L, K3);
  | while AC(L, K3) > 0.4 do
  | | L = LC;
  | | K3 = K3,C + 120i;
  | | i = i + 1;
  | end
  | (LC, K3,C) = (L, K3);
end

2.
i = 1;
while K3 > 120 do
  | while AC(L, K3) < 0.4 do
  | | L = LC;
  | | K3 = K3,C - 120i;
  | | i = i + 1;
  | end
  | (LC, K3,C) = (L, K3);
  | while AC(L, K3) > 0.4 do
  | | L = LC + 420i;
  | | K3 = K3,C;
  | | i = i + 1;
  | end
  | (LC, K3,C) = (L, K3);
end

```

Algorithm 2: Finding pairs with small AC

Table 5.3: Using permutations with s50 data: autocorrelations and computation times for different pairs of (L, K_3) , where $K_6 = 0$.

K_3	L							
	420		840		1260		1680	
	AC	Time	AC	Time	AC	Time	AC	Time
120	0.60	340s						
240			0.49	646s				
360					0.43	999s	0.41	1163s
480			0.44	842s	0.40(A)	1055s	0.35(B)	1386s
600			0.45	921s				
720			0.41	1003s				
840	0.49	866s	0.38(C)	1114s				

Table 5.4: Using moves with s50 data: autocorrelations and computation times for different pairs of (L, K_6) , where $K_3 = 0$.

K_6	L					
	420		840		1260	
	AC	Time	AC	Time	AC	Time
60	0.53	387s	0.44	541s	0.41	740s
120	0.45	452s	0.37(D)	678s		
180	0.41	565s				
240	0.40(E)	685s				

the large computation time (shown in Table 5.3). There is little to choose between (A) and (C), and between (D) and (E), but we can see that the computation time is reduced using moves rather than permutations (by more than a third).

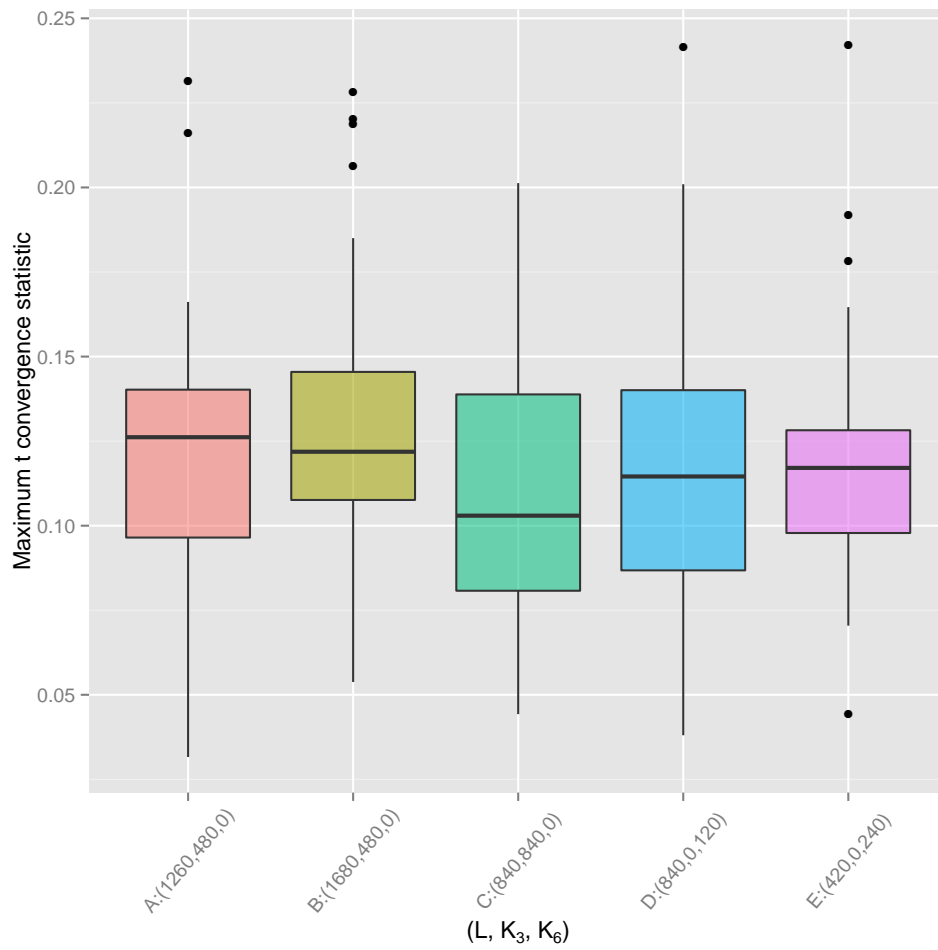


Figure 5.2: Maximum t convergence statistics using different values of (L, K_3, K_6) with autocorrelations at most 0.4.

5.5.2 Example 2: Glasgow data

We now consider the first two observations of the Glasgow dataset, which has 129 actors. We consider a model with network effects *outdegree*, *reciprocity*, *transitive triplets*, *smoking similarity* and *gender similarity*. The current default choice in Siena gives $L = 1680$ and $K_3 = 480$, and, as with the *s50* data, we find that in this case, these values do not give sufficiently small autocorrelations for this choice of model.

We follow the same method as in Section 5.5.1 to find pairs of (L, K_3) and (L, K_6) that give sufficiently small autocorrelations, but using increments of $(1680, 480)$ for case 1, and $(1680, 240)$ for case 2. The results are shown in Tables 5.5 and 5.6. The results give a number of pairs of both (L, K_3) and (L, K_6) that provide sufficiently small autocorrelations, so we now need to compare the times and convergence statistics to find the optimal pairs.

Since in the previous section, looking at 50 repetitions to look for differences in t convergences statistics did not provide much further information, and because in this section, computation times in case 1 are very long, we consider the t convergence statistics based on just 5 repetitions. These are shown in Figure 5.3, and we see that the maximum convergence statistics are generally quite low, and in almost all cases are less than 0.2. The variance of the statistics is fairly high in some cases (which could just be because we use only 5 repetitions in each case). If we say that a pair is optimal if the mean and variance of the maximum convergence statistic is low, then we would probably conclude that pair (E) is the optimal choice

Table 5.5: Using permutations with Glasgow data: autocorrelations and computation times for different pairs of (L, K_3) , where $K_6 = 0$

K_3	L							
	1680	5040	8400	10080	11760	13440	15120	16800
	AC (Time)	AC (Time)	AC (Time)	AC (Time)	AC (Time)	AC (Time)	AC (Time)	AC (Time)
0								0.39(A) (59551s)
480	0.66 (9059s)							0.38(B) (60512s)
960								0.40(C) (61573s)
1440		0.56 (26593s)						0.38(D) (62892s)
1920								0.38(E) (65298s)
2400			0.45 (44456s)					0.37(F) (68832s)
2880				0.45 (53358s)		0.42 (64173s)	0.41 (68512s)	0.36(G) (73082s)
3360					0.42 (62113s)	0.38(H) (68417s)		
3840				0.44 (61080s)	0.39(I) (67026s)	0.40(I) (71319s)		
4320			0.43 (57535s)	0.40(K) (64030s)				
4800			0.44 (60063s)					
5760			0.43 (64823s)					

Table 5.6: Using moves with Glasgow data: autocorrelations and computation times for different pairs of (L, K_6) , where $K_3 = 0$

K_6	L					
	1680		3360		5040	
	AC	Time	AC	Time	AC	Time
240	0.58	10139s				
480			0.45	20428s	0.41	25866s
720	0.46	18338s	0.38(L)	24166s	0.35(M)	30804s
960	0.44	22519s				

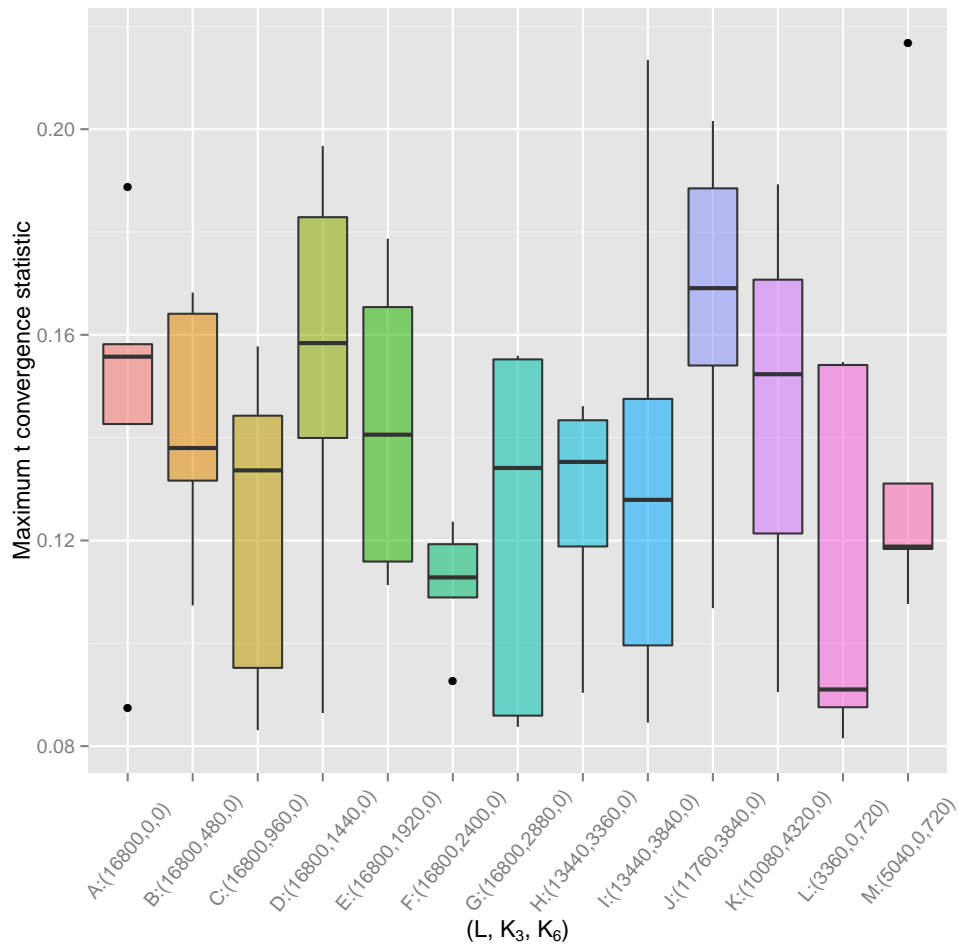


Figure 5.3: Maximum t convergence statistics using different values of (L, K_3, K_6) with autocorrelations at most 0.4.

of those using permutations and pair (M) is the optimal choice of those using moves. The distribution of the convergence statistics results for these two pairs are similar, and the average autocorrelations are slightly lower for pair (M). Looking at the average computation times, which are 65298 seconds for pair (E) and 30804 seconds for pair (M), we can conclude that the optimal pair using moves is more than twice as fast as the optimal pair using permutations.

5.6 Non-local models

We now consider a modification of the new update type which can be used with non-local models.

The recursive formula that was used to calculate $\{\pi_r = p(v'_r) : r = r_{\min} + 1, \dots, r_{\max}\}$ (see equations (5.5) and (5.6)) is no longer applicable for a non-local model, and so calculation of this will be very time consuming. Instead we propose to choose r_2 according to the probabilities

$$(\tilde{\pi}_{r_{\min}+1}, \dots, \tilde{\pi}_{r_{\max}}) / \tilde{\pi}_+,$$

where

$$\tilde{\pi}_r = \begin{cases} A_r B_r p_{r-1}, & r < r_1, \\ \frac{B_r}{A_r} p_{r-1}, & r \geq r_1, \end{cases} \quad (5.7)$$

and where

$$A_r = \begin{cases} \frac{p((i_r, j_r) | x'_{r-1})}{p((i_r, j_r) | x'_{r-1}(i \rightsquigarrow j))}, & \text{if } i_r \in \{i, j\} \text{ or } j_r \in \{i, j\}, \\ 1 & \text{otherwise,} \end{cases}$$

and

$$B_r = \begin{cases} \frac{p((i, j) | x'_r)}{p((i, j) | x'_{r-1})}, & \text{if } i_r \in \{i, j\} \text{ or } j_r \in \{i, j\}, \\ 1 & \text{otherwise.} \end{cases}$$

We then accept the update with probability

$$\min \left\{ 1, \frac{p(\tilde{v})\tilde{\pi}_{r_1}}{p(v)\tilde{\pi}_{r_2}} \right\}.$$

This should hopefully give a high proportion of accepted updates, because in many cases

$$(\tilde{\pi}_{r_{\min}+1}, \dots, \tilde{\pi}_{r_{\max}}) / \tilde{\pi}_+$$

will be fairly close to

$$(\pi_{r_{\min}+1}, \dots, \pi_{r_{\max}}) / \pi_+.$$

Example

We consider the `s50` data, and modify the model considered earlier in this section, and include a distance-2 effect instead of the drinking similarity effect, so that we have a non-local model.

We use the results from Section 5.5.1, and considering pairs (A), (C), (D) and (E) from Tables 5.3 and 5.4 as our optimal pairs (we exclude (B) due to a high computation time), we compare

(A) using permutations, with $(L, K_3, K_6) = (1260, 480, 0)$;

(C) using permutations, with $(L, K_3, K_6) = (840, 840, 0)$;

(D) using moves, with $(L, K_3, K_6) = (840, 0, 120)$;

(E) using moves, with $(L, K_3, K_6) = (420, 0, 240)$.

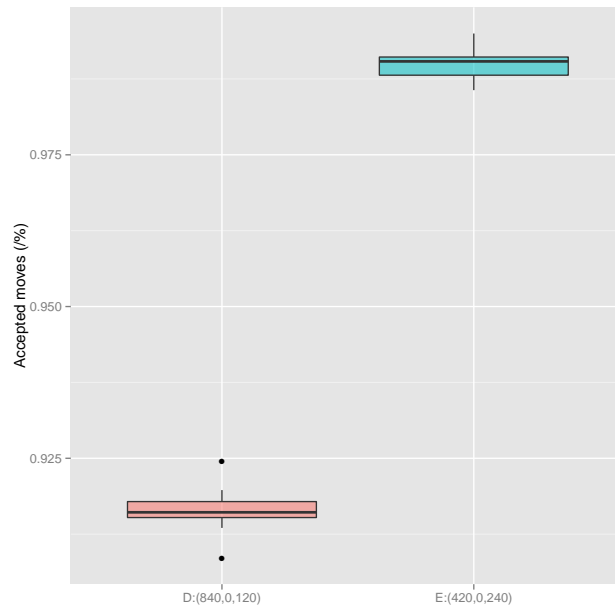


Figure 5.4: Proportion of accepted 'move' updates for a non local model using different values of (L, K_3, K_6)

We consider the results based on 10 repetitions of the analyses.

Firstly, in Figure 5.4 shows the proportion of 'move' update types using pairs (D) and (E); in both cases, the proportion is very high, on average 0.92 and 0.99, respectively. This confirms our hope that there would be a high proportion of accepted 'moves'.

Figure 5.5 shows the maximum autocorrelations for pairs (A), (C), (D) and (E). Looking at this plot, we conclude that (C) does not give sufficiently small autocorrelations, but that the others do.

Figures 5.6 and 5.7 show the computation times and the maximum autocorrelations and maximum convergence t statistics, respectively. The t statistics are better for (E) than for (D), and this also has a lower compu-

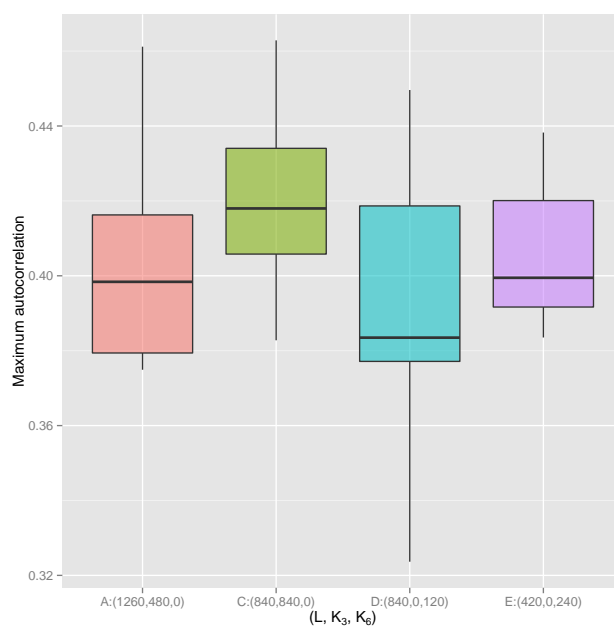


Figure 5.5: Maximum autocorrelations for a non local model using different values of (L, K_3, K_6) .

tation time; we conclude that our optimal pairs using permutations and moves are (A) and (E) respectively. (E) has improve t statistics compare to (A), and computation times reduced by around 37%; we can conclude that the ‘move’ update type is an improvement over the permutation, even for non-local models.

5.7 Effective sample size

In this chapter, we used maximum autocorrelations at lag 1 as a metric to avoid too highly correlated samples; an alternative method, commonly used in MCMC diagnostics, is to consider *effective sample size* (Hoff, 2009). There are various formulae for calculating effective sample size (Thiébaux

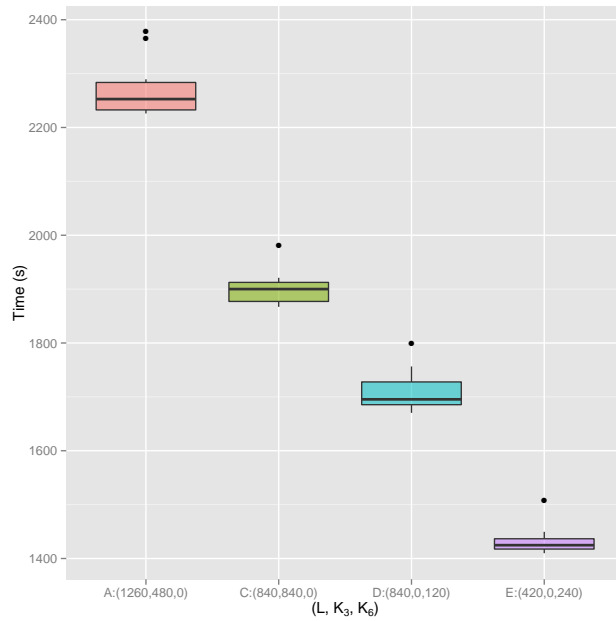


Figure 5.6: Computation times for a non local model using different values of (L, K_3, K_6) .

