

Anomaly Detection in Dynamic Networks

A thesis presented for the degree of
Doctor of Philosophy of Imperial College London
and the
Diploma of Imperial College by

Melissa Turcotte

Department of Mathematics
Imperial College
180 Queen's Gate, London, SW7 2AZ

FEBRUARY 2014

I certify that this thesis, and the research to which it refers, are the product of my own work, and that any ideas or quotations from the work of other people, published or otherwise, are fully acknowledged in accordance with the standard referencing practices of the discipline.

Copyright

The copyright of this thesis rests with the author and is made available under a Creative Commons Attribution Non-Commercial No Derivatives licence. Researchers are free to copy, distribute or transmit the thesis on the condition that they attribute it, that they do not use it for commercial purposes and that they do not alter, transform or build upon it. For any reuse or redistribution, researchers must make clear to others the licence terms of this work.

Acknowledgements

First I would like to thank my supervisor, Nick Heard, for his encouragement and immense support throughout my PhD. From the start, Nick has made research exciting and engaging and I consider myself extremely fortunate to have been supervised by him. Throughout my PhD Nick went out of his way to create fantastic opportunities for furthering my career and for this I am extremely grateful. He also deserves extra credit for his infinite patience in dealing with me, I imagine it was difficult at times.

I would also like to acknowledge all the people whom I worked with and got to know at ACS-PO at Los Alamos National Laboratory, without their expertise and support the chapter on computer network security would not have been possible. In particular, I would like to thank Joshua Neil, who supervised me during my time at Los Alamos and introduced me to the challenges of cyber-security. His genuine enthusiasm for this particular problem makes him a pleasure to work with.

I would like to recognise the support from EPSRC and the Institute for Security Science and Technology at Imperial College London for funding this research.

Finally, Niall Adams, Patrick Rubin-Delanchy and Ed Cohen deserve a mention for the many stimulating discussions throughout my time at Imperial and generally creating a very pleasant work environment.

Melissa Turcotte

Abstract

Anomaly detection in dynamic communication networks has many important security applications. These networks can be extremely large and so detecting any changes in their structure can be computationally challenging; hence, computationally fast, parallelisable methods for monitoring the network are paramount. For this reason the methods presented here use independent node and edge based models to detect locally anomalous substructures within communication networks. As a first stage, the aim is to detect changes in the data streams arising from node or edge communications. Throughout the thesis simple, conjugate Bayesian models for counting processes are used to model these data streams. A second stage of analysis can then be performed on a much reduced subset of the network comprising nodes and edges which have been identified as potentially anomalous in the first stage.

The first method assumes communications in a network arise from an inhomogeneous Poisson process with piecewise constant intensity. Anomaly detection is then treated as a changepoint problem on the intensities. The changepoint model is extended to incorporate seasonal behaviour inherent in communication networks. This seasonal behaviour is also viewed as a changepoint problem acting on a piecewise constant Poisson process. In a static time frame, inference is made on this extended model via a Gibbs sampling strategy. In a sequential time frame, where the data arrive as a stream, a novel, fast Sequential Monte Carlo (SMC) algorithm is introduced to sample from the sequence of posterior distributions of the changepoints over time.

A second method is considered for monitoring communications in a large scale computer network. The usage patterns in these types of networks are very bursty in nature and don't fit a

Poisson process model. For tractable inference, discrete time models are considered, where the data are aggregated into discrete time periods and probability models are fitted to the communication counts. In a sequential analysis, anomalous behaviour is then identified from outlying behaviour with respect to the fitted predictive probability models. Seasonality is again incorporated into the model and is treated as a changepoint model on the transition probabilities of a discrete time Markov process. Second stage analytics are then developed which combine anomalous edges to identify anomalous substructures in the network.

Table of contents

Abstract	5
1 Introduction	12
1.1 Anomaly detection approach	14
1.2 Bayesian changepoint modelling	16
1.3 Data sets	19
1.3.1 VAST data	19
1.4 Outline of thesis	21
2 Anomaly detection for Poisson processes	23
2.1 Inhomogeneous Poisson process changepoint model without seasonality	24
2.2 Modelling seasonality	25
2.2.1 Shared seasonality in a network	28
2.3 Anomaly detection	30
3 Fixed time inference	33
3.1 Gibbs sampling	34
3.1.1 VAST data	35
3.2 Approximate inference	41
3.2.1 VAST data	42
3.3 Sample size allocation for rival samplers	43
3.3.1 Theory	43
3.3.2 VAST data	47
3.3.3 Discussion	50
4 Real time inference	51
4.1 Sequential Monte Carlo	51
4.1.1 Importance sampling	52
4.1.2 Sequential importance sampling	54
4.2 Sequential Monte Carlo algorithm for changepoint detection	55
4.2.1 Normalising constant estimates	58
4.2.2 Choices of t_{n-1}^*	59
4.3 Adaptive sequential Monte Carlo	61
4.3.1 Increasing the weighted particle set	62
4.3.2 Algorithm	64
4.4 Examples	64

4.4.1	Simple example	66
4.4.2	Multiple simulated data sets	68
4.4.3	Coal data	72
4.4.4	VAST data	75
5	Computer network anomaly detection	80
5.1	LANL data	83
5.2	Markov chain changepoint model for seasonal node behaviour	86
5.2.1	Seasonal changepoints	86
5.2.2	Application to LANL data	89
5.3	Edge model	90
5.3.1	Markov chain for activity status	91
5.3.2	Negative Binomial distribution for counts	92
5.4	Monitoring	95
5.4.1	Predictive distributions	96
5.4.2	Anomaly graph	97
5.5	Results	100
6	Conclusion and Future Work	106
6.1	Anomaly detection for telecommunication networks	106
6.2	Computer network anomaly detection	109
	References	117
A	Calculating conditional changepoint distributions	118
A.1	Individual changepoints	118
A.2	Seasonal changepoints	119
B	Sequential Monte Carlo for non-conjugate changepoint models	121

List of Figures

1.1	VAST data event times organised by individual	20
2.1	$h_\delta(t)$ given a changepoint at 0.3 with $\nu = 1$	31
3.1	Histogram of call start times for the VAST data set	36
3.2	Posterior changepoint distributions	37
3.3	Number of anomalous individuals in the VAST data set	38
3.4	Spectral cluster plot of the anomalous individuals in the VAST data set	40
3.5	Number of anomalous individuals in the VAST data set using the approximate model	42
3.6	Box plot of the allocated sample sizes and the Jensen-Shannon divergence	49
4.1	Left: ESS, BF and the online estimate of the expected number of changepoints. Right: Online estimate of $\mathbb{P}_\pi(h_{\min(t,2)}(t) < 0)$	67
4.2	Online estimated intensity function for the coal-mining disaster data	74
4.3	Effective sample size	74
4.4	Posterior changepoint distributions	75
4.5	Number of anomalous individuals in the VAST data	77
4.6	Online estimate of $\mathbb{P}_\pi(h_t(t) < 0)$ for the individual with ID 392.	78
4.7	Box plot of the sample sizes allocated to each individual on each update interval	79
4.8	Allocated sample sizes for each update interval	79
5.1	An example of a traversal attack	83
5.2	Number of active nodes and edges in the LANL data	84
5.3	Activity status of an edge in the LANL computer network	85
5.4	Maximum count versus the median count for the observed counts in the training period for a random sample of 3,000 edges.	85
5.5	Activity status of a node in the LANL computer network	90
5.6	The empirical count distribution with the corresponding fitted Negative Binomial and Poisson distributions	94
5.7	Empirical distribution of the autocorrelation function lag 1.	95
5.8	Anomaly scores for components of the anomaly graphs	102
5.9	Heat map of the first detected anomalous component	103
5.10	Heat map of component containing anomalous hosts	104
5.11	Indegree of the two core servers in the LANL computer network	105
6.1	Empirical count distributions for edges in the LANL network	110

List of Tables

4.1	Results from the SMC and SMCMC algorithms for the 100 simulations. For the first three rival methods the summary statistics are the averages across all the runs, and standard deviations are given in parentheses.	71
-----	---	----

List of Algorithms

1	Gibbs Sampler	35
2	Approximate Gibbs sampler	41
3	Sample allocation for rival samplers	48
4	Increasing particle set from N to M	63
5	SMC algorithm for changepoint detection	65

Chapter 1

Introduction

Networks are representational forms for systems in nature, society and technology. Kolaczyk (2000) gives a good overview of networks and example contexts in which they arise. Dynamic networks are generally regarded as networks that evolve over time. Networks are often represented by graphs with a finite set of nodes or vertices and edges defining the pairwise relations between them. Relations may not continuously exist between two nodes and often edges represent sequences of instantaneous communications. Hence the information about a dynamic network is stored in a time series of graphs $\{G_t\}$, where G_t consists of a set of nodes and edges that exist at time t . In practice the node set could also be changing over time but for the purposes of this thesis it is assumed that there is a fixed number of entities in the network and the edges between them evolve over time.