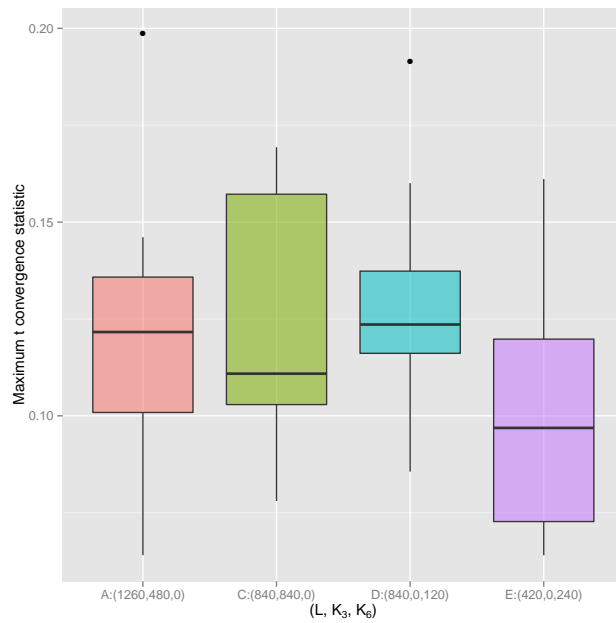


Figure 5.7: Maximum t convergence statistics for a non local model using different values of (L, K_3, K_6) .

and Zwiers, 1984); we consider

$$\frac{n}{1 + 2 \sum_{k:\rho_k \geq 0.05}^{\infty} \rho_k},$$

where ρ_k is the autocorrelation at lag k (Kass et al., 1998). This gives us an effective sample size for each component of our vector-valued score. Figure 5.8 shows the effective sample size for each of these components for a single analysis of the *s50* dataset (using the effects described in Section 5.5.1), using different values of (L, K_3, K_6) . We saw earlier that there seemed to be little difference in autocorrelations between (A), (C), (D) and (E), but that the latter two had lower computation times. We now see that (D) has a higher effective sample size, and (E) has the lowest, and so we may conclude that (D) is the optimal choice.

5.8 Summary of results

Combining the results of Sections 5.3 and 5.4, the methods described in this chapter give the following improvements:

- For the *s50* dataset and a local model, computation times are reduced by 63%, so that the algorithm is nearly three times as fast;
- For the Glasgow dataset, and a local model, computation times are reduced by 85%, so that the algorithm is nearly seven times as fast.
- For a *s50* data and a non-local model, computation times are re-

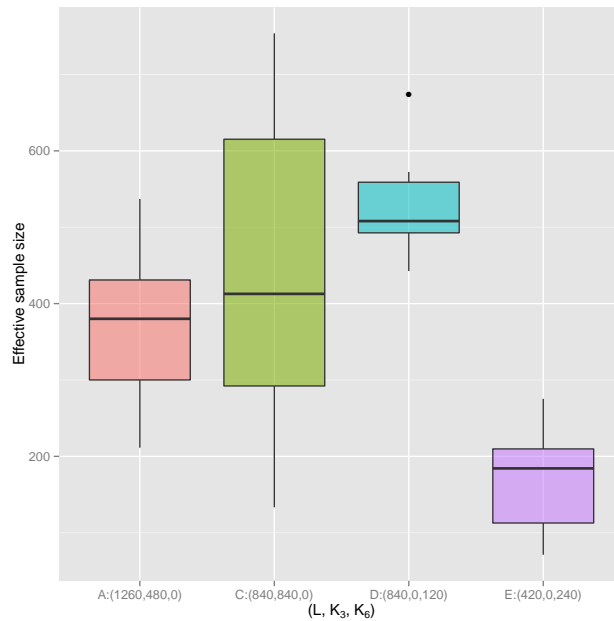


Figure 5.8: Effective sample sizes by parameter for a local model using different values of (L, K_3, K_6) .

duced by 37%.

Evaluating the methods on a non-local model for the Glasgow dataset was computationally unfeasible, mainly because we would need to compare it to the original method, which would be prohibitively slow; however, looking at the results above, we can assume that they would be a reduction of more than that achieved for the s_{50} data (i.e. more than 37%).

In Section 5.7, we briefly considered using effective sample size as a diagnostic for the MCMC simulations, as an alternative to autocorrelations at lag 1. One possible way of using effective sample size could be that, rather than specifying that the autocorrelations must be kept below 0.4, and then rejecting an analysis afterwards if it did not achieve this, when running the

estimation procedure, keep generating samples until an effective sample size of some prespecified value is achieved. It would be interesting to see how using effective sample size as a metric would affect the results seen in this chapter.

Part IV

Importance Sampling in Maximum Likelihood Estimation

Chapter 6

Introduction

As we have seen in the previous part, simulation with maximum likelihood estimation is computationally expensive. It is therefore very important that we put the simulations to best use. In this part, we apply importance sampling (Cappé et al., 2005), bridge sampling (Gelman and Meng, 1998), and thermodynamic integration (Neal, 1993) to improve the use of the simulated data for estimating standard errors and performing likelihood ratio tests. In this chapter, we will describe the assumptions needed to construct strongly consistent estimators of ratios of likelihoods and ratios of expectations.

As in the previous chapter, we restrict our attention to network models, only consider network models with constant rate functions and assume that there are only two observations of the network; again, the final assumption is made for simplicity in explanations, and the methods extend easily to cases with more observations.

6.1 Simulations in *siena07*

Denote $\mathcal{D} = \{(X(t_0) = x(t_0), X(t_1) = x(t_1))\}$. Throughout the Siena maximum likelihood estimation algorithm, we simulate chains of ministeps which take us from one observation to the next, hence conditioning on \mathcal{D} : for some $\theta \in \Theta$, we simulate

$$v \sim p_{\theta}(\cdot|\mathcal{D}).$$

Recall from Section 2.3.1 that the Robbins-Monro stochastic approximation algorithm uses three phases; in this part, we are interested in Phases 2 and 3. In Phase 2, there are n_{sub} subphases (the default value is $n_{\text{sub}} = 4$). For Subphase $k \in \{1, \dots, n_{\text{sub}}\}$, we have a sequence of parameter values $\theta_{k1}, \dots, \theta_{kn_2}$, for some n_{2k} , and for each one, we simulate a single chain: for $i \in \{1, \dots, n_{2k}\}$,

$$v_{2ki} \sim p_{\theta_{ki}}(\cdot|\mathcal{D}).$$

In Phase 3, we simulate n_3 chains using our final parameter estimate $\hat{\theta}$: for $i \in \{1, \dots, n_3\}$,

$$v_{3i} \sim p_{\hat{\theta}}(\cdot|\mathcal{D}).$$

So, from Phases 2 and 3, we have a set of $n_{21} + \dots + n_{2n_{\text{sub}}} + n_3$ simulated chains.

In this chapter, we will use the simulations from Phase 3; later, in Chapter 8, we will also use those from Phase 2.

6.2 Importance sampling

Importance sampling can be used to construct an estimator of the expectation of a function of a random variable with distribution g using simulations from a different distribution f (Cappé et al., 2005).

The basic idea is that if we draw X from f then

$$a(X) = \frac{g(X)}{f(X)} h(X)$$

is an estimator of $\mathbb{E}_g(h(Y))$, where Y is a random variable with distribution g . If the support of g is contained within the support of f then this estimator is unbiased:

$$\mathbb{E}_f \left(\frac{g(X)}{f(X)} h(X) \right) = \int_x \left(\frac{g(x)}{f(x)} h(x) \right) f(x) dx = \int_x g(x) h(x) dx = \mathbb{E}_g(h(Y)).$$

There are a number of reasons why we may prefer to estimate $\mathbb{E}_g(h(Y))$ using $a(x)$, where $x \sim f$ rather than $h(y)$, where $y \sim g$. One reason is for variance reduction: comparing the variances of the two estimators, if we assume that the support of f and g are the same, then

$$\text{Var}_g(h(Y)) - \text{Var}_f(a(X)) = \int_x \left(1 - \frac{g(x)}{f(x)} \right) h^2(x) g(x) dx,$$

which could be negative (or positive), depending on the choice of f . Another reason is that it may be easier to sample from f than from g ; or, as will be the case in this chapter, we may find that we already have simu-

lated samples using f , and can effectively re-use them, using importance sampling, to estimate quantities of the form $\mathbb{E}_g(h(Y))$, for some $g \neq f$. This will be useful for reducing computation times.

In this chapter, we describe how to reuse the chains simulated in Phase 3 to estimate ratios of likelihoods and ratios of expectations.

Definition 6.2.1 (Importance weight). Given a chain v^{sim} simulated according to $p_{\theta^{\text{sim}}}(\cdot|\mathcal{D})$, the importance weight is a function of $\theta \in \Theta$, given by

$$w(\theta; v^{\text{sim}}, \theta^{\text{sim}}) = \frac{p_{\theta}(v^{\text{sim}})}{p_{\theta^{\text{sim}}}(v^{\text{sim}})}.$$

Note that the probabilities in the ratio are not conditional on \mathcal{D} .

For simplicity in the notation, we define importance weights with a second notation, which will be used when it is more convenient.

Definition 6.2.2 (Importance weights). Given a set of data $v_3 = (v_{31}, \dots, v_{3n_3})$ of n_3 chains, simulated according to $p_{\hat{\theta}}(\cdot|\mathcal{D})$, the importance weights are n_3 functions of $\theta \in \Theta$, denoted by $w(\theta; v_3) = (w_1(\theta; v_3), \dots, w_{n_3}(\theta; v_3))$, where, for $k = 1, \dots, n_3$,

$$w_k(\theta; v_3) = w(\theta; v_{3k}, \hat{\theta}).$$

Assumption 1. *The total rate parameter λ (the expected number of ministeps in the chain) lies in a finite interval: there exists $L < \infty$ such that $\lambda \in [0, L]$.*

For the remainder of this part we assume that Assumption 1 holds.

As mentioned in the previous chapter, the proposal distribution allows us to reach any chain that satisfies the observed data, given any starting point, and so the Markov chain is irreducible.

Assumption 2. *The Markov chain of chains generated by the Metropolis Hastings algorithm is positive recurrent.*

This assumption is unproven for our Markov chain, which has a countable, but not finite, state space; however, we could slightly modify our model, and instead of assuming that the number of ministeps in the chain is distributed as a Poisson random variable, we could truncate the space, making the restriction that there is some finite maximum number of ministeps. We would then have a finite state space, and positive recurrence would be guaranteed (Norris, 1997).

Theorem 6.1. *Let F be a function of a chain such that there exists a constant c_F such that $|F(V)| < c_F M(V)$, where $M(V)$ is the number of ministeps in the chain. Then, for every $\theta \in \Theta$,*

$$IS(\theta) = \frac{1}{n_3} \sum_{k=1}^{n_3} w_k(\theta; v_3) F(v_{3k}),$$

is an unbiased estimator of

$$\mu(\theta) = \frac{p_\theta(\mathcal{D})}{p_{\hat{\theta}}(\mathcal{D})} \mathbb{E}_\theta(F(V)|\mathcal{D}).$$

Moreover, given Assumption 2, for every $\theta \in \Theta$, it is strongly consistent.

Proof. For $k = 1, \dots, n_3$,

$$\mathbb{E}_{\hat{\theta}}(w_k(\theta; v_3) F(V_k)|\mathcal{D}) = \int_{v|\mathcal{D}} \frac{p_\theta(v)}{p_{\hat{\theta}}(v)} F(v) p_{\hat{\theta}}(v|\mathcal{D}) dv, \quad (6.2)$$

$$= \frac{p_\theta(\mathcal{D})}{p_{\hat{\theta}}(\mathcal{D})} \int_{v|\mathcal{D}} F(v) p_\theta(v|\mathcal{D}) dv, \quad (6.3)$$

$$= \frac{p_\theta(\mathcal{D})}{p_{\hat{\theta}}(\mathcal{D})} \mathbb{E}_\theta(F(V)|\mathcal{D}), \quad (6.4)$$

and so the estimator is unbiased.

Given Assumption 2, by the Birkhoff ergodic theorem (see, e.g. Norris (1997)), to show that the estimator is strongly consistent, it is sufficient to show that $\mu(\theta)$ is bounded. By Assumption 1,

$$\left| \frac{p_\theta(\mathcal{D})}{p_{\hat{\theta}}(\mathcal{D})} \mathbb{E}_\theta(F(V)|\mathcal{D}) \right| \leq \frac{c_F L}{p_{\hat{\theta}}(\mathcal{D})} < \infty.$$



□

Note that, in Theorem 6.1, and elsewhere in this part, we use the term *strongly consistent* to mean that, conditional on the observed data \mathcal{D} , as we increase the number of simulated chains, the estimator converges almost surely to the quantity of interest.

Note further that the score of a chain satisfies the condition required by the function F .

Corollary 6.5.

$$\frac{1}{n_3} \sum_{k=1}^{n_3} w_k(\theta; v_3),$$

is an unbiased estimator of

$$\frac{p_\theta(\mathcal{D})}{p_{\hat{\theta}}(\mathcal{D})},$$

and, given Assumption 2, it is strongly consistent.

Proof. This follows from Theorem 6.1, substituting $F = 1$.

□

Corollary 6.6. Let F_A and F_B be functions of a chain such that there exists a constant c such that $|F_A(V)| < cM(V)$ and $|F_B(V)| < cM(V)$, where $M(V)$ is the number of ministeps in the chain. Then, given Assumption 2, for every $\theta \in \Theta$,

$$IS(\theta) = \frac{A}{B},$$

where

$$A = \frac{1}{n_3} \sum_{k=1}^{n_3} w_k(\theta; v_3) F_A(v_{3k}),$$

and

$$B = \frac{1}{n_3} \sum_{k=1}^{n_3} w_k(\theta; v_3) F_B(v_{3k}),$$

is a strongly consistent estimator of

$$\frac{\mathbb{E}_\theta(F_A(V)|\mathcal{D})}{\mathbb{E}_\theta(F_B(V)|\mathcal{D})}.$$

Proof. By Theorem 6.1,

$$A \xrightarrow{a.s.} \mathbb{E}_{\hat{\theta}}(A|\mathcal{D}) = \frac{p_\theta(\mathcal{D})}{p_{\hat{\theta}}(\mathcal{D})} \mathbb{E}_\theta(F_A(V)|\mathcal{D})$$

and

$$B \xrightarrow{a.s.} \mathbb{E}_{\hat{\theta}}(B|\mathcal{D}) = \frac{p_\theta(\mathcal{D})}{p_{\hat{\theta}}(\mathcal{D})} \mathbb{E}_\theta(F_B(V)|\mathcal{D}).$$

Therefore, by the continuous mapping theorem,

$$IS(\theta) \xrightarrow{a.s.} \frac{\mathbb{E}_\theta(F_A(V)|\mathcal{D})}{\mathbb{E}_\theta(F_B(V)|\mathcal{D})}.$$

□

6.2.1 Entropy

Entropy is a measure of disorder: in the form we will consider here it measures the disorder of a finite length vector of real numbers.

Definition 6.2.3 (Entropy).

$$Entropy(x_1, \dots, x_n) = -\frac{1}{\log n} \sum_{i=1}^n \left(\frac{x_i}{x_+} \right) \log \left(\frac{x_i}{x_+} \right).$$

In our definition we normalise by dividing by the logarithm of the length of the vector; we do this so that the entropy always lies in the interval $[0, 1]$.

We will use entropy to diagnose poor importance weights, as described by Kong (1992). For a set of importance weights $w(\theta; v_3)$, where $v_3 \sim \hat{\theta}$, we can calculate

$$e(\theta; v_3) = Entropy(w_1(\theta; v_3), \dots, w_{n_3}(\theta; v_3)),$$

for every $\theta \in \Theta$. If the entropy is low, this indicates that variance of the importance weight is high; there will be a small fraction of samples with large weights, and a large number with small weights. This implies a small effective sample size, where the estimate is mostly based on only a few samples.

Throughout this part, we will use entropy as a rejection criteria for sets of importance weights: we will choose a minimum allowed entropy, and then reject weights if and only if their entropy is too small. Low entropy

could happen because the parameter value θ may be too far from $\hat{\theta}$, and so a possible remedy if we need to reject a set of importance weights is to simulate data again using a parameter value that is closer to θ .

Chapter 7

Likelihood ratio test

7.1 Introduction

A likelihood ratio test can be performed to compare the fit of nested models (Neyman and Pearson, 1928; Wilks, 1938), which can then be used for model selection. There are a number of reasons why a likelihood ratio test is appealing. The test statistic is easy to interpret: the likelihood ratio tells us how much more likely one model is compared to the other. It is also transformation invariant, unlike the Wald test (Sorensen and Gianola, 2002). If we are considering simple hypotheses, the likelihood ratio test is the uniformly most powerful test (Neyman and Pearson, 1933).

In practice, however, given the intractability of the likelihood of the observed data for a Siena model, calculating ratios of likelihoods is difficult. In this chapter we will consider methods of using Monte Carlo simula-

tions to do this. The methods are inspired by those described by Gelman and Meng (1998).

7.2 Likelihood ratio test

Recall that we denote the observation of the data by $\mathcal{D} = \{(X(t_0) = x(t_0), X(t_1) = x(t_1))\}$. Then, to perform a likelihood ratio test, given a null hypothesis

$$H_0 : \theta \in \Theta_0,$$

and an alternative hypothesis

$$H_1 : \theta \in \Theta \setminus \Theta_0,$$

we are required to estimate the test statistic

$$T = 2 \log \left(\frac{\sup_{\theta \in \Theta} \{L(\theta|\mathcal{D})\}}{\sup_{\theta \in \Theta_0} \{L(\theta|\mathcal{D})\}} \right) = 2 \log \left(\frac{p_{\hat{\theta}}(\mathcal{D})}{p_{\hat{\theta}_0}(\mathcal{D})} \right), \quad (7.1)$$

which we then compare to a χ_p^2 distribution, where $p = |\Theta| - |\Theta_0|$ (Wilks, 1938).

In this chapter, we will suggest and compare three methods for estimating T : fast forward selection, bridge sampling, and thermodynamic integration. In Section 7.3 we will describe these three methods. Then, in Section 7.4, we will describe a method for evaluating the difficulty of various hypothesis tests, for use when we perform simulation studies. In Section 7.5,

we will perform simulation studies to compare the three methods.

7.3 Approximating the test statistic

In this section, we describe three methods for approximating the test statistic. Firstly, we require an informal definition:

Definition 7.3.1 (Model [*informal*]). In this chapter, we say *model* to mean a set of Siena model effects; for example, if we are modelling a dynamic network, a set of network effects. When we talk about choosing a model, we mean choosing a set of effects.

7.3.1 Fast forward selection

In this section, we will describe ‘fast forward selection’, a quick way of performing multiple likelihood ratio tests using one set of simulated data. This may be used for forward model selection.

For some model \mathcal{M}_0 , we can use the Siena algorithm to estimate the maximum likelihood estimate $\hat{\theta}_0$. Phase 3 will give us simulated chains v_3 .

For any model \mathcal{M} , let the corresponding parameter space be denoted by $\Theta_{\mathcal{M}}$ and the score function for a chain v by $S_{\mathcal{M}}(\theta|v)$. Then there exists a constant $c < \infty$ such that $|S_{\mathcal{M}}(\theta|v)| < cM$, where M is the number of ministeps in v . Then, given Assumption 2, so that there is sufficient mixing

in the chains simulated in Phase 3, we know from Corollary 6.6 that

$$IS_{\mathcal{M}}(\theta; v_3) = \frac{\frac{1}{n_3} \sum_{k=1}^{n_3} w_k(\theta; v_3) S_{\mathcal{M}}(\theta|v_{3k})}{\frac{1}{n_3} \sum_{k=1}^{n_3} w_k(\theta; v_3)}$$

is a strongly consistent estimator of

$$\mathbb{E}_{\theta}(S_{\mathcal{M}}(\theta|V)|\mathcal{D}) = \mathcal{S}_{\mathcal{M}}(\theta|\mathcal{D}).$$

The solution to $\mathcal{S}_{\mathcal{M}}(\theta|\mathcal{D}) = 0$ gives us $\hat{\theta}_{\mathcal{M}}$, the maximum likelihood estimate for \mathcal{M} . Therefore, following the method suggested by Gelman (1995), estimating the solution to

$$\mathcal{S}_{\mathcal{M}}(\theta|\mathcal{D}) = 0,$$

by the solution to

$$IS_{\mathcal{M}}(\theta; v_3) = 0. \tag{7.2}$$

gives us a method of estimating $\hat{\theta}_{\mathcal{M}}$. We can solve equation (7.2) using a root finding method such as Newton-Raphson. Using Newton-Raphson requires calculation of the derivative of $IS_{\mathcal{M}}(\theta; v_3)$ at every value of $\theta \in \Theta$.

Claim 7.3. *The derivative of $IS_{\mathcal{M}}(\theta; v_3)$ is given by*

$$\frac{\partial IS_{\mathcal{M}i}}{\partial \theta_j} = \frac{1}{n_3} \sum_{k=1}^{n_3} w_k^* \left[\left(S_{kj} - \frac{1}{n_3} \sum_{h=1}^{n_3} w_h^* S_{hj} \right) S_{ki} + \frac{\partial S_{ki}}{\partial \theta_j} \right],$$

where $S_k = S_{\mathcal{M}}(\theta|v_{3k})$ and

$$w_k^* = \frac{w_k(\theta; v_3)}{\frac{1}{n_3} \sum_h w_h(\theta; v_3)}.$$

Proof. Firstly,

$$\frac{\partial w_k(\theta; v_3)}{\partial \theta_j} = S_{kj} w_k(\theta; v_3),$$

and so

$$\begin{aligned} \frac{\partial w_k^*}{\partial \theta_j} &= \frac{w'_k}{\bar{w}} - \frac{w_k}{\bar{w}^2} \left(\frac{1}{n_3} \sum_h w'_h \right) = \frac{w_k S_{kj}}{\bar{w}} - \frac{w_k}{\bar{w}^2} \left(\frac{1}{n_3} \sum_h w_h S_{hj} \right), \\ &= w_k^* \left[S_{kj} - \frac{1}{n_3} \sum_h w_h^* S_{hj} \right], \end{aligned}$$

We can rewrite $IS_{\mathcal{M}}(\theta; v_3)$ as

$$IS_{\mathcal{M}}(\theta; v_3) = \frac{1}{n_3} \sum_{k=1}^{n_3} w_k^*(\theta; v_3) S_{\mathcal{M}}(\theta|v_{3k}),$$

and so

$$\begin{aligned} \frac{\partial IS_{\mathcal{M}i}}{\partial \theta_j} &= \frac{1}{n} \sum_{k=1}^n \left[\frac{\partial w_k^*}{\partial \theta_j} S_{ki} + w_k^* \frac{\partial S_{ki}}{\partial \theta_j} \right], \\ &= \frac{1}{n} \sum_{k=1}^n w_k^* \left[\left(S_{kj} - \frac{1}{n} \sum_{h=1}^n w_h^* S_{hj} \right) S_{ki} + \frac{\partial S_{ki}}{\partial \theta_j} \right]. \end{aligned}$$

□

This means, that for any model \mathcal{M} , we can try to find an approximation of the corresponding maximum likelihood estimate $\hat{\theta}_{\mathcal{M}}$, using our single set of simulated data v_3 . Given our estimate of $\hat{\theta}_{\mathcal{M}}$, by Corollary 6.5,

$$\frac{1}{n_3} \sum_{i=1}^{n_3} w_i(\hat{\theta}_{\mathcal{M}}; v_3)$$

is an unbiased (and, given Assumption 2, strongly consistent) estimator of

$$\frac{p_{\hat{\theta}_{\mathcal{M}}}(\mathcal{D})}{p_{\hat{\theta}_0}(\mathcal{D})},$$

and so, if one of \mathcal{M}_0 and \mathcal{M} is nested in the other, we can use this estimate to perform a likelihood ratio test.

This means we can perform what we will refer to as ‘fast forward selection’: for a null hypothesis $H_0 : \mathcal{M}_0$, let $H_{11} : \mathcal{M}_1, \dots, H_{1K} : \mathcal{M}_K$, for some K , be a variety of alternative hypotheses such that, for $k \in \{1, \dots, K\}$, \mathcal{M}_0 is nested in \mathcal{M}_k . We can then apply Algorithm 3 to perform all these tests using a single set of simulated data.

Possible limitations

If the maximum likelihood estimate for an alternative hypothesis is far from that for the null hypothesis, then it is likely that, when solving the Newton-Raphson equation, the entropy of the importance sampling weights

input : Model $\mathcal{M}_0, \mathcal{M}_1, \dots, \mathcal{M}_K$, such that, for $k = 1, \dots, K$, \mathcal{M}_0 is nested in \mathcal{M}_k .

output: For $k = 1, \dots, K$, test statistic for the test $H_0 : \mathcal{M}_0, H_1 : \mathcal{M}_k$.

Estimate MLE $\hat{\theta}_0$ for \mathcal{M}_0 ;

Generate chains using $\hat{\theta}_0$, giving us N samples v ;

for $k = 1, \dots, K$ **do**

Estimate $\hat{\theta}_k$ by solving $IS_{\mathcal{M}_k}(\theta; v) = 0$;

Estimate k th test statistic by

$$\hat{T}_k = 2 \log \left(\frac{1}{N} \sum_{i=1}^N w_i(\hat{\theta}_k; v) \right).$$

end

return $\hat{T}_1, \dots, \hat{T}_K$

Algorithm 3: Fast forward selection

will become small: as described in Section 6.2.1, if this happens, we will reject the importance weights, because we do not believe an estimate obtained using them will be reliable. If this happens, we cannot find a solution, and do not obtain an estimate of the test statistic.

7.3.2 Bridge sampling

In this section, we describe performing a single likelihood ratio test, using an algorithm inspired by the methods of Gelman and Meng (1998). Let \mathcal{M}_0 and \mathcal{M}_1 denote our null and alternative models, respectively. We use the Siena algorithm to find the corresponding maximum likelihood estimates, $\hat{\theta}_0$ and $\hat{\theta}_1$. This gives us sets of data simulated from each of these

parameters:

$$v_0 = (v_{01}, \dots, v_{0n_3}) \sim \hat{\theta}_0,$$

and

$$v_1 = (v_{11}, \dots, v_{1n_3}) \sim \hat{\theta}_1.$$

Let $\bar{w}(\theta; v_j)$ denote the mean of the importance weights for the samples drawn according to $\hat{\theta}_j$, for $j = 0, 1$. Then $\bar{w}(\hat{\theta}_1; v_0)$ and $\bar{w}(\hat{\theta}_0; v_1)$ are unbiased estimators of $\frac{p_{\hat{\theta}_1}(\mathcal{D})}{p_{\hat{\theta}_0}(\mathcal{D})}$ and $\frac{p_{\hat{\theta}_0}(\mathcal{D})}{p_{\hat{\theta}_1}(\mathcal{D})}$, respectively. However, as was discussed in the previous section, if $\hat{\theta}_0$ and $\hat{\theta}_1$ are far from one another, then these estimates may not be very good. As previously described, we can use the entropy of the weights to diagnose if the estimates are likely to be poor.

In this section we apply bridge sampling, as defined by Meng and Wong (1996), to improve upon these estimates.

Single bridge sampling

Firstly, note that, for any density q , and any $\theta \in \Theta$,

$$\mathbb{E}_\theta \left(\frac{q(v)}{p_\theta(v)} \middle| \mathcal{D} \right) = \int_{v|\mathcal{D}} \frac{p_\theta(v|\mathcal{D})}{p_\theta(v)} q(v) dv = \frac{1}{p_\theta(\mathcal{D})} \int_{v|\mathcal{D}} q(v) dv,$$

and so, for any $\theta_0, \theta_1 \in \Theta$,

$$\frac{\mathbb{E}_{\theta_0} \left(\frac{q(v)}{p_{\theta_0}(v)} \middle| \mathcal{D} \right)}{\mathbb{E}_{\theta_1} \left(\frac{q(v)}{p_{\theta_1}(v)} \middle| \mathcal{D} \right)} = \frac{p_{\theta_1}(\mathcal{D})}{p_{\theta_0}(\mathcal{D})}.$$

Therefore, in this section, we consider estimating $\frac{p_{\hat{\theta}_0}(\mathcal{D})}{p_{\hat{\theta}_1}(\mathcal{D})}$ using estimators of the form

$$r(q) = \frac{\frac{1}{n_3} \sum_{k=1}^{n_3} \frac{q(v_{0k})}{p_{\hat{\theta}_0}(v_{0k})}}{\frac{1}{n_3} \sum_{k=1}^{n_3} \frac{q(v_{1k})}{p_{\hat{\theta}_1}(v_{1k})}}.$$

Considering independent samples, Gelman and Meng (1998) show that the optimal choice for q is given by

$$q_{\text{opt}}(v) \propto \frac{p_{\hat{\theta}_1}(v)p_{\hat{\theta}_0}(v)}{p_{\hat{\theta}_1}(v) + p_{\hat{\theta}_0}(v)}.$$

We refer to this as the *bridge density*.

Proposition 7.4. *Given Assumption 2,*

$$r(q_{\text{opt}}) = \frac{\frac{1}{n_3} \sum_{k=1}^{n_3} \frac{q_{\text{opt}}(v_{0k})}{p_{\hat{\theta}_0}(v_{0k})}}{\frac{1}{n_3} \sum_{k=1}^{n_3} \frac{q_{\text{opt}}(v_{1k})}{p_{\hat{\theta}_1}(v_{1k})}}, \quad (7.5)$$

is a strongly consistent estimator of

$$\frac{p_{\hat{\theta}_0}(\mathcal{D})}{p_{\hat{\theta}_1}(\mathcal{D})}.$$

Proof. Let C be the normalising constant for q_{opt} . Then $q_{\text{opt}} \leq Cp_{\hat{\theta}_0}$ and $q_{\text{opt}} \leq Cp_{\hat{\theta}_1}$. For $j = 0, 1$, let $F_j(v) = q_{\text{opt}}/p_{\hat{\theta}_{1-j}}$. Then, by Theorem 6.1, and setting $\theta = \hat{\theta}_{1-j}$,

$$\frac{1}{n_3} \sum_{k=1}^{n_3} \frac{q_{\text{opt}}(v_{jk})}{p_{\hat{\theta}_j}(v_{jk})} \xrightarrow{p} \mathbb{E}_{\hat{\theta}_{1-j}} \left(\frac{q_{\text{opt}}(V)}{p_{\hat{\theta}_{1-j}}(V)} \middle| \mathcal{D} \right) = \frac{1}{p_{\hat{\theta}_{1-j}}(\mathcal{D})} \int_{v|\mathcal{D}} q_{\text{opt}}(v) dv$$

Therefore, by the continuous mapping theorem,

$$r(q_{\text{opt}}) \xrightarrow{p} \frac{p_{\hat{\theta}_0}(\mathcal{D})}{p_{\hat{\theta}_1}(\mathcal{D})}.$$

□

We therefore can estimate the likelihood ratio using $r(q_{\text{opt}})$. The method used to apply single bridge sampling is summarised in Algorithm 4.

The intuition that the estimate $r(q_{\text{opt}})$ should be better than either $\bar{w}(\hat{\theta}_1; v_0)$ or $\bar{w}(\hat{\theta}_0; v_1)$ comes from the fact that q_{opt} should be ‘closer’ in some sense to both $p_{\hat{\theta}_1}$ or $p_{\hat{\theta}_0}$ than they are to each other. However, if $\hat{\theta}_1$ and $\hat{\theta}_0$ are very far apart, then even q_{opt} may be too far from $p_{\hat{\theta}_1}$ or $p_{\hat{\theta}_0}$. We can diagnose this by looking at the entropy of either $w(\hat{\theta}_1; v_0)$ or $w(\hat{\theta}_0; v_1)$. If we decide that the entropy is too low, then we will need to consider multiple bridge densities.

Multiple bridge sampling

Firstly, let us generalise the ratio $r(q)$ given in equation (7.5): for two sets of data v_1 and v_2 , simulated according to parameters θ_1 and θ_2 , respectively, define a ratio given by

$$r(v_1, v_2) = \frac{r_{\text{top}}(v_1)}{r_{\text{bottom}}(v_2)}, \quad (7.7)$$

input : Models \mathcal{M}_0 and \mathcal{M}_1 such that \mathcal{M}_0 is nested in \mathcal{M}_1 .
output: Test statistic for the test $H_0 : \mathcal{M}_0, H_1 : \mathcal{M}_1$.
 Estimate MLEs $\hat{\theta}_0$ and $\hat{\theta}_1$ for models \mathcal{M}_0 and \mathcal{M}_1 , respectively;
for $j \in \{0, 1\}$ **do**
 | Generate chains using parameter $\hat{\theta}_j$, giving us N samples v_j ;
end
 Estimate T by

$$\hat{T} = -2 \log(r),$$

where

$$r = \frac{\frac{1}{n_3} \sum_{i=1}^{n_3} \frac{p_{\hat{\theta}_1}(v_{0i})}{p_{\hat{\theta}_0}(v_{0i}) + p_{\hat{\theta}_1}(v_{0i})}}{\frac{1}{n_3} \sum_{i=1}^{n_3} \frac{p_{\hat{\theta}_0}(v_{1i})}{p_{\hat{\theta}_0}(v_{1i}) + p_{\hat{\theta}_1}(v_{1i})}}. \quad (7.6)$$

return \hat{T}

Algorithm 4: Single bridge sampling

where

$$r_{\text{top}}(v_1) = \frac{1}{n_3} \sum_{i=1}^{n_3} \frac{p_{\theta_2}(v_{1i})}{p_{\theta_1}(v_{1i}) + p_{\theta_2}(v_{1i})},$$

and

$$r_{\text{bottom}}(v_2) = \frac{1}{n_3} \sum_{i=1}^{n_3} \frac{p_{\theta_1}(v_{2i})}{p_{\theta_1}(v_{2i}) + p_{\theta_2}(v_{2i})}.$$

Then, if we assume sufficient mixing of the simulated chains, by Proposition 7.4 this is a strongly consistent estimator of

$$\frac{p_{\theta_1}(\mathcal{D})}{p_{\theta_2}(\mathcal{D})}.$$

Therefore, for a sequence of parameters $\theta_1, \dots, \theta_K$, for some K , suppose we simulate corresponding independent sets of sufficiently mixed data

v_1, \dots, v_K . Then we can combine the ratios in what Gelman and Meng (1998) describe as a ‘telescoping fashion’:

$$\prod_{i=1}^{K-1} r(v_i, v_{i+1}).$$

This is then a strongly consistent estimator of

$$\frac{p_{\theta_1}(\mathcal{D})}{p_{\theta_K}(\mathcal{D})}.$$

Hence choosing $\theta_1 = \hat{\theta}_0$ and $\theta_K = \hat{\theta}_1$ (or vice versa) will give us a likelihood ratio estimator.

For the remaining parameters $\{\theta_k : 1 < k < K\}$, it makes sense intuitively to choose them to lie on a path from $\hat{\theta}_0$ to $\hat{\theta}_1$: for such a path $\theta(t)$, we will choose

$$\theta_k = \theta(t_k),$$

where $0 < t_2 < \dots < t_K < 1$. In this Chapter, we propose choosing t_2, \dots, t_K by considering entropy: at the j th iteration of our algorithm, given chains v_j simulated using θ_j , we choose t_{j+1} to be as large as possible while keeping the entropy of the importance weights $w(\theta(t_{j+1}); v_j)$ sufficiently high (see Algorithm 5 for details). This algorithm requires the choice of two tuning parameters: c and N , where c is the minimum entropy allowed, and N is the number of samples that we draw to make up each set of chains.