Anomaly detection in dynamic networks is the continuous monitoring of these time evolving graphs for unusual events or trends. The precise definition of an anomaly depends on the application of interest but could be characterised by sudden changes in connectivity between nodes in a network, new ties forming between seemingly unrelated nodes or general irregular changes within the substructure of the graph. Generally anomaly detection systems aim to build a model based on normal graph characteristics that are observed from the network over time and then search for outlying behaviour with respect to the model, indicating the presence of unusual or possibly malicious behaviour in the network. The survey by Chandola et al. (2009) provides

a comprehensive overview of various anomaly detection techniques and their applications. Another large area of research is signature-based anomaly detection where signatures based on previous attacks are used to search for current attack behaviour, (Axelsson, 2000; Cahill et al., 2004). Non signature-based anomaly detection differs in that any type of deviations from the normal state of the network are sought and has a clear advantage in that new types of threats can be identified.

The application for the work in this thesis is anomaly detection in communication networks; in particular, in the context of monitoring for security purposes. There are many security contexts in which monitoring a large dynamic network for suspicious or fraudulent behaviour is important and two motivating network examples are considered: telecommunication networks and computer networks. Monitoring telecommunication networks for fraudulent behaviour is a well established problem and another context that has important significance to government agencies, is the online surveillance of these networks to uncover terrorist cells. Computer networks are of rapidly growing importance. With advances in networking technology and the reliance on these technologies within government and commercial enterprises, the threat from cyber-attacks has drastically increased and keeping computer networks secure is now critically important.

For most of the thesis anomaly detection is viewed as a changepoint problem and novel algorithms are developed for performing changepoint analysis on a collection of stochastic processes in both the static and continuous time frames. The main contribution of the thesis is a novel sequential Monte Carlo (SMC) algorithm for sequential changepoint detection that is computationally fast and obtains the most efficiency when applied across a large number of data streams.

More generally, a dynamic sample size allocation strategy is presented for making optimal use of a fixed computational resource when analysing multiple data streams. Sampling effort is allocated sequentially to the posterior distributions relating to the data streams according to the estimated relative complexities of those distributions. This problem has not previously been considered in the literature.

Finally, seasonality in behaviour, which is inherent in user driven communication networks, is also treated as a changepoint problem and this is also novel with respect to the current literature. The final chapter of the thesis is concerned with using statistical models for anomaly detection in computer networks. Statistical modelling for cyber security is a relatively new area of research in statistics, and an anomaly detection system is developed for a specific application of cyber security: detecting an intruder in a internal computer network (Neil et al., 2013a).

1.1 Anomaly detection approach

Communication networks such as telecommunication or computer networks are typically very large with thousands or millions of edges at any one time. To enable deployment on large graphs any anomaly detection schemes have to be computationally fast, scalable and ideally parallelisable.

One method of anomaly detection in networks is concerned with finding anomalous sub-structures of the network. Spectral graph techniques are commonly used to find unusual sub-graphs such as small cliques or nodes with unusual neighbourhoods (von Luxburg, 2007; Skillicorn, 2007; Idé and Kashima, 2004). Scan statistics have also been used in detecting local anomalies in networks (Priebe et al., 2005), where a local statistic for some graph invariant is calculated over sliding windows. The maximum across the graph of the locality statistic is defined as the scan statistic and compared against some null hypothesis of normal behaviour. In Park et al. (2013) this is extended so that a fusion of graph invariants are considered. For social networks analysis, methods have been developed that infer information about links and nodes over time using information from an existing set of attributes for the network graph such as node labels (Cortes et al., 2003; Rhodes and Keefe, 2007). All of these approaches either place an implicit specification of the type of anomaly sought or have a high computational cost when applying them to larger networks. For a more general overview of various techniques related to anomaly detection see Chandola et al. (2009)

The approach taken in this thesis is to treat each node or edge in the network graph as

independent or conditionally independent entities that have an associated counting process describing the communications process of that node or edge. For telecommunication networks this process could represent the timings of phone calls, or in computer networks the connection times between pairs of IP addresses. Heard et al. (2010) suggests that if a network has fundamentally changed in some significant way then in most contexts this indicates that some entities are communicating more or less frequently or there are new links between entities. A two-stage approach, as presented in Heard et al. (2010), will be used for the work in this thesis. First, potentially anomalous nodes are identified by deviations from their usual connectivity by treating either the pairwise communications or the directed or undirected communications for single entities as independent processes. Initially taking this broad view allows a fast sweep of the whole network to identify good anomalous targets to zoom in on. The second stage is then to construct a subgraph around these anomalous nodes, this will give a much reduced sub-network for which standard graph analysis tools can be more feasibly deployed to examine the more subtle network structure.

In this thesis, anomaly detection in networks is viewed from both continuous and discrete time perspectives. Simple conjugate Bayesian models are used for both perspectives allowing for fast tractable inference, which is imperative for the large networks of interest.

For the first motivating problem, which is anomaly detection in telecommunication networks, the event times emanating from each edge or node in the graph are treated as continuous time counting processes that are monitored for changes over time. Frost and Melamed (1994) gives an overview of commonly used models for telecommunications traffic. Inhomogeneous Poisson processes with piecewise constant intensities, where the jumps in the intensity correspond to changepoints or anomalies, provide a flexible, computationally tractable framework.

The main focus of this thesis is developing algorithms for changepoint detection on the collection of stochastic processes emanating from each node or edge. In particular a new sequential Monte Carlo algorithm is presented for fast sequential changepoint detection. Although the algorithm is designed primarily for online changepoint detection for continuous time processes, it can also be used for discrete time processes.

For the second motivating example, detecting an intruder on a computer network, discrete time models are used where the time domain is aggregated into intervals and statistical summaries of activity are collected in each interval, such as the number of connections made along an edge, to form a discrete time stochastic process. For computer network data the continuous time models used for the telecommunication networks would not be appropriate for the “bursty” behaviour that is observed.

A probability model is sequentially fitted to the discrete time process using the data observed so far. Independent hierarchical models are used to model each node’s activity status and then conditionally on that node’s status, edges are modelled conditionally independent of one another. Anomalies are then defined to be observations with low predictive probability according to the fitted model. This is an extension of the work in Heard et al. (2010), which was applied to a much simpler telecommunication network problem.

Seasonality inherent in these user driven networks adds to the challenge of monitoring these processes. For both the continuous and discrete time perspectives seasonality is also viewed as a changepoint problem. In continuous time the changepoint model for the intensity of an inhomogeneous Poisson process is extended to include seasonal changepoints and for the discrete time models the seasonal changepoints act on the transition probabilities of a discrete time Markov chain modelling the on/off activity status of nodes in the network.

As changepoint models underpin most of the methodology presented in this thesis, the following section gives a brief introduction to the Bayesian changepoint model.

1.2 Bayesian changepoint modelling

Changepoint models are widely used for many data analysis problems where the data generating process being observed undergoes characteristic changes over time. A possibly unknown number of changepoints split up the data into disjoint segments, where the data are assumed to arise from a single generative model within each segment but different models across segments. Changepoint detection is concerned with finding the location of these changes, and in a

sequential setting, detecting the change soon after it has occurred.

Changepoint models can broadly be split up into two categories:

1. The data across segments follow unrelated probability distributions; in a non-parametric setting these would be considered unknown, or
2. The data follow a probability distribution of the same functional form but the underlying parameters of the model change between segments.

This thesis and all the references to changepoint models within are concerned with inference on data that is assumed to fall under the latter category.

Models that have been considered in the literature include piecewise constant intensity Poisson processes (Green, 1995; Carlin et al., 1992; Fearnhead, 2006; Hawkins, 2001; Del Moral et al., 2006); changing linear regression (Carlin et al., 1992; Stephens, 1994; Barry and Hartigan, 1993; Fearnhead, 2006; Hinkley, 1970; Breiman et al., 1984; Hawkins, 2001) and Markov models with time-varying transition matrices (Carlin et al., 1992). These models are used in many applications such as finance, engineering and quality control.

Let $\{y(t) : 0 \leq t \leq T\}$ be a stochastic process on $[0, T]$, which will be referred to as the data. For notational brevity, the values taken by the process over any subset $B \subseteq [0, T]$ will be denoted as $y(B)$; so $y(B) = \{(t, y(t)) : t \in B\}$.

The changepoint model assumes that an unknown number of changepoints split up the data into disjoint segments where the data in each of the segments are typically independent of one another. Here it is further assumed that the distribution of $y(t)$ varies between segments only through changes in a generic parameter vector $\theta \in \Theta$.

Let k be the number of changepoints over $[0, T]$. If $k = 0$ let $\tau_{1:k} = \emptyset$, else if $k > 0$ let $\tau_{1:k} = (\tau_1, \dots, \tau_k)$ represent the ordered locations of these changepoints, so that $\tau_{1:k} \in \mathbb{T}_k$ where

$$\mathbb{T}_k = \{\tau_{1:k} : 0 < \tau_1 < \dots < \tau_k < T\}.$$

The changepoints, $\tau_{1:k}$ split up the data into $k + 1$ independent segments. For simple notation let $\tau_0 = 0$ and $\tau_{k+1} = T$. Let $\theta_{0:k} = (\theta_0, \dots, \theta_k)$ be the corresponding parameter vectors for the

observed process data in each segment $y((\tau_i, \tau_{i+1}])$ for $i = 0, \dots, k$.

Given the vectors of changepoints and parameters, the likelihood of the data over the interval $[0, T]$ can then be expressed as

$$\mathcal{L}(y([0, T]) | \tau_{1:k}, k, \theta_{0:k}) = \prod_{i=0}^k \mathcal{L}(y((\tau_i, \tau_{i+1}]) | \theta_i),$$

where $\mathcal{L}(\cdot | \theta_i)$ is a generic likelihood function for the data generating process given fixed parameters θ_i .

Given a prior distribution for the parameters and the changepoints, denoted $p(\tau_{1:k}, k, \theta_{0:k})$, the posterior distribution of interest over the interval $[0, T]$ given the observed process data is

$$\pi(\tau_{1:k}, k, \theta_{0:k} | y([0, T])) = \frac{\gamma(\tau_{1:k}, k, \theta_{0:k}, y([0, T]))}{Z}, \quad (1.1)$$

where

$$\gamma(\tau_{1:k}, k, \theta_{0:k}, y([0, T])) = \mathcal{L}(y([0, T]) | \tau_{1:k}, k, \theta_{0:k}) p(\tau_{1:k}, k, \theta_{0:k}) \quad (1.2)$$

is assumed to be known pointwise and Z , which is the corresponding normalising constant of π , may not be known.

The prior $p(\tau_{1:k}, k, \theta_{0:k})$ and the posterior $\pi(\tau_{1:k}, k, \theta_{0:k} | y([0, T]))$ both have support on the disjoint union

$$E_{\Theta} = \bigcup_{k=0}^{\infty} \{k\} \times \Theta^{k+1} \times \mathbb{T}_k.$$

The prior on the parameters and the changepoints is typically constructed as

$$p(\tau_{1:k}, k, \theta_{0:k}) = p(\tau_{1:k}, k) p(\theta_{0:k} | \tau_{1:k}, k),$$

where under the assumption of independence of the parameters between segments,

$$p(\theta_{0:k} | \tau_{1:k}, k) = \prod_{i=0}^k p(\theta_i).$$

If the model parameters $\theta_{0:k}$ can be integrated out from (1.2), as is the case when $p(\theta_i)$ is chosen to be the conjugate prior for the likelihood function for the data, then the marginal likelihood function of the data given the changepoints

$$\mathcal{L}(y([0, T]) | \tau_{1:k}, k) = \prod_{i=0}^k \int \mathcal{L}(y((\tau_i, \tau_{i+1}]) | \theta_i) p(\theta_i) d\theta_i \quad (1.3)$$

can be obtained. The posterior distribution of the changepoints given the data is then

$$\pi(\tau_{1:k}, k | y([0, T])) = \frac{\gamma(\tau_{1:k}, k, y([0, T]))}{Z}, \quad (1.4)$$

with support

$$E = \bigcup_{k=0}^{\infty} \{k\} \times \mathbb{T}_k.$$

For simplification of notation throughout the rest of the thesis, where appropriate, when referring to the posterior distribution of (1.4) the dependency on k and the data will be suppressed, so that $\gamma(\tau_{1:k}) = \gamma(\tau_{1:k}, k, y([0, T]))$ and $\pi(\tau_{1:k}) = \pi(\tau_{1:k}, k | y([0, T]))$ over some specified interval $[0, T]$.

1.3 Data sets

Two network data sets are used for demonstrations of the methods developed in this thesis. A description of the first data set, which is used in Chapters 2, 3 and 4 for the continuous time models, is given below. The second data set, which is from Los Alamos National Laboratory's large internal computer network, is described in Section 5.1 in Chapter 5.

1.3.1 VAST data

The VAST data are synthetic data taken from the mobile call mini challenge from the VAST Challenge 2008 (<http://www.cs.umd.edu/hcil/VASTchallenge2008>). The data comprise information on mobile call records for 400 unique caller IDs over a ten-day period

on a fictional island. The data include the following fields: identifier for caller, identifier for receiver, time when call was placed, call duration and cell tower from where the call originated. The challenge was aimed at social network analysis, with the aim to discover how the social structure of the network changes over the 10 day period. A possible identifier for the leader of the network was provided. The data set has sufficient realism that daily cyclical effects are present in the data as can be seen in Figure 1.1, which shows scatterplots of the call times organised by individual.

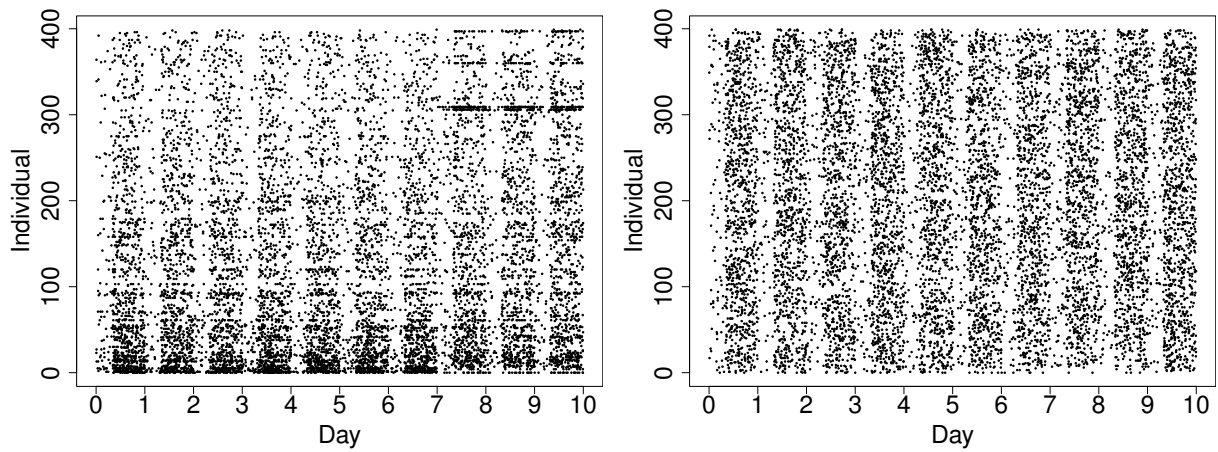


Figure 1.1: VAST data event times organised by individual. Left: Incoming call data. Right: Outgoing call data.

The results from the award winning published work on this challenge, Ye et al. (2008), suggest that the structure of the social network greatly changes on the 8th day and the change involves a list of eleven individuals including the suggested leader of the social network. Heard et al. (2010) also analysed these data and obtained comparable results through considering only the node call times. The nature of this change in the network, which can be perceived as anomalous, serves as a good example in demonstrating the methods developed throughout the thesis. In all of the analyses presented in this thesis, the call origination cell tower and call duration information is discarded.

1.4 Outline of thesis

Chapter 2 gives a description of the Bayesian changepoint model for an inhomogeneous Poisson process, which will serve as an underlying model for the communications in a network modelled in continuous time. An extension to the basic changepoint model is presented to incorporate seasonal changepoints to capture the seasonal behaviour in these networks. Finally a function of interest is presented that provides a classification for an anomaly at any time point.

Chapter 3 discusses inference for a static time frame and a Gibbs sampling algorithm is described for sampling from the joint posterior of the seasonal changepoints and the ‘real’ changepoints. This algorithm is demonstrated on the VAST Challenge network data. Full inference is computationally demanding, and so an approximate inference strategy is designed. This is also demonstrated on the VAST Challenge network data with similar results to the full inference procedure.

When sampling from the changepoint distributions for each node in the VAST network, the sample size allocated to each sampler is initially assumed to be fixed and the same for all nodes. This sample size allocation strategy doesn’t take into account the complexities of the various changepoint distributions, which could be arbitrarily different. Given a constrained total number of samples that can be obtained across all the distributions, a dynamic sample size allocation strategy from Heard and Turcotte (2013a) is implemented, which addresses how to best divide that total number of samples between the various distributions according to their relative complexities.

A novel sequential Monte Carlo (SMC) algorithm is presented in Chapter 4 to sample from the sequence of posteriors of the changepoints over time in an online setting, this is developed generically for any type of changepoint problem. The SMC algorithm is demonstrated both on simulated data, which follow an inhomogeneous Poisson process, and the VAST Challenge network data. The sample size allocation algorithm detailed in Chapter 3 is also implemented within the SMC algorithm, providing a fully adaptive sample size SMC procedure.

Finally in Chapter 5, an anomaly detection system is developed for detecting intruders on

a computer network. Hierarchical, discrete time models are proposed for the communication counts over each edge. A second stage of analysis is presented whereby edges that are identified as anomalous according to the edge-based analysis from the discrete time models are combined to form an anomaly graph. Further analysis is conducted on the anomaly graph to find anomalous substructures in the network that could indicate an intruder moving around the computer network. The methods are demonstrated on Los Alamos National Laboratory's computer network data set, which has a recorded attack present in the data.