input : Models \mathcal{M}_0 and \mathcal{M}_1 such that \mathcal{M}_0 is nested in \mathcal{M}_1 .

output: Test statistic for the test $H_0 : \mathcal{M}_0, H_1 : \mathcal{M}_1$.

Estimate MLEs $\hat{\theta}_0$ and $\hat{\theta}_1$ for models \mathcal{M}_0 and \mathcal{M}_1 , respectively;

Set $j = 1, \theta^* = \hat{\theta}_1, t_{\max} = 0, z = 1$;

while $t_{\max} < 1$ **do**

Generate chains using parameter θ^* , giving us N samples v^* ;

if $j > 1$ **then**

Set

$z = z/r_{\text{bottom}}(v^*)$

end

Use interval bisection to approximate t_{\max} , the largest $t \in [0, 1]$ such that $\text{Entropy}(w(\theta(t); v^*)) > c$, for some constant c , where

$\theta(t) = t\hat{\theta}_0 + (1 - t)\theta^*$.

(Here we make the assumption that $\text{Entropy}(w(\theta(t); v^*))$ is roughly increasing in t);

Set

$z = zr_{\text{top}}(v^*)$,

and $\theta^* = \theta(t_{\max})$;

Set $j = j + 1$;

end

Generate chains using parameter $\hat{\theta}_0$, giving us N samples v^* ;

Set

$z = z/r_{\text{bottom}}(v^*)$.

Estimate T by

$\hat{T} = -2 \log(z)$.

return \hat{T}

Algorithm 5: Multiple bridge sampling

7.3.3 Thermodynamic integration

We can estimate the test statistic by using an identity used in thermodynamic integration (Neal, 1993), given by

$$\log \left(\frac{p_{\hat{\theta}_1}(\mathcal{D})}{p_{\hat{\theta}_0}(\mathcal{D})} \right) = \int_0^1 \frac{d\theta}{dt} \cdot S(\theta(t)|\mathcal{D}) dt,$$

where $\theta(t)$ is a path from $\hat{\theta}_0$ to $\hat{\theta}_1$. If we consider the path $\theta(t) = t\hat{\theta}_1 + (1 - t)\hat{\theta}_0$, then

$$\log \left(\frac{p_{\theta_1}(\mathcal{D})}{p_{\theta_0}(\mathcal{D})} \right) = (\theta_1 - \theta_0) \cdot \int_0^1 S(\theta(t)|\mathcal{D}) dt, \quad (7.8)$$

$$\approx \frac{1}{H+1} (\theta_1 - \theta_0) \cdot \sum_{h=0}^H S(\theta(h/H)|\mathcal{D}). \quad (7.9)$$

Snijders et al. (2010) execute a likelihood ratio test for Siena models using this approximation, by estimating $S(\theta(h/H)|\mathcal{D})$ by drawing samples from $p_{\theta(h/H)}(\cdot|\mathcal{D})$; however, this is computationally expensive, especially if H is not very small.

Instead, we propose using importance sampling: for $h = 0, \dots, H$, we can, by Corollary 6.6, consistently estimate $S(\theta(h/H)|\mathcal{D})$ by

$$r(h; v_3) = \frac{\frac{1}{n_3} \sum_{k=1}^{n_3} w_k(\theta(h/H); v_3) S(\theta(h/H)|v_{3k})}{\frac{1}{n_3} \sum_{k=1}^{n_3} w_k(\theta(h/H); v_3)},$$

where v_3 is data simulated according to some parameter $\theta \in \Theta$ (in practice, either $\hat{\theta}_0$ or $\hat{\theta}_1$).

As with bridge sampling, estimates may become poor as we move away from the parameter used to simulate the chains, so we can apply Algorithm 6. This algorithm requires the choice of three tuning parameters: H , c and N .

7.4 Evaluating the difficulty of the test

In Section 7.5, we will perform simulation studies and conduct likelihood ratio tests on simulated data. Before we simulate data from a specified model and set of parameters, we would like to be assess how difficult it will be for an algorithm to detect the true model. For example, if we were to include an effect, but choose the associated parameter value to be very small in size, then it might be impossible to detect that we included that effect in the true model. To address this, we consider the *relative importance* of the included effects, using the method developed by Indlekofer and Brandes (2013). For each effect, a measure is constructed by considering how conditional choice probabilities for a mini-step change when that effect is removed from the model. These are normalised across all effects to give a measure of relative importance.

We apply the following algorithm before performing simulation studies: for each model under consideration, we

1. estimate the parameters given this model, using the observed data and method of moments estimation;

input : Models \mathcal{M}_0 and \mathcal{M}_1 such that \mathcal{M}_0 is nested in \mathcal{M}_1 .

output: Test statistic for the test $H_0 : \mathcal{M}_0, H_1 : \mathcal{M}_1$.

Estimate MLEs $\hat{\theta}_0$ and $\hat{\theta}_1$ for models \mathcal{M}_0 and \mathcal{M}_1 , respectively;

Set $j = 1, \theta^* = \hat{\theta}_1, h_{\min} = 0, h_{\max} = 0$;

while $h_{\max} < H$ **do**

Generate chains using parameter θ^* , giving us N samples v^* ;

if $j > 1$ **then**

Set

$$z = z / r_{\text{bottom}}(v^*)$$

end

Use interval bisection to approximate h_{\max} , the largest integer $h \in \{h_{\min}, h_{\min} + 1, \dots, H\}$ such that $\text{Entropy}(w(\theta(h/H); v^*)) > c$, for some constant c , where

$$\theta(t) = t\hat{\theta}_0 + (1 - t)\theta^*.$$

(Here we make the assumption that $\text{Entropy}(w(\theta(t); v^*))$ is roughly increasing in t);

Store

$$z_j = \frac{1}{H + 1}(\theta_1 - \theta_0) \cdot \sum_{h=h_{\min}}^{h_{\max}} r(h; v^*).$$

if $h_{\max} < H$ **then**

Set $h_{\min} = h_{\max} + 1, \theta^* = \theta(h_{\max}/H)$ and $j = j + 1$.

end

end

Estimate T by

$$\hat{T} = \sum_j z_j.$$

return \hat{T}

Algorithm 6: Thermodynamic integration

Table 7.1: Effects included in each model (density and reciprocity included in every model).

Model	Effects			
	TT	3C	B	DS
1	×	×	×	×
2	×	×	×	
3	×	×		×
4	×	×		
5	×		×	×
6	×		×	
7	×			×
8	×			
9		×	×	×
10		×	×	
11		×		×
12		×		
13			×	×
14			×	
15				×
16				

- calculate the relative importance of the effects, R_1, \dots, R_p , where p is the number of effects in the model.
- If $\min_j \{R_j\} < c/p$, for some constant $c < 1$, then we say that at least one of the effects has too small relative importance. We decide that it is unlikely that the true model will be detected if we performed a simulation study, so we do not use this model as a generating model.

In the simulations in Section 7.5, all the models we consider have outdegree and reciprocity effects, and the remaining effects in a model will lie within the set {transitive triplets, 3-cycles, betweenness, drinking similarity}. This gives us 16 potential models, the order of which is given in Table 7.1.

Table 7.2: Relative importance of effects.

Model	p_{eval}	Effects				$p_{\text{eval}} \min_j \{R_j\}$
		TT	3C	B	DS	
1	6	0.11	0.04	0.19	0.08	0.22
2	5	0.12	0.04	0.2		0.20
3	5	0.15	0.01		0.09	0.05
4	4	0.16	0.01			0.03
5	5	0.15		0.19	0.07	0.37
6	4	0.16		0.2		0.62
7	4	0.16			0.09	0.36
8	3	0.17				0.51
9	5		0.13	0.22	0.08	0.40
10	4		0.13	0.22		0.54
11	4		0.15		0.08	0.34
12	3		0.16			0.48
13	4			0.42	0.14	0.55
14	3			0.46		0.69
15	3				0.11	0.32

In Table 7.2 we show the relative importance for the effects (we do not show those for outdegree and reciprocity, as in all cases they never have minimum importance).

Choosing, say, $c = 0.5$ as the minimum value allowed for $p_{\text{eval}} \min_j \{R_j\}$, indicates that for models 6, 8, 10, 13 and 14, all effects are important enough that we hope that they could be detected in a test.

7.5 Simulation studies

In this section we will consider data simulated from Model 8 as the effects for this models have large enough minimum relative importance that we

hope that it will be possible for a test to detect the true model.

We firstly consider using fast forward selection, since this is the quickest of our three methods. We will then apply the other two methods to tests where fast forward selection did not perform well.

7.5.1 Fast forward selection

Model 8 is nested in Models 1 to 7, while only Model 16 is nested in Model 8. Therefore, to examine the type 1 and type 2 errors of the test, we consider performing the following tests: for $j = 1, \dots, 7$,

$$H_0 : \text{Model 8}, H_1 : \text{Model } j,$$

and

$$H_0 : \text{Model 16}, H_1 : \text{Model 8}.$$

We generate 50 datasets according to Model 8, and, for each simulated dataset, attempt to perform the eight tests using fast forward selection.

Recall that a potential problem with fast forward selection is that we may fail to find an estimate using the Newton-Raphson method, either because the method does not converge to a solution, or because the entropy becomes too small. So, for each of the tests, we firstly want to see the percentage of datasets for which we succeed in estimating the test statistic; these results are given in Table 7.3. We see that 6 of the 8 tests have good success rates at solving the Newton-Raphson equation and finding an estimate of

Table 7.3: Fast forward selection with $\theta_0 = (6.2, -2.50, 2.11, 0.58, 0, 0, 0)$.

Test	H_0	H_1	d.o.f.	Estimated (%)
1	8	1	3	64
2	8	2	2	78
3	8	3	2	80
4	8	4	1	82
5	8	5	2	80
6	8	6	1	86
7	8	7	1	98
8	16	8	1	22

the test statistic (75% or higher). The first test, which has the largest degree of freedom between null and alternative models, has a lower success rate of estimation: intuitively, it is plausible that having more effects in the alternative model means the estimation will be more difficult. The final test, where the null hypothesis is false, has a low success rate.

Of the cases where we do successfully obtain a test statistic, we want to look at the distribution of the test statistic, and the quality of the results. In Figure 7.1, we compare the empirical distribution of the test statistics with the corresponding χ_k^2 distributions (where k is the relevant degrees of freedom); in these tests the null hypothesis is true, and so if our estimation of the test statistics are good, then we would expect the empirical and theoretical distributions to look similar.

Another way of assessing the performance of the test is by finding the type 1 error under different rejection criteria (i.e. different significance levels). Figure 7.2 shows the type 1 error (estimated by proportion of rejected null hypotheses) for significance levels between 0 and 0.2, for the first seven

tests. Because the null hypotheses are true and the empirical distributions of the test statistics should be close to the relevant χ_k^2 distributions (where k is the relevant degrees of freedom), the type 1 error should be close to the significance level. If, for a particular test and significance level, the type 1 error is higher, it indicates an overly *aggressive* test, where the true null hypothesis is rejected too often; conversely, if the type 1 error is lower, it indicates an overly *conservative* test, where the true null hypothesis is rejected too rarely. For the first two tests, the results are quite good, with type 1 errors close to the significance levels; however, in other cases, the test seems quite variable, with results indicating that for some hypotheses it will be overly aggressive and, for others, overly conservative.

Figure 7.3 shows the empirical distribution of the test statistic compared to the χ_1^2 distribution, and the power of the final test for different significance levels between 0 and 0.002, where by power we mean the proportion of cases where the null hypothesis is correctly rejected (because, for this test, it is not true). Although we saw in Table 7.3 that, for this test, the estimation of the test statistic had a low success rate, with only 22%, Figure 7.3 shows that, conditional on the estimation being successful, the test statistic almost always takes values much higher than would be expected for a χ_1^2 distributed random variable, and thus the test has very high power.

These results show that, if we perform fast forward selection and the Newton-Raphson procedure is successful then the test results have high power, but variable type 1 error. Although the type 1 errors are not very close to the significance levels, given the fast forward selection is so quick to perform

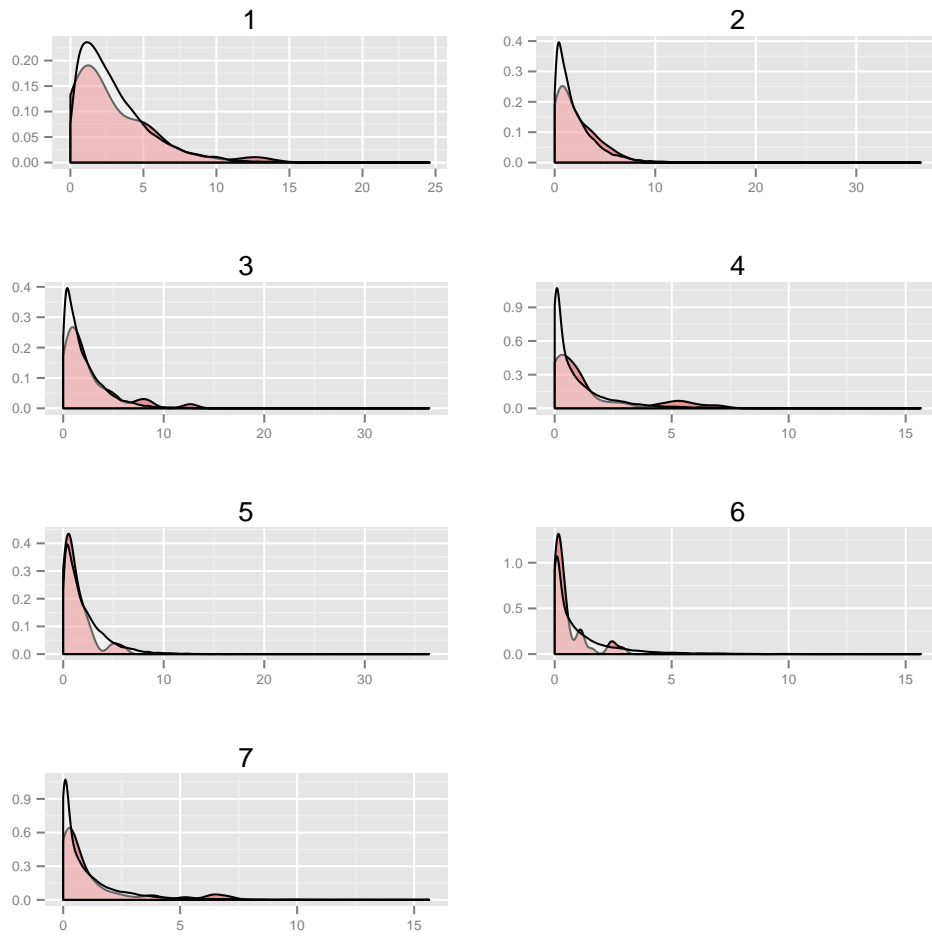


Figure 7.1: Fast forward selection; empirical distribution of test statistics are plotted in red; translucent distributions show the corresponding χ_k^2 distribution, where k is the degrees of freedom of the test.

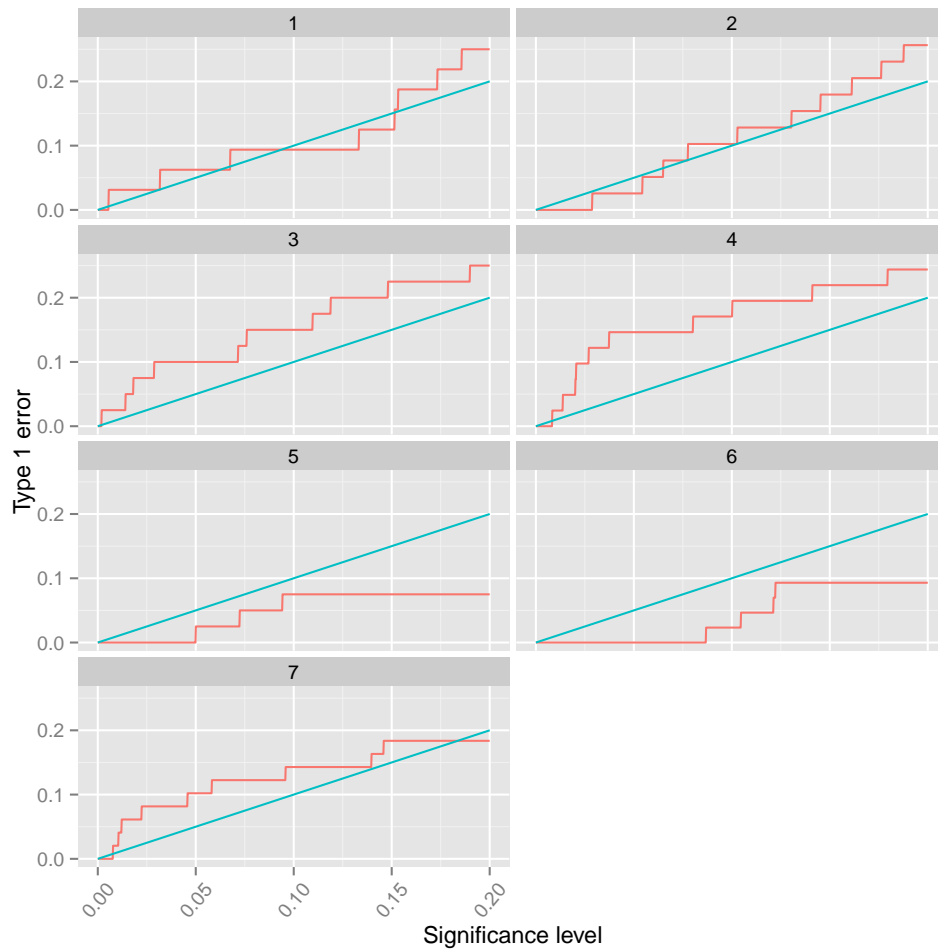


Figure 7.2: Fast forward selection; type 1 errors for different significance levels, for tests where the null hypothesis is true.