Chapter 2

Anomaly detection for Poisson processes

Poisson processes provide the most convenient models for continuous time event data. The data considered in this chapter are the event times of phone calls either made or received in a telecommunications network. Inhomogeneous Poisson processes are commonly used to model these types of data, see Gans et al. (2003) and the references therein. There are two sources of inhomogeneity in telephone call data: there are seasonal effects such as diurnal and weekly behavioural patterns which cyclically recur, and also longer term drift in usage patterns that might occur.

Seasonality is often dealt with by splitting up the seasonal cycle into small intervals and estimating a different set of parameters for each interval, (Massey et al., 1996; Weinberg et al., 2007). This simple approach precludes the advantages of borrowing strength from neighbouring sub-intervals that are very similar.

Scott (2001) uses a Markov modulated Poisson process to detect anomalous behaviour in telecommunications traffic where under the null hypothesis of normal behaviour it is assumed the communications follow an inhomogeneous Poisson process. It is then assumed that the presence of anomalous behaviour in the data generates additional traffic according to a homogeneous Poisson process. A two state Markov process governs the presence or absence of anomalous activity, where the data generating process switches between an inhomogeneous Poisson process and a homogeneous Poisson process. Seasonality is incorporated using step

functions where there is an associated intensity multiplier for each hour and each day.

For this thesis, it is assumed that the data generating process is an inhomogeneous Poisson process where the rate undergoes multiplicative seasonal effects, which are determined by changepoint analysis. Conditional on these seasonal changepoints, a second changepoint analysis is then used to find anomalies in the data stream. Changepoint models have been used extensively in the literature for analysing inhomogeneous Poisson processes, as described in Section 1.2. The extension of using changepoints to deal with cyclical behaviour is novel; the seasonal changepoint model is more adaptive and therefore more efficient than splitting up the seasonal period into fixed intervals and fitting separate parameters for each interval.

2.1 Inhomogeneous Poisson process changepoint model without seasonality

The data generating process $y(t)$ is assumed to be the increments of an inhomogeneous Poisson process with piecewise constant intensity $\lambda(t)$. So the stochastic process $y(t) = 0$ almost everywhere, otherwise $y(t) = 1$ at finitely many t , which will be referred to as the event times of the process.

For a data process observed over the interval $[0, T]$ the jumps in the intensity will correspond to the vector of changepoints $\tau_{1:k}$. Define the parameter vector $\lambda_{0:k} = (\lambda_0, \dots, \lambda_k)$, such that $\lambda(t) = \sum_{i=0}^k \lambda_i \mathbb{I}_{(\tau_i, \tau_{i+1}]}(t)$, where $\lambda_i \in \mathbb{R}^+$ is the intensity of the process between τ_i and τ_{i+1} . Before any consideration of seasonality, the likelihood of the observed process data is

$$\mathcal{L}(y([0, T]) | \tau_{1:k}, k, \lambda_{0:k}) = \prod_{i=0}^k \lambda_i^{r_i} e^{-\lambda_i(\tau_{i+1} - \tau_i)},$$

where $r_i = \int_{\tau_i}^{\tau_{i+1}} y(t) dt$ is the number of events between τ_i and τ_{i+1} .

Over the interval $[0, T]$ the prior on the number of k changepoints is assumed to follow a

Poisson distribution with rate νT for $\nu \in \mathbb{R}^+$. Conditional on k

$$p(\tau_{1:k}|k) = \frac{k!}{T^k} \mathbb{I}_{\mathbb{T}_k}(\tau_1, \dots, \tau_k),$$

so that the prior for the changepoints is a homogeneous Poisson process,

$$p(\tau_{1:k}, k) = \nu^k e^{-\nu T} \mathbb{I}_{\mathbb{T}_k}(\tau_1, \dots, \tau_k). \quad (2.1)$$

Note an alternative prior could be chosen to induce spacings between changepoints; see Green (1995) for an example.

The $(k + 1)$ intensities will be assumed to follow the independent conjugate priors, $\lambda_i \sim \text{Gamma}(\alpha, \beta)$, so that

$$p(\lambda_{0:k}|\tau_{1:k}, k) = \prod_{i=0}^k \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda_i^{\alpha-1} e^{-\lambda_i \beta}. \quad (2.2)$$

For posterior inference the intensities $\lambda_{0:k}$ can be integrated out as in (1.3) to give the posterior distribution for the changepoints defined in (1.4), which is known only up to proportionality through

$$\gamma(\tau_{1:k}) = \nu^k e^{-\nu T} \prod_{i=0}^k \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{\Gamma(\alpha + r_i)}{(\beta + \tau_{i+1} - \tau_i)^{\alpha+r_i}}, \quad (2.3)$$

since Z does not have an analytical solution.

And it follows that, conditional on the changepoints $\tau_{1:k}$, $\{\lambda_i\}$, for $i = 0 \dots, k$, have the independent posterior distributions

$$[\lambda_i|\tau_{1:k}, k, y([0, T])] \equiv \text{Gamma}(\alpha + r_i, \beta + \tau_{i+1} - \tau_i). \quad (2.4)$$

2.2 Modelling seasonality

To incorporate the seasonal behaviour exhibited in user driven communication networks the model from the previous section needs to be extended. It will be assumed that the intensity of each communication process is subject to a multiplicative seasonal effect $\mu(t)$, so that at time

t the intensity of an individual process is $\lambda(t)\mu(t)$. For simplicity, the seasonal effect $\mu(t)$ will also be assumed to be piecewise constant over a seasonal period $(0, S]$, which repeats itself every S units of time, for $S > 0$. The seasonal period S is a length of time over which the processes can be expected to exhibit repetitive intensity patterns, such as a day.

It should be noted that the methodology presented here could be adapted to incorporate smoother or more flexible models for seasonal behaviour, such as a kernel density estimate or a Dirichlet process mixture model. However, a piecewise constant density is mathematically convenient and asymptotically would converge to the true density as more data are obtained.

Let $s_{1:\ell} = (s_1, \dots, s_\ell)$ be an ordered vector of seasonal changepoints on $(0, S]$ with seasonal multipliers $\mu_{0:\ell} = (\mu_0, \dots, \mu_\ell)$ corresponding to the periodic seasonal changes in the data, where $\mu_i \in \mathbb{R}^+$ and ℓ is unknown. Again for simplicity let $s_0 = 0$ and $s_{\ell+1} = S$ and define $\mu(t) = \sum_{j=0}^{\ell} \mu_j \mathbb{I}_{(s_j, s_{j+1}]}(t)$.

The intensity for an individual process $\lambda(t)\mu(t)$ at any time t will then be $\lambda_i\mu_j$ for some $0 \leq i \leq k$ and $0 \leq j \leq \ell$.

After time T , for a pair (i, j) , where $0 \leq i \leq k$ and $0 \leq j \leq \ell$, define the set B_{ij} to be the time points t where the process intensity $\lambda(t)\mu(t)$ is equal to $\lambda_i\mu_j$, so that

$$B_{ij} = \{t : 0 \leq t \leq T, \tau_i < t \leq \tau_{i+1}, s_j < r(t, S) \leq s_{j+1}\}$$

where $r(t, S)$ is the unique real in $(s_j, s_{j+1}]$ s.t. $\exists c \in \mathbb{N}_0$ s.t. $t = cS + r(t, S)$. The sets B_{ij} , for $0 \leq i \leq k$ and $0 \leq j \leq \ell$, will then be unions of intervals that form a partition of $[0, T]$.

If a network exhibits seasonal behaviour, the aim is to make joint inference on this extended model about both the seasonal changepoints and true changepoints, exploiting the fact that the seasonal changepoints are repeated every S units of time. When referring to posteriors the dependency on the data and ℓ will be suppressed.

The likelihood of the data for an individual process is

$$\mathcal{L}(y([0, T]) | \tau_{1:k}, \lambda_{0:k}, s_{1:\ell}, \mu_{0:\ell}) = \prod_{i=0}^k \prod_{j=0}^{\ell} (\lambda_i \mu_j)^{r_{ij}} e^{-\lambda_i \mu_j m_{ij}}, \quad (2.5)$$

where m_{ij} is the Lebesgue measure of the set B_{ij} and $r_{ij} = \int_{B_{ij}} y(t) dt$ is the number of events which fall inside B_{ij} .

The priors for the seasonal changepoints and intensities are the same as those for the changepoints in (2.1) and (2.2), so that s follows a homogeneous Poisson process on $(0, S]$ with intensity $\tilde{\nu}$ and $\mu_j \sim \text{Gamma}(\tilde{\alpha}, \tilde{\beta})$ independently for $1 \leq j \leq \ell$; for identifiability fix $\mu_0 = 1$.

The posterior distribution of interest is $\pi(\tau_{1:k}, \lambda_{0:k}, s_{1:\ell}, \mu_{0:\ell})$. Inference on this posterior will make use of the full conditional distributions:

$$\pi(\tau_{1:k} | s_{1:\ell}, \mu_{0:\ell}), \quad (2.6)$$

$$\pi_{(0,S]}(s_{1:\ell} | \tau_{1:k}, \lambda_{0:k}, \mu_0), \quad (2.7)$$

after integration over $\lambda_{0:k}$ and $\mu_{1:\ell}$ respectively. These conditional distributions are known up to proportionality respectively through the following equations (see Appendix A for calculations):

$$\gamma(\tau_{1:k}, s_{1:\ell}, \mu_{0:\ell}) = c_1 \prod_{i=0}^k \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{\Gamma(\alpha + \sum_{j=0}^\ell r_{ij})}{\left(\beta + \sum_{j=0}^\ell \mu_j m_{ij}\right)^{\alpha + \sum_{j=0}^\ell r_{ij}}} \quad (2.8)$$

$$\gamma_{(0,S]}(s_{1:\ell}, \mu_0, \tau_{1:k}, \lambda_{0:k}) = c_2 \prod_{j=1}^\ell \frac{\tilde{\beta}^{\tilde{\alpha}}}{\Gamma(\tilde{\alpha})} \frac{\Gamma(\tilde{\alpha} + \sum_{i=0}^k r_{ij})}{\left(\tilde{\beta} + \sum_{i=0}^k \lambda_i m_{ij}\right)^{\tilde{\alpha} + \sum_{i=0}^k r_{ij}}}, \quad (2.9)$$

where

$$c_1 = \nu^k e^{-\nu T} \left(\prod_{j=0}^\ell \mu_j^{\sum_{i=0}^k r_{ij}} \right),$$

$$c_2 = \tilde{\nu}^\ell e^{-\tilde{\nu} S} \left(\prod_{i=0}^k \lambda_i^{\sum_{j=0}^\ell r_{ij}} e^{-\lambda_i m_{i0}} \right).$$

It is also easy to see that $\{\lambda_i\}$ and $\{\mu_j\}$ have the independent posterior distributions

$$[\lambda_i | \tau_{1:k}, s_{1:\ell}, \mu_{0:\ell}, y([0, T])] \equiv \text{Gamma} \left(\alpha + \sum_{j=0}^\ell r_{ij}, \beta + \sum_{j=0}^\ell \mu_j m_{ij} \right) \quad (2.10)$$

and

$$[\mu_j | \tau_{1:k}, \lambda_{0:k}, s_{1:\ell}, y([0, T])] \equiv \text{Gamma} \left(\tilde{\alpha} + \sum_{i=0}^k r_{ij}, \tilde{\beta} + \sum_{i=0}^k \lambda_i m_{ij} \right) \quad (2.11)$$

for $0 \leq i \leq k$ and $1 \leq j \leq l$.

2.2.1 Shared seasonality in a network

Consider a network with L entities where the seasonal changepoints $s_{1:\ell}$ and intensity multipliers $\mu_{0:\ell}$ are shared across all individuals in the network. Joint seasonal inference is then required across all individuals in the network. Let $\tau_{1:k}^z$ be the vector of changepoints and $\lambda_{0:k}^z$ the intensities for the directed communications for individual z , for $z \in \{1, \dots, L\}$. The posterior distribution of changepoints for individual z is then:

$$\pi(\tau_{1:k}^z | s_{1:\ell}, \mu_{0:\ell}, y^z([0, T])),$$

where $y^z([0, T])$ is the data process observed for individual z .

Alternatively, changepoint modelling might be applied to the edges of the network for monitoring pairwise connections. For ease of notation the exposition will assume modelling of individuals but can very easily be extended to the pairwise case. Note that this modelling approach would not apply for monitoring the undirected communications for each entity within the network, as the data would contain duplicated event times across pairs of individuals.

The full conditional distribution for shared seasonal changepoints is simply

$$\pi_{(0, S]}(s_{1:\ell} | \tau^{1:L}, \lambda^{1:L}, \mu_0, y^{1:L}([0, T])), \quad (2.12)$$

where $\tau^{1:L} = (\tau_{1:k}^1, \dots, \tau_{1:k}^L)$, $\lambda^{1:L} = (\lambda_{0:k}^1, \dots, \lambda_{0:k}^L)$ and

$$y^{1:L}([0, T]) = (y^1([0, T]), \dots, y^L([0, T])).$$

Equations (2.9) and (2.11) concerning the seasonal process then become

$$\gamma_{(0,S]}(\tau^{1:L}, \lambda^{1:L}, s_{1:\ell}, \mu_0, y^{1:L}([0, T])) = c_2 \prod_{j=1}^{\ell} \frac{\tilde{\beta}^{\tilde{\alpha}}}{\Gamma(\tilde{\alpha})} \frac{\Gamma(\tilde{\alpha} + \sum_{z=1}^L \sum_{i=0}^{k^z} r_{ij}^z)}{(\tilde{\beta} + \sum_{z=1}^L \sum_{i=0}^{k^z} \lambda_i^z m_{ij}^z)^{\sum_{z=1}^L \sum_{i=0}^{k^z} r_{ij}^z + \tilde{\alpha}}}, \quad (2.13)$$

where

$$c_2 = \tilde{\nu}^{\ell} e^{-\tilde{\nu}S} \left(\prod_{z=1}^L \prod_{i=0}^{k^z} \lambda_i^z \sum_{j=0}^{\ell} r_{ij}^z e^{-\lambda_i^z m_{i0}^z} \right)$$

and

$$[\mu_j | \tau^{1:L}, \lambda^{1:L}, s_{1:\ell}, y^{1:L}([0, T])] \equiv \text{Gamma} \left(\tilde{\alpha} + \sum_{z=1}^L \sum_{i=0}^{k^z} r_{ij}^z, \tilde{\beta} + \sum_{z=1}^L \sum_{i=0}^{k^z} \lambda_i^z m_{ij}^z \right). \quad (2.14)$$

See Appendix A for details on calculations for (2.13).

An approximation to the full joint inference procedure could also be considered, where the seasonal changepoints are regarded as independent of the changepoints and intensities of the individuals in the network. The model for the seasonality is then simply the one specified in Section 2.1. The posterior distribution $\pi_{(0,S]}(s_{1:\ell} | y^{1:L}([0, T]))$ is known up to proportionality through

$$\gamma_{(0,S]}(s_{1:\ell}, y^{1:L}([0, T])) = \nu^{\ell} e^{-\ell S} \prod_{i=0}^{\ell} \frac{\beta^{\alpha}}{\Gamma(\alpha)} \frac{\Gamma(\alpha + r_i)}{(\beta + n_S L(s_{i+1} - s_i))^{\alpha + r_i}}, \quad (2.15)$$

where n_S is the number of seasons in the interval $[0, T]$ and r_i is the number of events observed in the seasonal segment $(s_i, s_{i+1}]$ across all individuals.

Assuming shared seasonal behaviours across all entities in a graph is appealing as this allows borrowing of strength across individuals for estimating seasonality. However, if this assumption is inappropriate for an application, then all individuals or pairs of individuals in the network could simply have their own sets of seasonal changepoints and intensities. Inference then reduces to sampling from the posteriors specified in (2.6) and (2.7) for each individual in the network.

2.3 Anomaly detection

Given a vector of changepoints $\tau_{1:k}$ in $[0, T]$, a function of interest is required to measure the level of anomalous behaviour at any time $t \in [0, T]$. In a changepoint model, a process could be considered anomalous in the immediate period following a changepoint. Suppose at time t , there will only be interest in changepoints that have occurred in the recent interval $[t - \delta(t), t]$ for some function $0 < \delta(t) \leq t$.

As a measure of an anomaly at time t a natural function of interest $g(t)$ to look at would then be the minimum of the distance to the nearest changepoint to the left and $\delta(t)$:

$$g_\delta(t) = \min(t - \tau_{i^*}, \delta(t)) \quad (2.16)$$

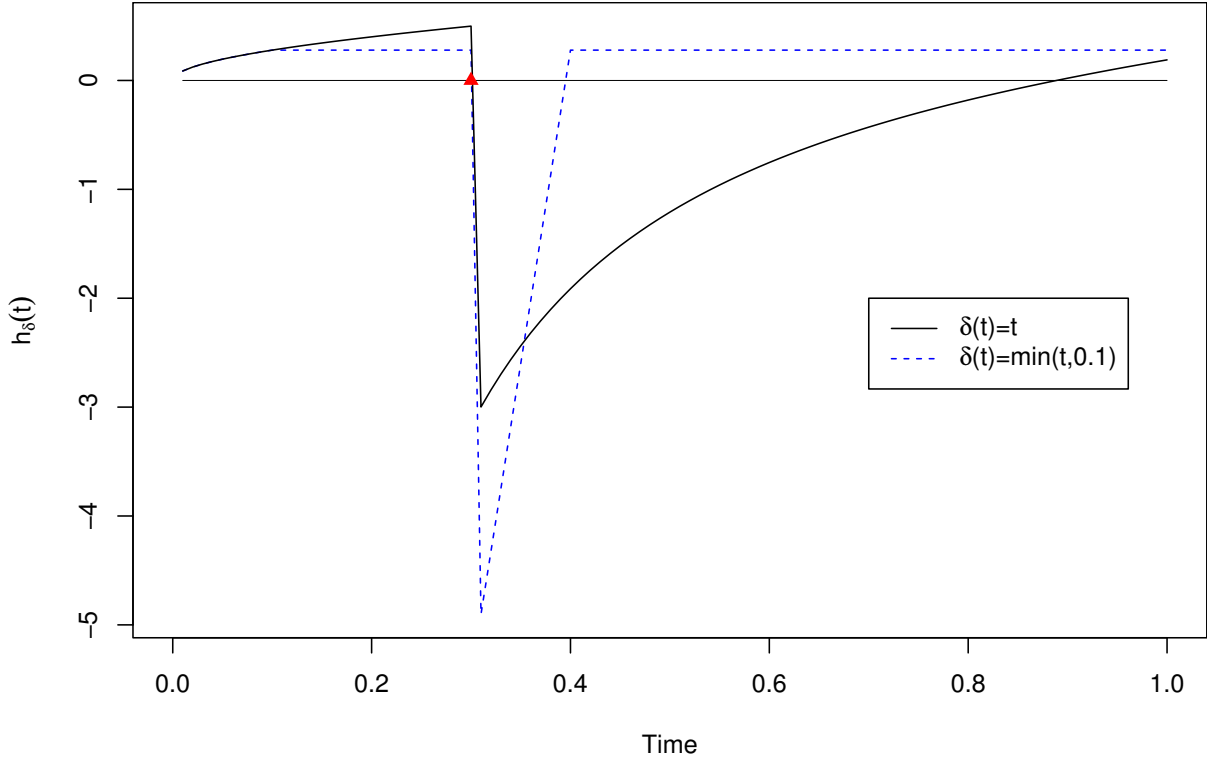
$$i^* = \max\{i \in \{0, \dots, k\} : \tau_i \leq t\}$$

for $0 < t \leq T$. A $g_\delta(t)$ near zero would represent a recent changepoint and hence signifies anomalous behaviour. $\delta(t)$ can be thought of as controlling how long the period following the changepoint is perceived as anomalous before the new behaviour is accepted. The default choice for $\delta(t)$ is $\delta(t) = t$; however $\delta(t)$ could be chosen to correspond to a fixed window size, giving $\delta(t) = \min(t, \delta)$ for some $\delta > 0$.

The prior expectation and variance of $g(t)$ can be calculated using the exponentially distributed inter-arrival time property for Poisson processes. Under the prior, $g_\delta(t)$ is the minimum of $\delta(t)$ and an $\text{Exp}(\nu)$ random variable, hence

$$\begin{aligned} \mathbb{E}[g_\delta(t)] &= \int_0^{\delta(t)} x\nu e^{-\nu x} dx + \int_{\delta(t)}^\infty \delta(t)\nu e^{-\nu x} dx = \frac{1}{\nu} (1 - e^{-\nu\delta(t)}), \\ \mathbb{E}[g_\delta(t)^2] &= \int_0^{\delta(t)} x^2\nu e^{-\nu x} dx + \int_{\delta(t)}^\infty \delta(t)^2\nu e^{-\nu x} dx = \frac{2}{\nu^2} (1 - e^{-\nu\delta(t)}(1 + \nu\delta(t))), \\ \mathbb{V}[g_\delta(t)] &= \frac{1}{\nu^2} (1 - 2\nu\delta(t)e^{-\nu\delta(t)} - e^{-2\nu\delta(t)}). \end{aligned}$$

Note that this assumes that the prior on the number and location of the changepoints is a Poisson process, which will be the case for all applications here where the function of interest is utilised.

Figure 2.1: $h_\delta(t)$ given a changepoint at 0.3 with $\nu = 1$.

As the prior expectation and variance is non-zero except at $t = 0$ and increasing when $\delta(t) = t$ or constant for $\delta(t) = \min(t, \delta)$ when $t > \delta$, standardising gives a revised function of interest, $h_\delta(t)$, which will drop below zero directly after a changepoint,

$$h_\delta(t) = \frac{g_\delta(t) - \mathbb{E}[g_\delta(t)]}{\sqrt{\mathbb{V}[g_\delta(t)]}},$$

for $0 < t \leq T$. The function $h_\delta(t)$ is clearly negative at $t = \tau_{i^*}$ and how long it remains negative will depend upon the choice of ν and $\delta(t)$. Figure 2.1 shows a simple example with one changepoint of the function $h_\delta(t)$ for two choices of $\delta(t)$. An anomaly can be classified at t if given the data,

$$\mathbb{E}_\pi[h_\delta(t)] < 0,$$

where $\mathbb{E}_\pi[h(t)]$ is the posterior expectation; or for a stronger requirement

$$\mathbb{P}_\pi(h_\delta(t) < 0) > 1 - p, \quad (2.17)$$

for some small p . For all the examples given in this thesis (2.17) will be used for detecting anomalies in the data streams and $1 - p$ will be referred to as the significance level. To calibrate the consequences of any proposed value of p , Monte Carlo simulations would be required to find the implied probability of falsely flagging an individual as anomalous at some time point in $[0, T]$ due to the multiplicity of the tests over time.

The next two chapters will be concerned with sampling from the posterior of the number and location of the changepoints. Once a sample of m , say, is obtained from the marginal distribution of $\pi(\tau_{1:k})$ the Monte Carlo estimate of the posterior probability $\mathbb{P}_\pi(h_\delta(t) < 0)$ is

$$\widehat{\mathbb{P}}_\pi(h_\delta(t) < 0) = \frac{1}{m} \sum_{i=1}^m \mathbb{I}(h_\delta^{(i)}(t) < 0), \quad (2.18)$$

where $h_\delta^{(i)}(t)$ is the function of interest evaluated at t for the i th sample.

Then for t an estimate for the location of the most recent changepoint is given by

$$\hat{\tau}_{i^*}(t) = t - \widehat{\mathbb{E}}_\pi[g_t(t)], \quad (2.19)$$

where

$$\widehat{\mathbb{E}}_\pi[g_t(t)] = \frac{1}{m} \sum_{i=1}^m g_t^{(i)}(t).$$

Chapter 3

Fixed time inference

The literature on inference for changepoint models is exhaustive for both the Bayesian and frequentist approaches. The frequentist approach is largely based on maximum likelihood estimation: Hinkley (1970) derives the asymptotic distribution of the maximum likelihood estimator for a single changepoint; Breiman et al. (1984) use the ideas of regression trees to find multiple changepoints where there is a hierarchical binary split of the data and Hawkins (2001) provides an alternative to this by performing a multiway split. All of these methods naturally identify changepoints as discrete random variables.

The focus for this work is on a fully Bayesian approach. Most of the modern literature is centred around using Markov chain Monte Carlo (MCMC) methods to sample from the posterior distribution. Since the seminal paper from Green (1995), reversible jump MCMC (RJMCMC), which is a Metropolis-Hastings algorithm used to sample from target distributions of varying dimension, is the most widely used. This method is particularly useful since it can deal with an unknown number of changepoints and changepoints can be modelled as continuous random variables. Much of the work before this modelled changepoints as discrete random variables and relied on conditioning on a fixed number of changepoints; for example, Carlin et al. (1992) use a Gibbs sampler given a hierarchical Bayesian model to find the location of a single changepoint and Stephens (1994) extends this to finding multiple changepoints in both a discrete and continuous setting.

For analytical solutions Barry and Hartigan (1992, 1993) propose a product partition model for the discrete changepoint problem where recursions are used to calculate the posterior means for the parameters of the model and Fearnhead (2006) extends this to show how it could be used for direct simulation from the posterior on the number and location of the changepoints. In a continuous setting Raftery and Akman (1986) uses numerical integration to find the location of a single changepoint. There is no known literature for finding analytical solutions for higher dimensions.

In a fixed time frame, a Gibbs sampler is presented to sample from the joint posterior of the seasonal and the individual changepoints detailed in Section 2.2 in a network with L individuals. The sampler is then demonstrated on the VAST data described in Section 1.3.1.

3.1 Gibbs sampling

As the full conditional distributions given in Equations (2.6), (2.10), (2.12) and (2.14) are known or can be sampled from easily via MCMC, a natural method to sample from the joint posterior of the seasonal and real changepoints is using a Gibbs sampler as described in Smith and Roberts (1993) and Brooks (1998). This approach is detailed in Algorithm 1.

The conditional posterior distributions of μ_j and λ_j (Lines 9 and 12) are of standard form and can be sampled from directly, whilst RJMCMC steps are introduced into the Gibbs sampler to obtain a sample of size m , say, from the conditional posterior distributions of $\tau_{1:k}$ and $s_{1:\ell}$ (Lines 8 and 11). The RJMCMC moves employed are similar to those of Green (1995), who also applies them to a Poisson process, and Denison et al. (2002) where the transitions are: (a) ‘birth’ of a new changepoint uniformly chosen over the time period; (b) ‘death’ of a randomly chosen changepoint; and (c) ‘move’ a randomly chosen changepoint uniformly over some interval around the changepoint.