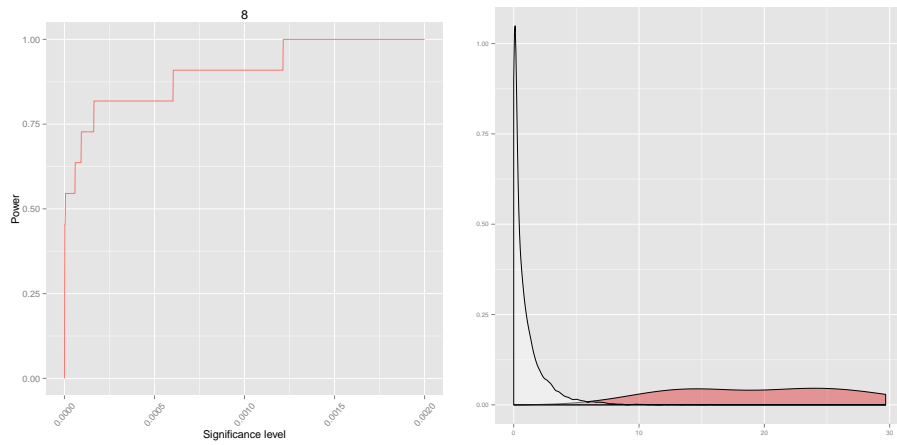


Figure 7.3: Fast forward selection; (left) power for different significance levels, for the test where the null hypothesis is false; (right) empirical distribution of test statistics are plotted in red, and translucent distributions shows the χ_1^2 distribution.

(if we use the simulations obtained in Phase 3 of the parameter estimation algorithm), they are close enough that it could still be worth employing this method, but taking a cautious approach to the results obtained. For example, if we wanted to perform a test using a particular significance level then if our test result gave a test statistic with a p value very far from this (either much smaller or much larger), we may be happy to accept the result, but if it is at all close to the significance level, then we may need to decide that the test is inconclusive. Moreover, if the estimation of the test statistic fails, we will need to try one of the other two, more time-consuming, methods.

7.5.2 Bridge sampling and thermodynamic integration

We now consider performing Test 1,

$$H_0 : \text{Model 8}, H_1 : \text{Model 1},$$

and Test 8,

$$H_0 : \text{Model 16}, H_1 : \text{Model 8},$$

using both bridge sampling and thermodynamic integration. As before, we consider data generated from Model 8, so that, in the first case, the null hypothesis is true, whilst in the second, it is false.

Model 8 vs. Model 1

In this section, we will use both bridge sampling and thermodynamic integration to perform the test

$$H_0 : \text{Model 8}, H_1 : \text{Model 1},$$

since estimating test statistics in this case seemed quite difficult with fast forward selection. Recall that we need to choose tuning parameters c and N to perform bridge sampling, and c , N , and H to perform thermodynamic integration.. In this section, we consider $c \in \{0, 0.4, 0.8\}$, $N \in \{100, 500, 1000\}$ and $H \in \{10, 20\}$, giving us 9 different bridge sampling estimates and 18 different thermodynamic integration estimates for each dataset. We use 100 simulated datasets.

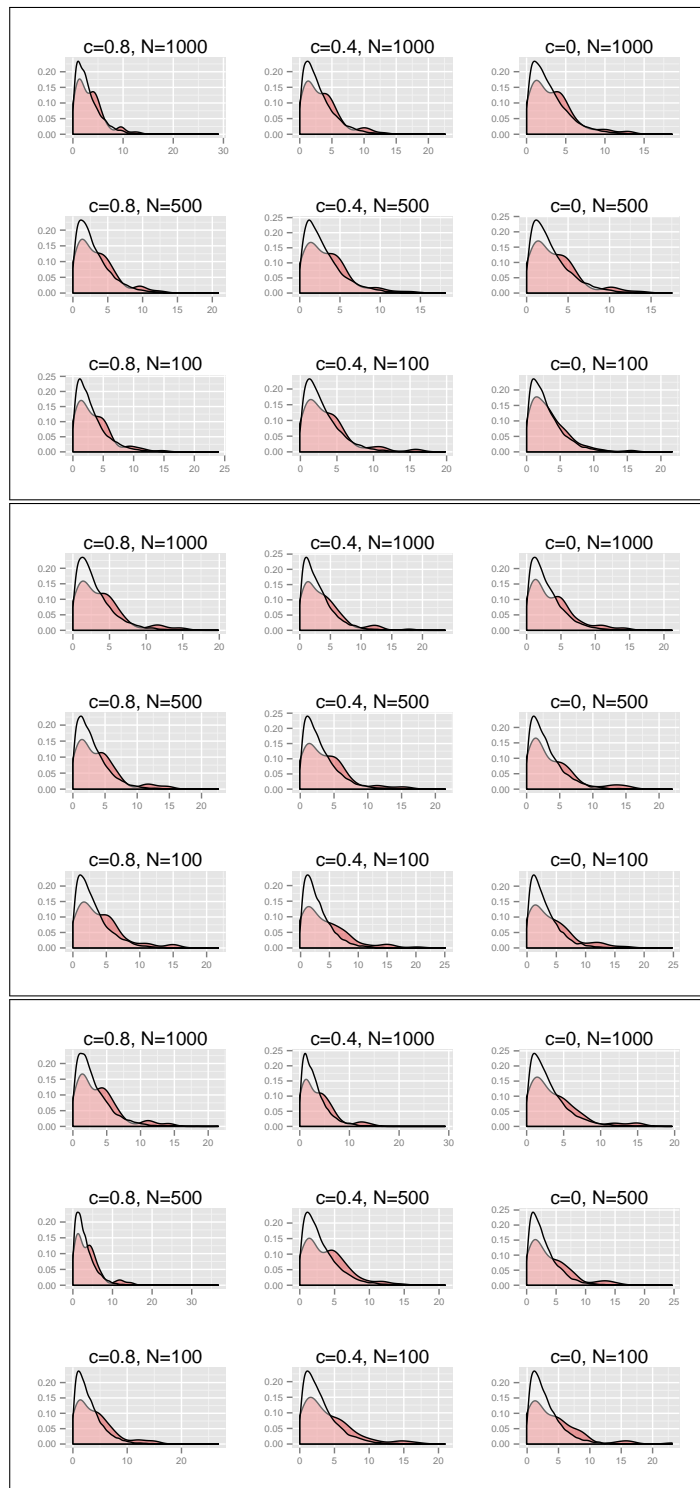


Figure 7.4: For different values of (c, N) , the empirical distribution of test statistics is shown in red; translucent distributions show the corresponding χ_k^2 distribution, where k is the degrees of freedom of the test: (top) bridge sampling; (middle) thermodynamic integration, $H = 10$; (bottom) thermodynamic integration, $H = 20$.

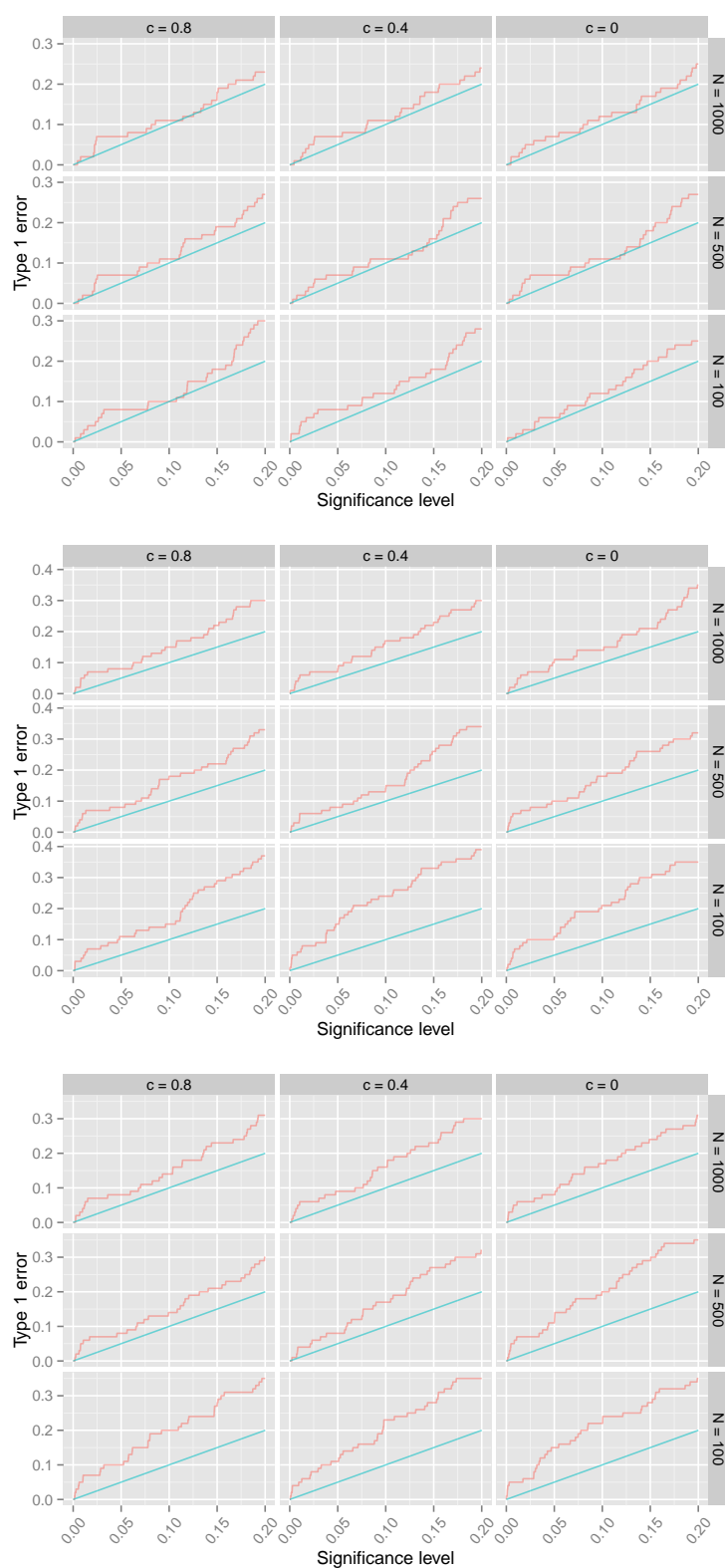


Figure 7.5: Type 1 errors for different significance levels, using different values of (c, N) and: (top) bridge sampling; (middle) thermodynamic integration, $H = 10$; (bottom) thermodynamic integration, $H = 20$.

In Figure 7.4, we see how the empirical distribution of the test statistics compares to the χ_1^2 distribution. Because the null hypothesis is true, we would hope these to be similar: there is only small differences between graphs, but we see across all estimates there is a slight tendency for the test statistic to be larger than we would expect.

Figure 7.5 shows the type 1 error for different significance levels between 0 and 0.2, for all estimates. We see overall that the test is too aggressive, with slightly too high type 1 errors, and too many rejected null hypothesis; however, we see that bridge sampling generally gives the best results, with type 1 errors closest to the significance levels. Looking at the bridge sampling results, the results are not very sensitive to the choice of c and N (although the results are slightly better with $N = 1000$). Looking at the thermodynamic integration results, it is surprising to see that increasing H from 10 to 20 does not seem to improve results. We also see that results are more sensitive to the choice of c and N , with the test becoming aggressive as you decrease either.

In Figure 7.6 we plot the times taken to estimate the test statistics; the times include the time taken to generate all simulated chains, and to calculate likelihoods, and finally to calculate the test statistics, but *not* the time taken to estimate the maximum likelihood estimates for both the null and alternative models; this means, in practice, given two nested models of interest, the total time taken to estimate parameters and perform would be the times stated plus twice the time taken to perform Phases 1 and 2 (but not Phase 3) of the Siena estimation procedure (twice because we are

considering two models).

As we would expect, computation times are increasing with increases in c , N , and H . Generally, bridge sampling is the slowest, taking slightly longer than thermodynamic integration with $H = 20$, and around twice as long as thermodynamic integration with $H = 10$.

Looking at all the plots together, we would probably conclude that bridge sampling is the best method; given a (c, N) pair, the time taken is longer, however, since the type 1 errors are better, we can choose a small (c, N) pair with bridge sampling if we want to reduce computation times, and still achieve better type 1 errors than using thermodynamic integration with a higher (c, N) pair.

Model 16 vs. Model 8

Since, for the previous test, bridge sampling seemed to give the best results, we will now use it to perform the test

$$H_0 : \text{Model 16}, H_1 : \text{Model 8},$$

since using fast forward selection was not very successful (because the Newton-Raphson method usually failed to find a solution). As with the previous test, we consider $c \in \{0, 0.4, 0.8\}$ and $N \in \{100, 500, 1000\}$, giving us 9 different bridge sampling estimates. We use 50 simulated datasets.

Figure 7.7 shows, for different values of c and N , the empirical distribution

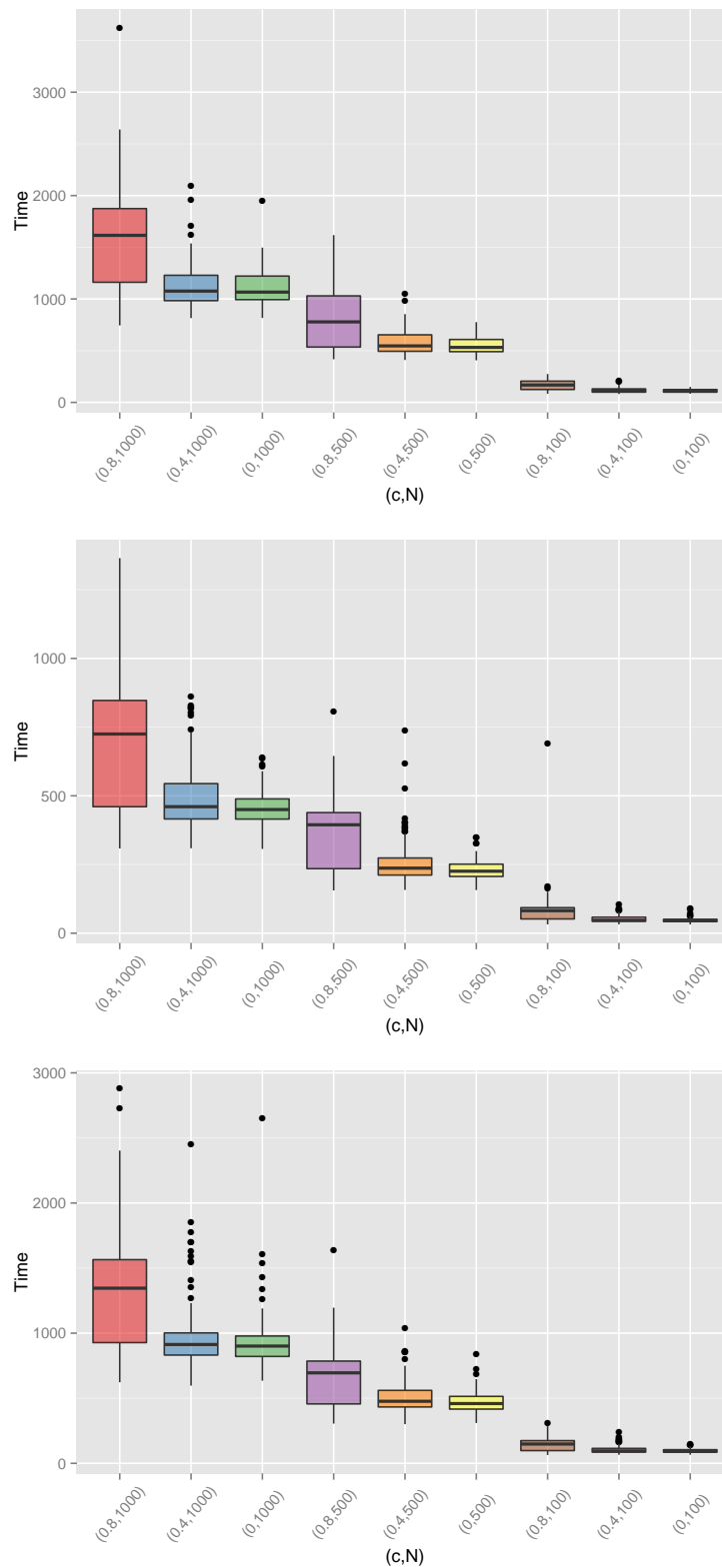


Figure 7.6: Time taken to perform test for different values of (c, N) and: (top) bridge sampling; (middle) thermodynamic integration, $H = 10$; (bottom) thermodynamic integration, $H = 20$.

of the test statistics, the power for different significance levels between 0 and 0.005, and the times taken to estimate the test statistics. The times taken are similar to those seen in the previous test. As we saw with fast forward selection, the power of the test is very high; we are able to reject the test for 100% of the simulated datasets, using any pair of (c, N) , with a significance level of less than 0.0025.

7.6 Discussion

In this chapter we proposed three methods for estimating the test statistic required to perform a likelihood ratio test. We saw that the first method, fast forward selection, enables us to perform the test very quickly, but does not have very accurate results, and quite often will fail to find the solution required to actually estimate the statistic. Given its speed, it could be worth using were we in an exploratory mode, and we wanted to quickly compare many models.

Of the other two methods, multiple bridge sampling seemed to outperform thermodynamic integration. Even with small numbers for the tuning parameters (c, N) , the power of the tests was very high, and the type 1 errors close to their theoretical counterparts.

Unlike fast forward selection, it is not possible to perform multiple tests simultaneously (because we are required to simulate data from the both of the parameters specified by the null and alternative hypotheses), but as long as we keep the number of models under consideration fairly small,

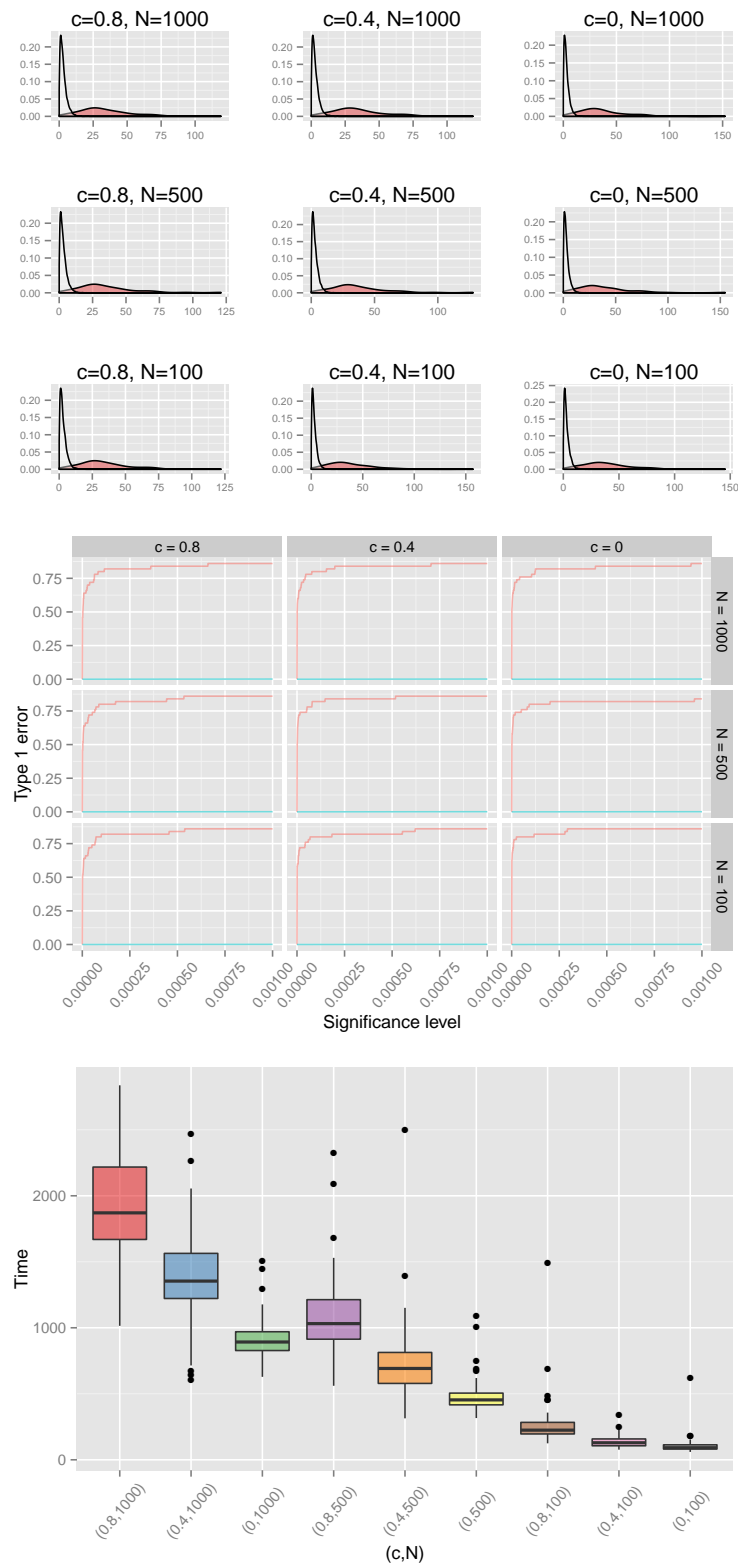


Figure 7.7: Using bridge sampling and different values of (c, N) : (top) empirical distribution of test statistics compared to χ_1^2 distribution; (middle) power for different significance levels; (bottom) time taken to estimate test statistics.

this method should provide accurate estimates of the test statistics, enabling us to compare different models, thus giving us an effective method for performing model selection.

Chapter 8

Standard error estimation

8.1 Introduction

Accurate estimation of the covariance matrix for the maximum likelihood estimate is very important: it is necessary for detecting high correlations between different model effects, and for finding reliable standard errors for the parameters, which can be used to perform significance tests (as is done in Section 3.5). However, in the current `RSiena` estimation procedure, it can also be very time-consuming when using maximum likelihood estimation, with a large amount of simulated data required for stable results.

In this chapter we propose a method for improved estimation of the covariance matrix for the maximum likelihood estimate. Recall from Section 2.3.1 that this matrix is estimated using data simulated in Phase 3 of

the estimation procedure. In this chapter, we propose using importance sampling to incorporate the data simulated in Phase 2 into our estimates (while still using the Phase 3 data), and show that this leads to improved performance.

In Section 8.2, we will briefly review how the covariance matrix is currently estimated in `RSiena`, before proposing estimates using importance sampling and the data simulated in the different subphases of Phase 2. We will then propose an approximate way of combining this estimates into one final estimate. In Section 8.3 we will apply the method to a real dataset, and show that the proposed estimate has an improved performance, in terms of bias and variance.

8.2 Covariance matrix estimation

Recall from Section 6.1 that we denote the observation of the data by $\mathcal{D} = \{(X(t_0) = x(t_0), X(t_1) = x(t_1))\}$. Recall further that we generate many chains of ministeps, after conditioning on \mathcal{D} , so that for each chain the observation of the data holds, and the state of the process is correct at the observation times. In Phase 3, we know our final parameter estimate, and generate n_3 chains, so that, for $i \in \{1, \dots, n_3\}$

$$v_{3i} \sim p_{\hat{\theta}}(\cdot | \mathcal{D}).$$

Recall that the covariance matrix for a maximum likelihood estimator can

be approximated by the inverse of the observed information (Efron and Hinkley, 1978). The observed information can be expressed (Orchard et al., 1972) as the difference between two positive definite symmetric matrices A_θ and B_θ , so that

$$-\frac{dS(\theta|\mathcal{D})}{d\theta} = A_\theta - B_\theta,$$

where

$$A_\theta(V) = \mathbb{E}_\theta \left(-\frac{dS(\theta|V)}{d\theta} \middle| \mathcal{D} \right),$$

and

$$B_\theta(V) = \text{Cov}_\theta \left(S(\theta|V) \middle| \mathcal{D} \right).$$

We can obtain Monte Carlo estimates of A_θ and B_θ using the chains simulated in Phase 3; we denote their difference, which is our estimate of the observed information, by $\hat{\Sigma}_0$.

8.2.1 Estimating the observed information in Phase 2

Recall from Section 6.1 that, in Phase 2, we simulate chains according to different parameter values: we have n_{sub} subphases, and in the k th subphase, we have a sequence of parameter values $\theta_{k1}, \dots, \theta_{kn_2}$ for some n_{2k} , and for each one, we simulate a single chain: for $i \in \{1, \dots, n_{2k}\}$,

$$v_{2ki} \sim p_{\theta_{ki}}(\cdot|\mathcal{D}).$$

In this section we will suggest assumptions under which we can construct strongly consistent estimators of the observed information, using the simulations from Phase 2. Recall from Section 2.3.1 that at each step we update the parameter using our Robbins-Monro iterative scheme, and so, for $i \in \{2, \dots, n_{2k}\}$, the parameter θ_{ki} depends on both v_{2ki-1} and θ_{ki-1} . This means that, prior to estimation, we consider each parameter as a random variable. Because the parameters are no longer a fixed constant value, we need to modify the assumptions and results from Chapter 6 showing strong consistency of importance sampling estimators.

Assumption 3. *The maximum value of an effect, for any state of the network, is bounded: there exists finite S such that, if \mathcal{P} is the set of effect indices,*

$$\max_{\{i \in \mathcal{N}, j \in \mathcal{P}, X \in \mathcal{X}\}} \{|s_{ij}(X)|\} < S.$$

Note that this assumption is satisfied by all effects considered in this thesis; for the remainder of this part we assume that Assumption 3 holds.

Assumption 4. *The parameter space Θ is a closed and bounded interval, with the total rate parameter bounded away from zero: $L_{\min} < \lambda < L_{\max}$, for finite L_{\max} and $L_{\min} > 0$.*

Lemma 8.1. *There exists $d_{\min} > 0$ such that*

$$\inf_{\theta \in \Theta} p_{\theta}(\mathcal{D}) > d_{\min}.$$

Proof. Firstly, Assumptions 3 and 4 imply that there exists a constant $C < \infty$ such that, for any $x_1, x_2 \in \mathcal{X}$, and any β such that $(\rho, \beta) \in \Theta$,

$$\exp(\beta^T(s_i(x_1) - s_i(x_2))) < C.$$

Then the probability of a minstep given a change opportunity is bounded below by $1/nC := m_{\min} < \infty$. This means that the probability of a chain of length k is bounded below by m_{\min}^k .

Now, let k_{\min} be the shortest length of a chain satisfying the data \mathcal{D} .

Then, for $\theta \in \Theta$,

$$\begin{aligned} p_{\theta}(\mathcal{D}) &= \sum_{k=k_{\min}}^{\infty} \mathbb{P}_{\theta}(M = k) \mathbb{P}(\mathcal{D} | M = k), \\ &\geq \sum_{k=k_{\min}}^{\infty} \frac{(\lambda m_{\min})^k e^{-\lambda}}{k!}, \\ &= e^{-L_{\max}(1-m_{\min})} \mathbb{P}(W \geq k_{\min}), \end{aligned} \tag{8.2}$$

where $W \sim \text{Pois}(L_{\min} m_{\min})$, by Assumption 4. \square

Assumption 5. The Markov chain $((V_{2kj}, \Theta_{kj}))_{j \geq 1}$ of chains and parameter values generated in the k th subphase is irreducible and positive recurrent.

Theorem 8.3. Let

$$F(\theta|v) = - \left(\frac{dS(\theta|v)}{d\theta} + S(\theta|V)[S(\theta|V) - S(\theta|\mathcal{D})]^T \right).$$

Given Assumptions 3,4 and 5, for every $\theta \in \Theta$,

$$IS(\theta) = \frac{A(\theta)}{B(\theta)}, \quad (8.4)$$

where

$$A(\theta) = \frac{1}{n_{2k}} \sum_{i=1}^{n_{2k}} w_i(\theta; v_{2k}) F(\theta|v_{2ki}),$$

and

$$B(\theta) = \frac{1}{n_{2k}} \sum_{i=1}^{n_{2k}} w_i(\theta; v_{2k}),$$

is a strongly consistent estimator of the observed information $-\frac{dS(\theta|\mathcal{D})}{d\theta}$.

Proof. Let Θ_i denote the (random variable) parameter used to simulate the i th chain in Subphase k .

For $i = 1, \dots, n_{2k}$, let $A_i(\theta) = w_i(\theta; v_{2k}) F(\theta|v_{2ki})$, so that

$$A(\theta) = \frac{1}{n_{2k}} \sum_{i=1}^{n_{2k}} A_i(\theta).$$

Following a similar method as in Theorem 6.1,

$$\begin{aligned}
 \mathbb{E}(A_i(\theta)|\mathcal{D}) &= \mathbb{E}_{\Theta_i} [\mathbb{E}(A_i(\theta)|\mathcal{D}, \Theta_i)], \\
 &= \mathbb{E}_{\Theta_i} \left[\frac{p_\theta(\mathcal{D})}{p_{\Theta_i}(\mathcal{D})} \mathbb{E}(F(\theta|V)|\mathcal{D}) \right], \\
 &= \mathbb{E}_{\Theta_i} \left(\frac{1}{p_{\Theta_i}(\mathcal{D})} \right) p_\theta(\mathcal{D}) \mathbb{E}_\theta(F(\theta|V)|\mathcal{D}), \tag{8.5}
 \end{aligned}$$

and, by Lemma 8.1,

$$\mathbb{E}(A_i(\theta)|\mathcal{D}) \leq \frac{1}{d_{\min}} \mathbb{E}_\theta(F(\theta|V)|\mathcal{D}) < \infty.$$

Then, by Assumption 5 and the Birkhoff ergodic theorem,

$$A(\theta) \xrightarrow{a.s.} \mathbb{E}_\pi \left(\frac{1}{p_\pi(\mathcal{D})} \right) p_\theta(\mathcal{D}) \mathbb{E}_\theta(F(\theta|V)|\mathcal{D}),$$

where π is the invariant distribution of the Markov chain. Similarly,

$$B(\theta) \xrightarrow{a.s.} \mathbb{E}_\pi \left(\frac{1}{p_\pi(\mathcal{D})} \right) p_\theta(\mathcal{D}),$$

and so, by the continuous mapping theorem,

$$IS(\theta) \xrightarrow{a.s.} \mathbb{E}_\theta(F(\theta|V)|\mathcal{D}) = -\frac{d\mathcal{S}(\theta|\mathcal{D})}{d\theta}.$$

□

Theorem 8.3 shows that we can construct an estimator of the observed information in Subphase k , for $k = 1, \dots, n_{\text{sub}}$, given by

$$\hat{\Sigma}_k = IS(\hat{\theta}),$$

using the function defined in equation (8.4).

8.2.2 Combining the estimates

We now have $n_{\text{sub}} + 1$ estimators for the observed information, denoted by $\hat{\Sigma}_0, \dots, \hat{\Sigma}_{n_{\text{sub}}}$. We would like to combine these to create an overall estimator which will hopefully be more efficient than $\hat{\Sigma}_0$.

We will consider a linear combination

$$\hat{\Sigma}(\alpha) = \alpha_0 \hat{\Sigma}_0 + \dots + \alpha_{n_{\text{sub}}} \hat{\Sigma}_{n_{\text{sub}}},$$

where $\sum_i \alpha_i = 1$.

Proposition 8.6. *An approximately optimal choice for $\alpha_0, \dots, \alpha_{n_{\text{sub}}}$ is given by*

$$\alpha_k \propto \frac{1}{\sum_{i,j} \text{Var}(\hat{\Sigma}_{kij})},$$

for $k = 0, \dots, n_{\text{sub}}$, where optimal means that $\mathbb{E}(\|\hat{\Sigma}(\alpha) - \Sigma\|^2)$ is minimised, where Σ is the true observed information and $\|\cdot\|$ denotes the Frobenius norm.

Sketch proof. Let

$$f(\alpha) = \mathbb{E} \left(\|\hat{\Sigma}(\alpha) - \Sigma\|^2 \right) = \sum_{i,j=1}^p f_{ij}(\alpha),$$

where $\alpha = (\alpha_0, \dots, \alpha_{n_{\text{sub}}})$ and

$$f_{ij}(\alpha) = \mathbb{E} \left(\left[\hat{\Sigma}(\alpha)_{ij} - \Sigma_{ij} \right]^2 \right).$$

Then, for all $i, j = 1 \dots, p$,

$$f_{ij}(\alpha) = \text{Var} \left(\hat{\Sigma}(\alpha)_{ij} \right) + \left[\mathbb{E} \left(\hat{\Sigma}(\alpha)_{ij} - \Sigma_{ij} \right) \right]^2, \quad (8.7)$$

$$\approx \text{Var} \left(\hat{\Sigma}(\alpha)_{ij} \right), \quad (8.8)$$

by the strong consistency of $\hat{\Sigma}(\alpha)$. Now, we assume that the dependency between $\hat{\Sigma}_l$ and $\hat{\Sigma}_k$, when $l \neq k$, is small, and so

$$f_{ij}(\alpha) \approx \sum_{k=1}^{n_{\text{sub}}} \alpha_k^2 \text{Var} \left(\hat{\Sigma}_{kij} \right). \quad (8.9)$$

Then

$$f(\alpha) \approx \sum_{i,j,k} \alpha_k^2 \text{Var} \left(\hat{\Sigma}_{kij} \right). \quad (8.10)$$

Then the approximately optimal choice for α can be found by minimis-

ing the Lagrangian

$$\Lambda(\alpha, \lambda) = \lambda \left(1 - \sum_k \alpha_k \right) + \sum_{i,j,k} \alpha_k^2 \text{Var} \left(\hat{\Sigma}_{kij} \right),$$

which gives us that

$$\alpha_k \propto \frac{1}{\sum_{i,j} \text{Var} \left(\hat{\Sigma}_{kij} \right)}.$$

□

We construct jackknife estimates of the variances: for $i, j = 1, \dots, p$,

$$\widehat{\text{Var}} \left(\hat{\Sigma}_{kij} \right) = \frac{n-1}{n} \sum_{h=1}^{n_{2k}} \left(\hat{\Sigma}_{Jkij}^{(-h)} - \hat{\Sigma}_{Jkij} \right)^2,$$

where $\hat{\Sigma}_{Jk}^{(-h)}$ is what we would obtain for $\hat{\Sigma}_k$ were the h th observation removed, and

$$\hat{\Sigma}_{Jk} = \frac{1}{n_{2k}} \sum_{h=1}^{n_{2k}} \hat{\Sigma}_{Jk}^{(-h)}.$$

Then our final estimate of the observed information is given by:

$$\hat{\Sigma} = \frac{\sum_{k=0}^{n_{\text{sub}}} \frac{\hat{\Sigma}_k}{\sum_{i,j=1}^p \widehat{\text{Var}} \left(\hat{\Sigma}_{kij} \right)}}{\sum_{k=0}^{n_{\text{sub}}} \frac{1}{\sum_{i,j=1}^p \widehat{\text{Var}} \left(\hat{\Sigma}_{kij} \right)}}.$$

8.2.3 Practical Implications

In the top plot of Figure 8.1 we see an example of how the estimate of one component of θ varies throughout the estimation procedure: we see that in early subphases, the estimate varies a lot, while in Subphase 4, it does not (and in Phase 3, it is fixed at its final value). Looking at the early fluctuations, we may have doubts about the quality of the estimate of the information at the early subphases. However, as we see in the bottom plot of Figure 8.1, our choice of α means that the early subphases have little contribution to the final estimate, and so hopefully potential inaccuracies in early subphases will not cause problems in the final estimate.