To get the starting vector of seasonal changepoints, RJMCMC is used to sample from the approximate posterior for the seasonal changepoints, (2.15), where the seasonal changepoints are assumed to be independent of the individual changepoints. The starting seasonal change-

Algorithm 1 Gibbs Sampler

```

1: for  $i = 1 : m_0$  do
2:   Draw  $s_{1:\ell}^{(i)}$  from  $\pi_{(0,S]}(s_{1:\ell}|y^{1:L}([0, T]))$  (2.15)
3: end for
4: Let  $\ell^* = \underset{\ell}{\operatorname{argmaxp}}(\ell|y^{1:L}([0, T]))$  and  $s_{1:\ell^*}^* = \underset{s_{1:\ell}^{(i)}}{\operatorname{argmax}} \pi_{(0,S]}(s_{1:\ell}^{(i)}|y^{1:L}([0, T])) \mathbb{I}_{\ell^{(i)}}(\ell^*)$ 
5: Set  $(s_{1:\ell}^{(0)}, \mu_{1:\ell}^{(0)}) = (s_{1:\ell^*}^*, \mathbb{E}_{\pi(\mu|s_{1:\ell^*}^*)}(\mu))$ 
6: for  $i = 1 : m$  do
7:   for  $j = 1 : L$  do
8:     Draw  $\tau_{1:k}^{j,(i)}$  from  $\pi(\tau_{1:k}^j | s_{1:\ell}^{(i-1)}, \mu_{0:\ell}^{(i-1)}, y^j([0, T]))$  (2.6)
9:     Draw  $\lambda_{0:k}^{j,(i)}$  from  $\pi(\lambda_{0:k}^j | s_{1:\ell}^{(i-1)}, \mu_{0:\ell}^{(i-1)}, \tau_{1:k}^{j,(i)}, y^j([0, T]))$  (2.10)
10:  end for
11:  Draw  $s_{1:\ell}^{(i)}$  from  $\pi_{(0,S]}(s_{1:\ell} | \tau^{1:L,(i)}, \lambda^{1:L,(i)}, y^{1:L}([0, T]))$  (2.12)
12:  Draw  $\mu_{1:\ell}^{(i)}$  from  $\pi_{(0,S]}(\mu_{1:\ell} | \tau^{1:L,(i)}, \lambda^{1:L,(i)}, s_{1:\ell}^{(i)}, y^{1:L}([0, T]))$  (2.14)
13: end for

```

points are then taken as the maximum a posteriori (MAP) from a sample of size m_0 , say, by first obtaining the MAP number of changepoints, and then conditional on the MAP number of changepoints obtaining the MAP changepoints, Line 5 in Algorithm 1. The starting values for the seasonal intensities $\mu_{1:\ell}$ are taken as the posterior expectations (2.14) given the starting seasonal changepoints.

Note that Algorithm 1 is only partially parallelisable as only Lines 7–10 can be computed in parallel; however, if each individual in the network is assumed to have their own seasonal behaviour then the entire algorithm becomes parallelisable.

3.1.1 VAST data

To demonstrate Algorithm 1 it will be applied to the VAST data set discussed in Section 1.3.1, where the call times for each individual in the network are assumed to be distributed as a Poisson process. To check the suitability of modelling the VAST data as a Poisson process with a shared seasonal intensity across individuals, a Kolmogorov-Smirnov test (Lewis, 1965) was performed which yielded a p-value of 0.5703. The seasonal intensity used for the test was the posterior expectation given the MAP seasonal changepoints obtained from the approximate model (Line 5 in Algorithm 1), using the event times for all individuals in the network.

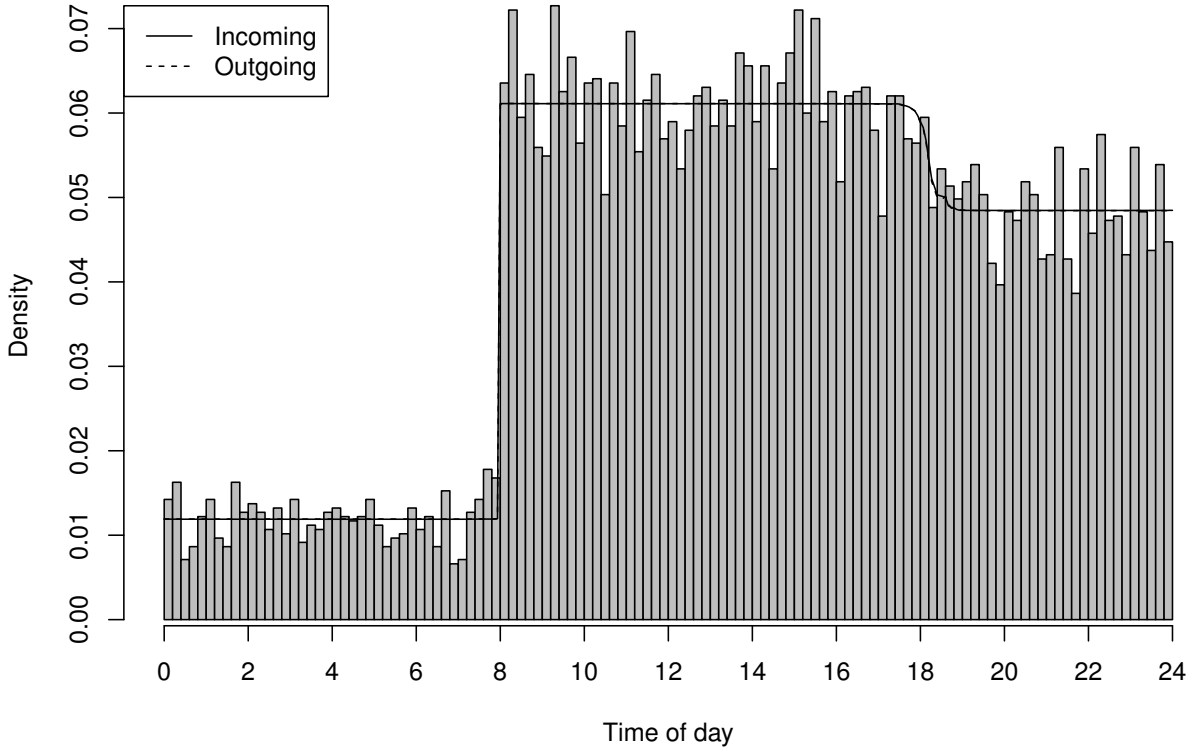


Figure 3.1: Histogram of call start times throughout the 10 day period across the whole network for the VAST data set and the posterior expectation of the seasonal intensity effects $\mu(t)$ for the outgoing and incoming calls.

Note that the Poisson process assumption is slightly unrealistic in that after a call is made or received there will be a period of time corresponding to the length of the call for which the individual is not available to receive or make any more calls. An extended model is given in Heard and Turcotte (2013b) where a two state Markov jump process is used to model both the rate at which calls are made and then once on the call the rate at which the calls are terminated. This extended model is also applied to the VAST data.

The seasonal period $(0, S]$ for the data is chosen to be one day and seasonality is assumed to be shared across all individuals in the network, this is a reasonable assumption as can be seen by Figure 1.1. Analysis is repeated for both the outgoing and incoming communications for each individual in the network.

Uninformative priors are chosen for the individual intensities and seasonal intensity multipliers, $\alpha = \beta = \tilde{\alpha} = \tilde{\beta} = 0.05$; other small values for the parameters could also be chosen to give similar inference. The prior intensity chosen for the seasonal changepoints is $\tilde{\nu} = 2$,