8.3 Example: *s50* data

We consider the first two observations of the *s50* dataset, and fit a model with outdegree, reciprocity, transitive triplets, 3-cycles, betweenness, drinking ego, drinking alter, and drinking similarity. We estimate the model parameters and covariance matrix (with the latter being estimated in the usual way, using simulations from Phase 3 only), and repeat this $N = 100$ times. We obtain a final estimate of the covariance matrix by averaging the 100 estimates of the information matrix (because each estimate is approximately unbiased, assuming that the parameter estimates are close to the maximum likelihood estimate) and then taking the inverse: we denote the final overall estimate by \hat{V} . Each time, we estimate the covariance matrix using $n_3 = 2000$ Phase 3 iterations, but, in this section, we consider

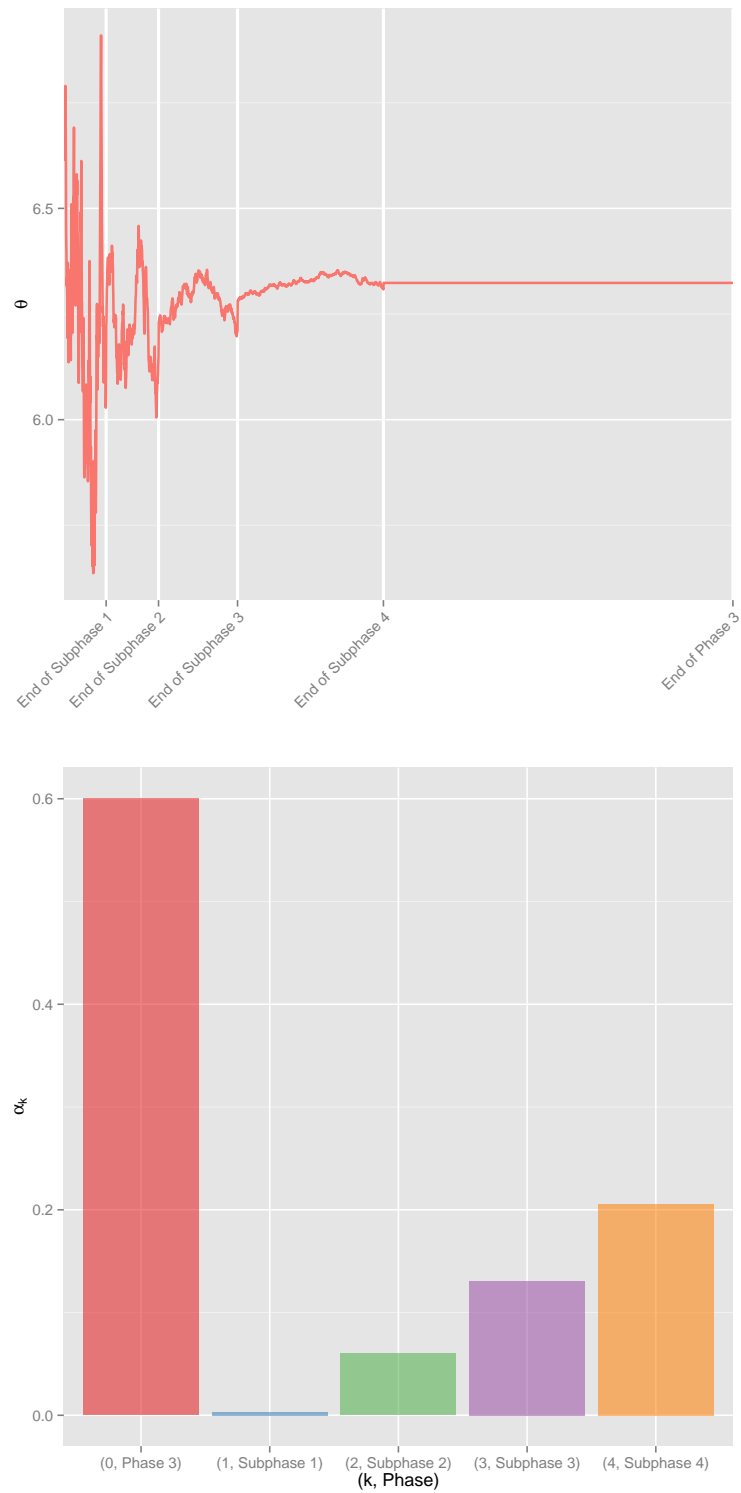


Figure 8.1: Top plot shows how the estimate of one component of θ (the rate parameter) varies throughout the estimation procedure, in Phases 2 and 3; bottom plot shows the corresponding values of $(\alpha_0, \dots, \alpha_4)$.

the results had we stopped Phase 3 at different values of n_3 . To compare the performance of the original method with the method proposed in this chapter, we consider these questions:

1. For a given n_3 , does the proposed method improve estimation of the covariance matrix, and if so, by how much?
2. How much time can we save? If the proposed method does perform better, how much can we reduce the number of Phase 3 simulations, while still obtaining a comparable performance to the original method?

8.3.1 Improving performance

In this section, we assess how much the proposed method improves estimation of the covariance matrix, for different values of n_3 . In Figures 8.2, 8.3, 8.4 and 8.5, we see the densities of the estimates of the variances of each of the parameter (i.e. the diagonals of the covariance matrix). We see that the bias (compared to our overall best estimates) and variance are reduced using the new method. As we would expect, the change is largest for smaller values of n_3 . Recall from Section 8.2 that the original method estimates the observed information by the difference of two positive definite matrices; this can lead to negative estimates of the variances for the parameters (as can be seen for $n_3 = 500$ and $n_3 = 1000$, in Figures 8.2 and 8.3), which is clearly undesirable. Using the new method, for each choice of n_3 , and each parameter, no variance estimate is negative.

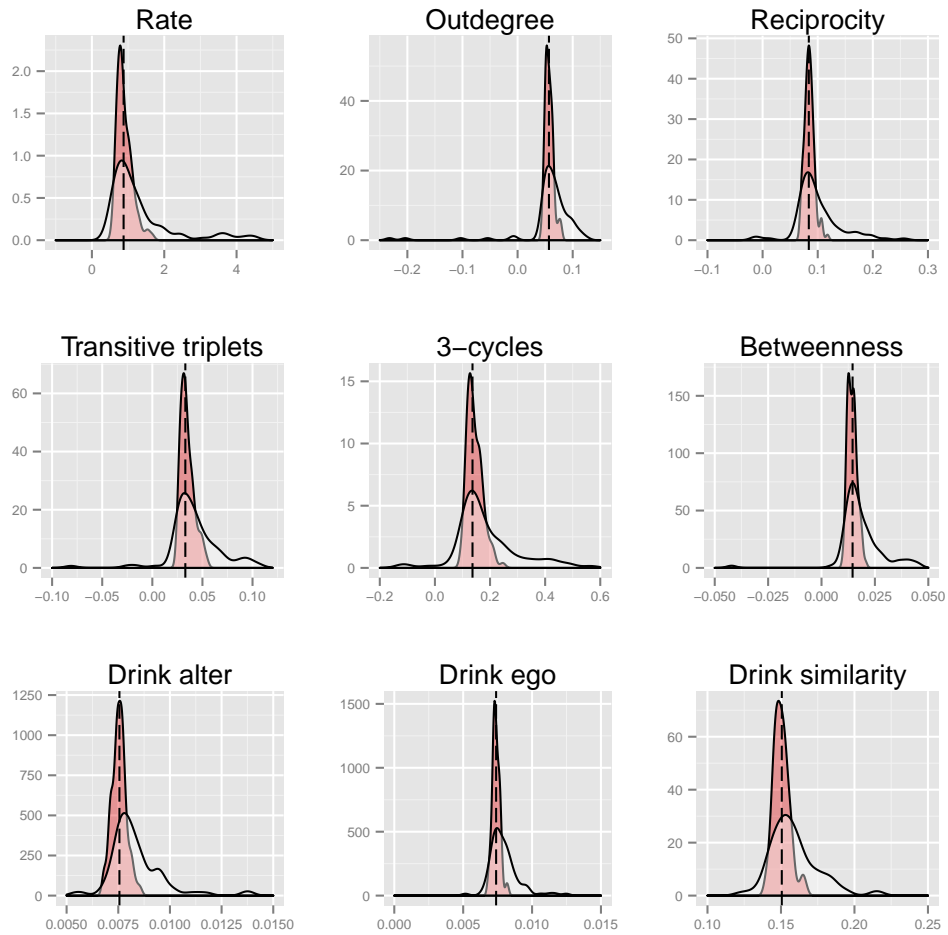
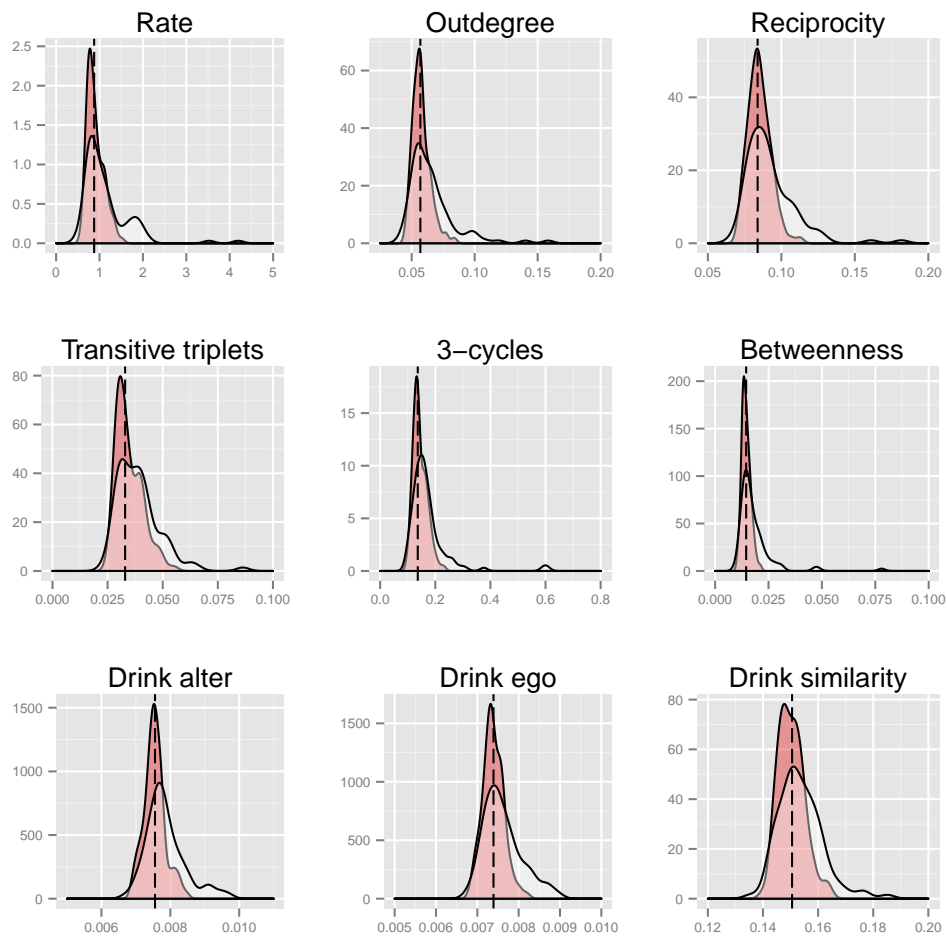
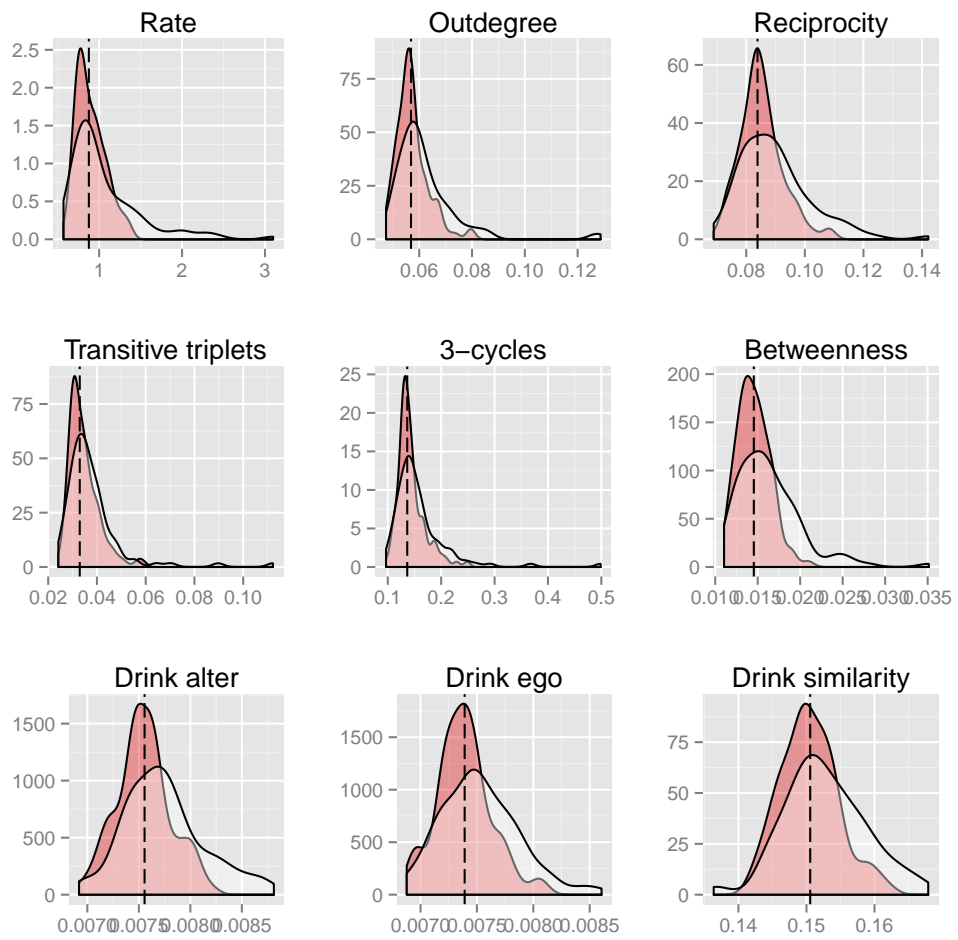
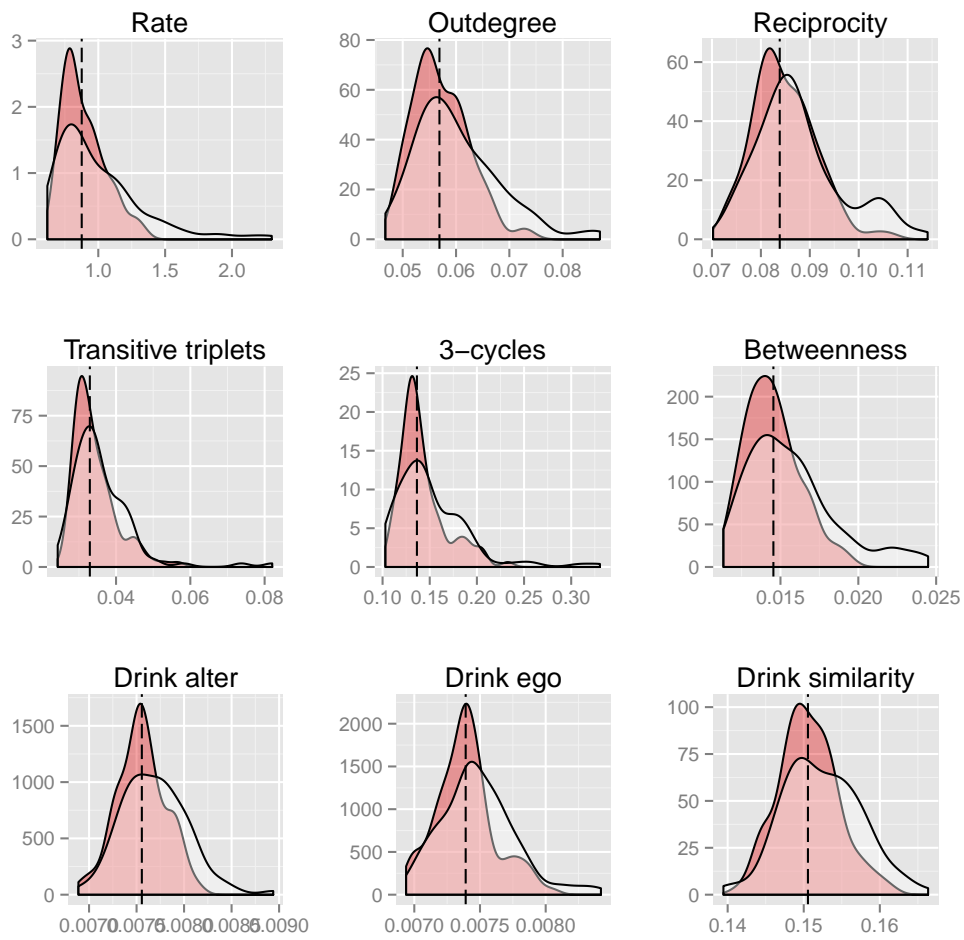


Figure 8.2: $n_3 = 500$; the density of the estimates of the variance (results using the new proposed method are shown in red, and the translucent distribution shows results using Phase 3 simulations only). The vertical line shows the overall best estimate, obtained by taking the inverse of the average of the information matrices across all $N = 100$ repetitions.

Figure 8.3: $n_3 = 1000$.

Figure 8.4: $n_3 = 1500$.

Figure 8.5: $n_3 = 2000$.

In Figure 8.6, we see how the ratio between root mean squared error for the new method compared to the original method varies with n_3 , for all 81 components of the covariance matrix. We would expect to see that the ratios are close to zero for very small n_3 , since the original method has a very small number of simulations to use for estimation, and so should perform badly. As $n_3 \rightarrow \infty$, we would expect to see that the ratios would tend to 1, as more weight will be placed on the Phase 3 simulations, and the contributions of the Phase 2 simulations will be unimportant. In practice, it seems like the ratios will converge to 1 quite slowly: at $n_3 = 1000$, almost all of the 81 components have a root mean squared error that is at least halved, and even at $n_3 = 2000$, the ratios are still between less than 0.5 and about 0.75. We can conclude from these results that using the new method improves upon the original method, with a very large improvement when n_3 is quite small.

8.3.2 Saving time

In Section 8.3.1, we saw that the bias, variance and root mean squared error was reduced using the new method, for values of n_3 up to 2000. In this section, we consider how much time could be saved using the new method. For a given n_3 , let $M_{n_3}^{\text{orig}}$ and $M_{n_3}^{\text{new}}$ denote the matrices of root mean squared errors for the components of the covariance matrix using

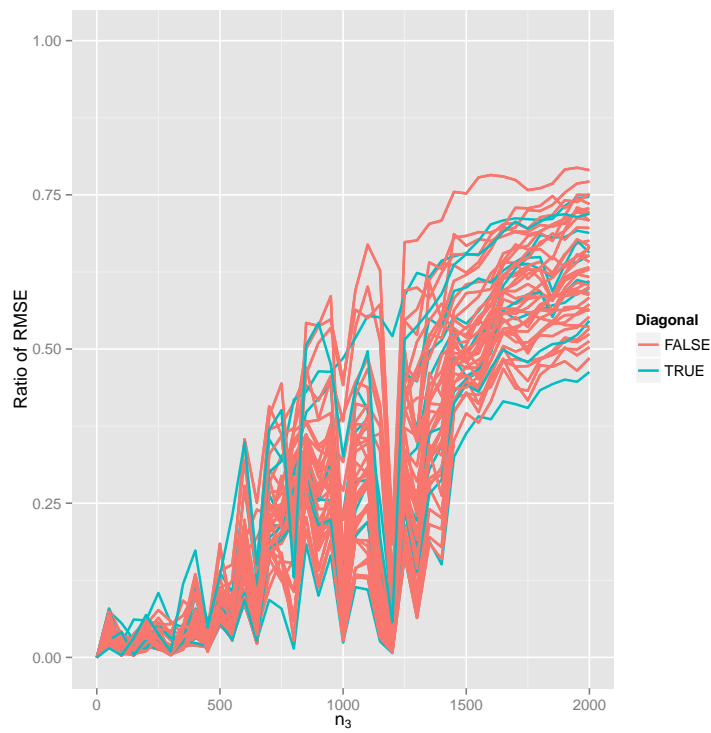


Figure 8.6: Ratio of root mean squared error for all components of the covariance matrix against the number of simulations in Phase 3, n_3 .

the original and new method, respectively. Now, let $\hat{n}_3(n_3)$ be defined as:

$$\hat{n}_3(n_3) = \min_{n \geq 50} \left\{ n : \max_{i,j} \left\{ \frac{(M_n^{\text{new}})_{ij}}{(M_{n_3}^{\text{orig}})_{ij}} \right\} \leq 1 \right\}.$$

So $\hat{n}_3(n_3)$ is the smallest number of simulations needed such that the root mean squared error using the new method is smaller than the root mean squared error using the original method with n_3 simulations, for all components of the covariance matrix. In other words, if you were going to use the original method, and n_3 simulations in Phase 3, this number tells you how many simulations you would need to use with the new method, without (hopefully) making the root mean squared error increase. Figure 8.7 shows the results for different values of n_3 . The graph shows that, for n_3 smaller than about 1400, the root mean squared error is smaller using the new method with only 50 simulations in Phase 3. For larger n_3 , this increases, but still $\hat{n}_3(n_3)$ is always much smaller than n_3 : for $n_3 = 2000$, it is still less than 1000. This shows that substantial time can be saved in Phase 3 using this new method.

8.4 Discussion

In this chapter, we have seen that we can use chains simulated in Phase 2 to improve estimation of the covariance matrix for the maximum likelihood estimate. We used importance sampling to construct estimates of the information matrix using Phase 2 simulations, and have shown that in-

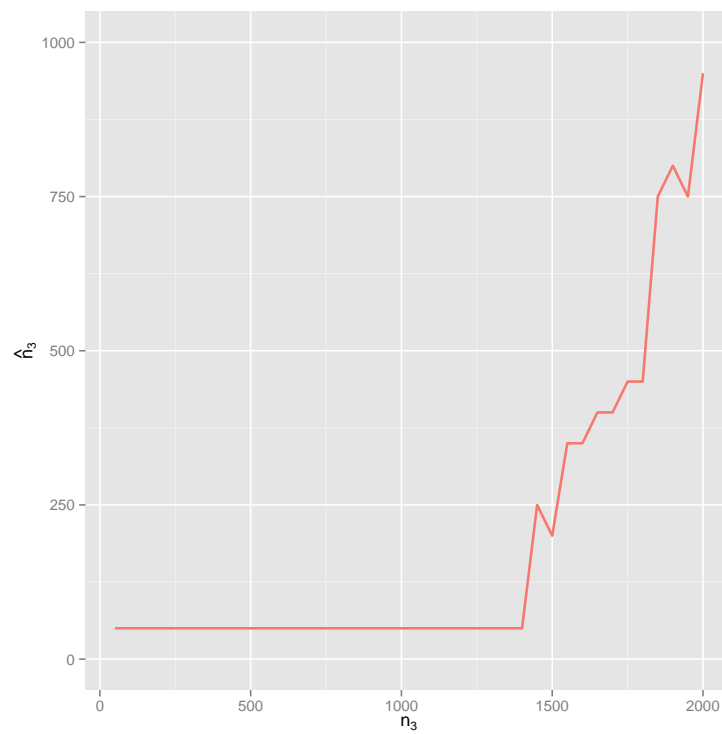


Figure 8.7: Proposed \hat{n}_3 to obtain a smaller root mean squared error, for all components of the covariance matrix, using the new method, than that obtained using n_3 and Phase 3 simulations only.

incorporating these estimates into the final covariance matrix estimate may reduce mean squared error, especially when the number of Phase 3 iterations is quite small. We could use this in two ways: firstly, by keeping the number of Phase 3 iterations the same, we can expect an improvement in estimation of the covariance using this new method. Secondly, if we would like the estimation procedure to be faster, using this method it appears that we can still get a reasonable estimate with a very small number of Phase 3 iterations. The latter could be very beneficial if we want to explore various models, and we are interested in quickly seeing rough approximations of standard errors.

In Section 8.2.3, we suggested that information estimates from the early Phase 2 subphases may be inaccurate, but argued that due to the fact they would only contribute little to the final estimate, this would not matter too much. It would be interesting to compare our results to those obtained by just excluding early subphases completely.

Bibliography

Allee, W. C. (1958). *The social life of animals*. Beacon Press Boston, MA.

Allison, P. D. (1984). *Event history analysis: regression for longitudinal event data*. London: SAGE.

Andersen, P. K. (1993). *Statistical models based on counting processes*. New York: Springer.

Barnes, J. A. (1954). *Class and committees in a Norwegian island parish*. Plenum.

Bartholomew, D. J. (1967). *Stochastic Models for Social Processes*. Chicester: Wiley.

Bauman, K. E. and S. T. Ennett (1996). On the importance of peer influence for adolescent drug use: commonly neglected considerations. *Addiction* 91(2), 185–198.

Behrman, J. R., H. Kohler, and S. Watkins (2002). Social networks and changes in contraceptive use over time: Evidence from a longitudinal study in rural Kenya. *Demography* 39(4), 713–738.

- Billingsley, P. (1995). *Probability and Measure*. New York: Wiley-Interscience.
- Bishop, Y., S. Fienberg, and P. Holland (1975). *Discrete multivariate analysis: theory and practice*.
- Blossfeld, H. (2002). *Techniques of event history modeling: new approaches to causal analysis*. London: Routledge.
- Blossfeld, H. (2005). *Globalization, uncertainty and youth in society*. London: Routledge.
- Bowman, K. O. and L. R. Shenton (1985). Method of moments. In *Encyclopedia of Statistical Sciences*, Volume 5. Chicester: Wiley.
- Cappé, O., E. Moulines, and T. Rydén (2005). *Inference in hidden Markov models*. New York: Springer.
- Carrington, P. J., J. Scott, and S. Wasserman (2005). *Models and methods in social network analysis*, Volume 28. Cambridge university press.
- Casella, G. and E. I. George (1992). Explaining the gibbs sampler. *The American Statistician* 46(3), 167–174.
- Coleman, J. S., E. Katz, and H. Menzel (1966). *Medical innovation: A diffusion study*. Indianapolis: Bobbs-Merrill Co.
- Cox, D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)* 34(2), 187–220.

- Davis, J. A. (1970). Clustering and hierarchy in interpersonal relations: Testing two graph theoretical models on 742 sociomatrices. *American Sociological Review*, 843–851.
- Doreian, P. and F. Stokman (2013). *Evolution of Social Networks*. Routledge Contemporary Human Geography. Taylor & Francis.
- Efron, B. and D. V. Hinkley (1978). Assessing the accuracy of the maximum likelihood estimator: Observed versus expected fisher information. *Biometrika* 65(3), 457–483.
- Faust, K. (2014). Animal social networks. In P. J. C. John Scott (Ed.), *The SAGE Handbook of Social Network Analysis*, pp. 148–167. SAGE Publications Ltd.
- Fisher, L. A. and K. E. Bauman (1988). Influence and selection in the friend-adolescent relationship: Findings from studies of adolescent smoking and drinking. *Journal of Applied Social Psychology* 18(4), 289–314.
- Gelman, A. (1995). Method of moments using Monte Carlo simulation. *Journal of Computational and Graphical Statistics* 4(1), 36–54.
- Gelman, A. and X. Meng (1998). Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical science*, 163–185.
- Giddings, F. H. (1896). *The principles of sociology: An analysis of the phenomena of association and of social organization*. Macmillan.

- Gilks, W. R., S. Richardson, and D. J. Spiegelhalter (1996). *Markov Chain Monte Carlo in Practice*. London: Chapman & Hall.
- Ginsberg, M. (1939). The problems and methods of sociology. In F. Bartlett, M. Ginsberg, E. Lindgren, and R. Thouless (Eds.), *The Study of Society (RLE Social Theory): Methods and Problems*, pp. 436–478. Routledge.
- Godambe, V. P. (1960). An optimum property of regular maximum likelihood estimation. *The Annals of Mathematical Statistics* 31(4), 1208–1211.
- Greenan, C. C. (2015). Diffusion of innovations in dynamic networks. *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 178(1), 147–166.
- Gu, M. G. and F. H. Kong (1998). A stochastic approximation algorithm with markov chain monte-carlo method for incomplete data estimation problems. *Proceedings of the National Academy of Sciences* 95(13), 7270–7274.
- Hanneman, R. A. and M. Riddle (2011). Concepts and measures for basic network analysis. In J. Scott, P. J. Carrington, and J. Scott (Eds.), *The SAGE Handbook of Social Network Analysis*, pp. 340–369. London: SAGE.
- Hoff, P. (2009). *A First Course in Bayesian Statistical Methods*. Springer Texts in Statistics. Springer New York.
- Holland, P. W. and S. Leinhardt (1977). A dynamic model for social networks. *Journal of Mathematical Sociology* 5(1), 5–20.

- Indlekofer, N. and U. Brandes (2013). Relative importance of effects in stochastic actor-oriented models. *Network Science* 1(03), 278–304.
- Iyengar, R., C. V. den Bulte, and T. W. Valente (2011). Opinion leadership and social contagion in new product diffusion. *Marketing Science* 30(2), 2195–212.
- Kass, R. E., B. P. Carlin, A. Gelman, and R. M. Neal (1998). Markov chain monte carlo in practice: a roundtable discussion. *The American Statistician* 52(2), 93–100.
- Kermack, W. O. and A. G. McKendrick (1927). A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 115(772), 700–721.
- Kirke, D. M. (2004). Chain reactions in adolescents cigarette, alcohol and drug use: similarity through peer influence or the patterning of ties in peer networks? *Social Networks* 26(1), 3–28.
- Knoke, D. and S. Yang (2008). *Social Network Analysis*. Number no. 154 in Quantitative Applications in the Social Sciences. SAGE Publications.
- Kong, A. (1992). A note on importance sampling using standardized weights. *University of Chicago, Dept. of Statistics, Tech. Rep 348*.
- Maddala, G. S. (1983). *Limited-dependent and Qualitative Variables in Econometrics*. Cambridge: Cambridge University Press.
- Marsden, P. V. (2002). Egocentric and sociocentric measures of network centrality. *Social networks* 24(4), 407–422.

- Marx, K. (1939). *Grundrisse der Kritik der politischen Ökonomie*. Europäische Verlags-Anstalt.
- McPherson, M., L. Smith-Lovin, and J. M. Cook (2001). Birds of a feather: Homophily in social networks. *Annual review of sociology*, 415–444.
- Meng, X.-L. and W. H. Wong (1996). Simulating ratios of normalizing constants via a simple identity: a theoretical exploration. *Statistica Sinica* 6(4), 831–860.
- Myers, D. J. (2000). The diffusion of collective violence: Infectiousness, susceptibility and mass media networks. *American Journal of Sociology* 106(1), 173–208.
- Neal, R. M. (1993). Probabilistic inference using markov chain monte carlo methods.
- Neyman, J. and E. Pearson (1933). On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society* 231, 289–337.
- Neyman, J. and E. S. Pearson (1928). On the use and interpretation of certain test criteria for purposes of statistical inference: Part i. *Biometrika*, 175–240.
- Norris, J. R. (1997). *Markov Chains*. Cambridge: Cambridge University Press.
- O’Quigley, J. (2008). *Proportional hazards regression*. New York: Springer.

- Orchard, T., M. A. Woodbury, et al. (1972). A missing information principle: theory and applications. In *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Theory of Statistics*. The Regents of the University of California.
- Pearson, M. and L. Michell (2000). Smoke rings: social network analysis of friendship groups, smoking and drug-taking. *Drugs: Education, Prevention, and Policy* 7(1), 21–37.
- Pearson, M., C. Steglich, and T. Snijders (2006). Homophily and assimilation among sport-active adolescent substance users. *Connections* 27(1), 47–63.
- Perrin, P. G. (1955). 'pecking order' 1927–54. *American Speech*, 265–268.
- Rhue, L. and A. Sundararajan (2014). Digital access, political networks and the diffusion of democracy. *Social Networks* 36, 40–53.
- Ripley, R. M., T. A. B. Snijders, and P. P. Lopez (2011). *Manual for SIENA version 4.0*. Department of Statistics, Oxford.
- Robbins, H. and S. Monro (1951). A stochastic approximation method. *The Annals of Mathematical Statistics* 22(3), 400–407.
- Robins, G. and P. Pattison (2005). Interdependencies and social processes: Dependence graphs and generalized dependence structures. In P. J. Carrington, J. Scott, and S. Wasserman (Eds.), *Models and Methods in Social Network Analysis*, pp. 192–214. Cambridge: Cambridge University Press.
- Roethlisberger, F. and W. Dickson (1939). Management and the worker.

- Rogers, E. M. (1995). *Diffusion of innovations* (Fourth ed.). New York: Simon and Schuster.
- Scott, J. (1988). Social network analysis. *Sociology* 22(1), 109–127.
- Shalizi, C. R. and A. C. Thomas (2011). Homophily and contagion are generically confounded in observational social network studies. *Sociological Methods Research* 40(2), 211–239.
- Silvapulle, M. J. and J. Burridge (1986). Existence of maximum likelihood estimates in regression models for grouped and ungrouped data. *Journal of the Royal Statistical Society, Series B* 48(1), 100–106.
- Snijders, T. A. B. (2001). The statistical evaluation of social network dynamics. In M. Sobel and M. Becker (Eds.), *Sociological Methodology*, pp. 361–395. London: Blackwell.
- Snijders, T. A. B., J. Koskinen, and M. Schweinberger (2010). Maximum likelihood estimation for social network dynamics. *The Annals of Applied Statistics* 4(2), 567–588.
- Snijders, T. A. B., C. E. G. Steglich, and M. Schweinberger (2007). Modeling the co-evolution of networks and behavior. In K. van Montfort, H. Oud, and A. Satorra (Eds.), *Longitudinal models in the behavioral and related sciences*, pp. 41–71. Mahwah NJ: Lawrence Erlbaum.
- Snijders, T. A. B., G. G. van de Bunt, and C. E. G. Steglich (2010). Introduction to stochastic actor-based models for network dynamics. *Social Networks* 32(1), 44–60.

- Sorensen, D. and D. Gianola (2002). *Likelihood, Bayesian, and MCMC Methods in Quantitative Genetics*. Statistics for Biology and Health. Springer.
- Steglich, C., T. A. B. Snijders, and M. Pearson (2010). Dynamic networks and behavior: Separating selection from influence. *Sociological Methodology* 40(1), 329–393.
- Strang, D. (1991). Adding social structure to diffusion models: An event history framework. *Sociological Methods and Research* 19, 324–353.
- Strang, D. and S. A. Soule (1998). Diffusion in organizations and social movements: From hybrid corn to poison pills. *Annual Review of Sociology* 24, 265–290.
- Strang, D. and N. B. Tuma (1993). Spatial and temporal heterogeneity in diffusion. *American Journal of Sociology* 99(3), 614–639.
- Suhov, Y. and M. Kelbert (2008). *Probability and Statistics by Example: Markov Chains: A primer in random processes and their applications*. Cambridge: Cambridge University Press.
- Thiébaux, H. J. and F. W. Zwiers (1984). The interpretation and estimation of effective sample size. *Journal of Climate and Applied Meteorology* 23(5), 800–811.
- Tuma, N. B. and M. T. Hannan (1984). *Social dynamics: Models and methods*. Orlando: Academic Press.
- Valente, T. W. (1995). *Network models of the diffusion of innovations*. New York: Hampton Press.

- Valente, T. W. (1996). Social network thresholds in the diffusion of innovations. *Social networks* 18(1), 69–89.
- Valente, T. W. (2005). Network models and methods for studying the diffusion of innovations. In P. J. Carrington, J. Scott, and S. Wasserman (Eds.), *Models and Methods in Social Network Analysis*, pp. 98–116. Cambridge: Cambridge University Press.
- Valente, T. W., P. Paredes, and P. R. Poppe (1998). Matching the message to the process the relative ordering of knowledge, attitudes, and practices in behavior change research. *Human Communication Research* 24(3), 366–385.
- van de Bunt, G. G., M. A. J. van Duijn, and T. A. B. Snijders (1999). Friendship networks through time: an actor-oriented dynamic statistical network model. *Computational and Mathematical Organization Theory* 5(2), 167–192.
- Veenstra, R. and C. Steglich (2012). Actor-based model for network and behavior dynamics: A tool to examine selection and influence processes. In B. Laursen, T. D. Little, and N. A. Card (Eds.), *Handbook of developmental research methods*, pp. 598–618. New York: Guildford Press.
- Wasserman, S. and K. Faust (1994). *Social Network Analysis: Methods and Applications*. Structural Analysis in the Social Sciences. Cambridge University Press.
- West, P. and H. Sweeting (1996). Nae job, nae future: young people and

health in a context of unemployment. *Health & Social Care in the Community* 4(1), 50–62.

Wilks, S. S. (1938). The large-sample distribution of the likelihood ratio for testing composite hypotheses. *The Annals of Mathematical Statistics* 9(1), 60–62.