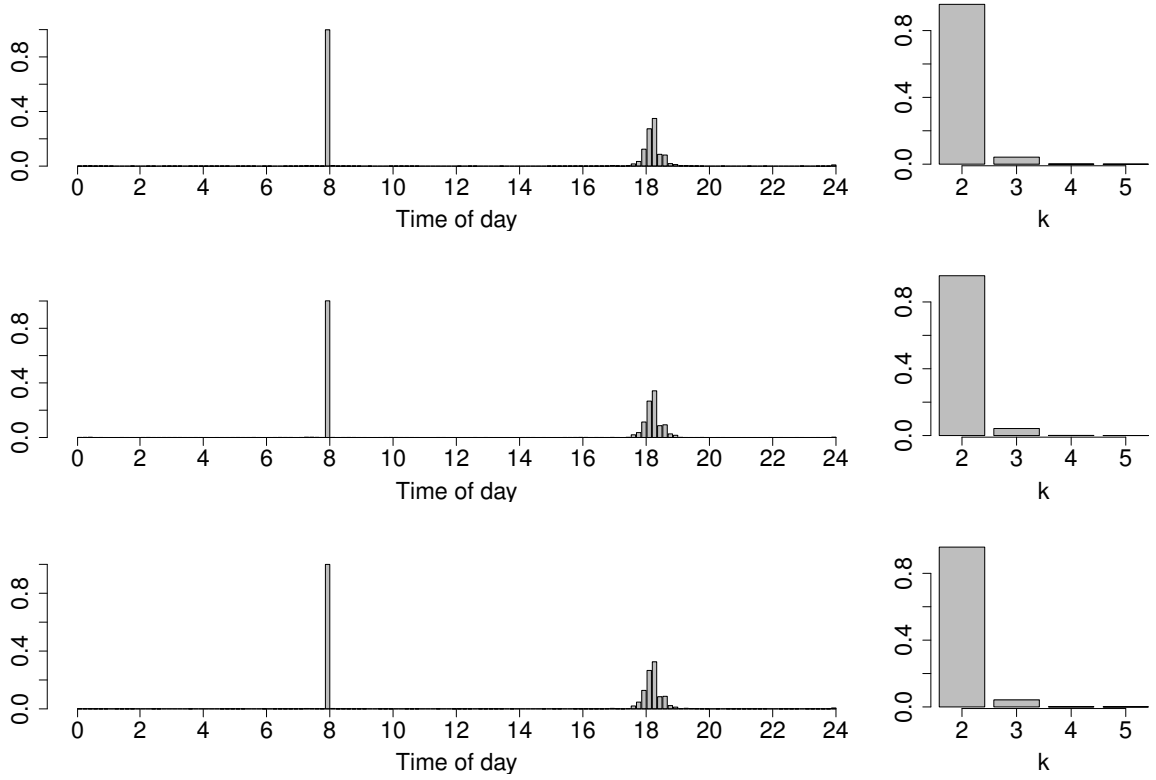


Figure 3.2: Left: Binned one-dimensional projection of the seasonal changepoint posterior where each bar shows the probability of a changepoint falling in that 4 hour bin. Right: Posterior probability of the number of changepoints. Top: Outgoing call analysis. Middle: Incoming call analysis. Bottom: Initial sample using the approximate model for the seasonal changepoints.

corresponding to a prior expectation of two changes to the rate of communications throughout the day, this can be seen by Figure 3.1, which shows a histogram of the call times of all the individuals in the network rolled over on to one day. When choosing the prior for the individual changepoints, intuitively it can be thought of as the proportion of the communication processes that are being monitored that would be expected to behave in an anomalous way, *a priori*. For this example, $\nu = 0.005$, this is approximately a prior belief that 5% of the individuals could potentially be considered anomalous over the 10 day period. A sample size of 100,000 was drawn from the conditional posteriors for the seasonal changepoints, (2.12), and the individual changepoints, (2.6), (Lines 8 and 11 in Algorithm 1), with one in every ten samples being retained and a burnin of 10,000 samples discarded for both the incoming and outgoing call analysis.

The starting sample for the seasonal changepoints (Lines 1:3 in Algorithm 1) was obtained

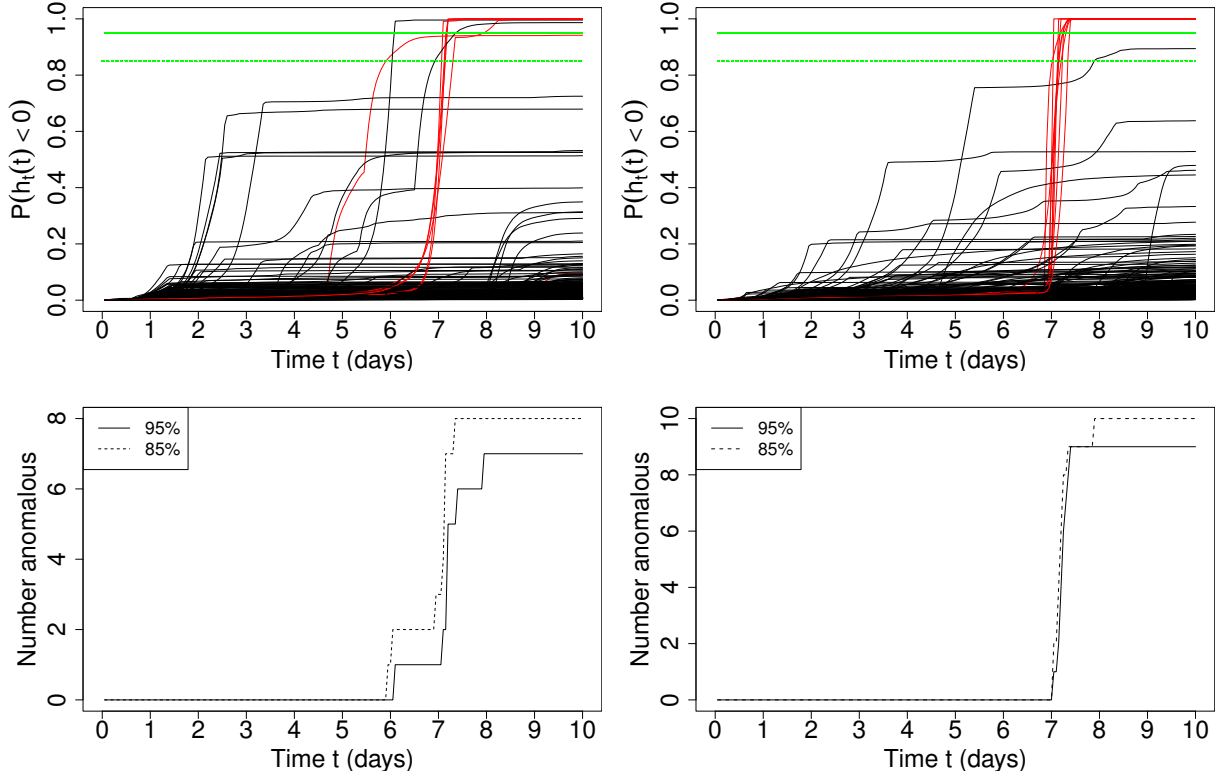


Figure 3.3: Top: $\hat{\mathbb{P}}_{\pi}(h_t(t) < 0)$, known anomalous caller ID's are in red. Bottom: Number of caller IDs considered anomalous at each time t for two different significance levels. Left: Outgoing call analysis. Right: Incoming call analysis.

from the approximate model for the seasonality using the event times for all the individuals. The posterior expectations of the seasonal intensity $\mu(t)$ for the outgoing and incoming call analyses are shown in Figure 3.1, rescaled so that $\mu(t)$ is represented as a density in each case. The corresponding histograms of the seasonal changepoints are shown in Figure 3.2 as well as the histogram for the seasonal changepoints from the initial sample obtained using the approximate model.

The posterior distributions of the seasonal changepoints, Figure 3.2, do not vary much across the incoming or outgoing call analyses, and are tightly centred around the well-chosen starting values for the seasonal changepoints, $s^{(0)} = (7.95, 18.36)$, from the approximate model for the seasonal changepoints.

Figure 3.3 shows the graph of the Monte Carlo estimate of $\mathbb{P}_{\pi}(h_t(t) < 0)$, (2.18), for both the incoming and outgoing call analyses for all 400 individuals. It also shows the number of

anomalous IDs found over time based on the criteria given in (2.17) for two different significance levels, 85% and 95%.

Clearly the most anomalous time period is at the start of the 8th day. At the 95% significance level for the outgoing call analysis seven anomalous individuals were found, which include five ID's from the analysis of Ye et al. (2008). The other two ID's could be considered as false detections. For the incoming call analysis nine anomalous individuals were identified, with five corresponding to the outgoing call analysis and all nine corresponding to Ye et al. (2008). The number of anomalous individuals found is fairly robust to the choice of the significance level at which the anomalies are flagged, with just one extra anomalous individual being flagged for the outgoing call analysis at the 85% significance level and for the incoming call analysis one extra individual flagged at the 85% significance level, which could be considered a false detection based on the analysis of Ye et al. (2008).

Figure 3.4 is a spectral cluster plot, using the first two principal components of the symmetric Laplacian of the unweighted adjacency matrix, of individuals who were in contact with the anomalous individuals identified at the 95% significance level. Analysis performed in Ye et al. (2008) using graph analysis tools identified the major anomalous activity as involving people who usually communicated with ID's 1, 2, 3 and 5 switching to communicating with ID's 309, 397, 360 and 306, the pairs appearing at the extremes of the diagram in Figure 3.4 and identified by changepoint analysis. The other individual found using both the incoming and outgoing call analyses, ID 300, was one of the three leaders of the social network who were identified in Ye et al. (2008), differing from the other two in that he only appeared after the 8th day. The other two individuals identified by Ye et al. (2008), ID's 200 and 0 are leaders in the social network and are not identified as behaving anomalously in any way, but rather communicated with all of the anomalous individuals. And as noted in Heard et al. (2010), "a simple investigation of the call activity of the set of anomalous nodes detected reveals ID 200 to be the most frequent communicator in the network with this group" and then Ye et al. (2008) states that "the person whose ID is 0 communicated with all the important people who communicated with 200". The analysis done here correctly identified one more ID on the anomalous 8th day

than Heard et al. (2010), where the total communications for each entity were monitored rather than the incoming and outgoing communications.

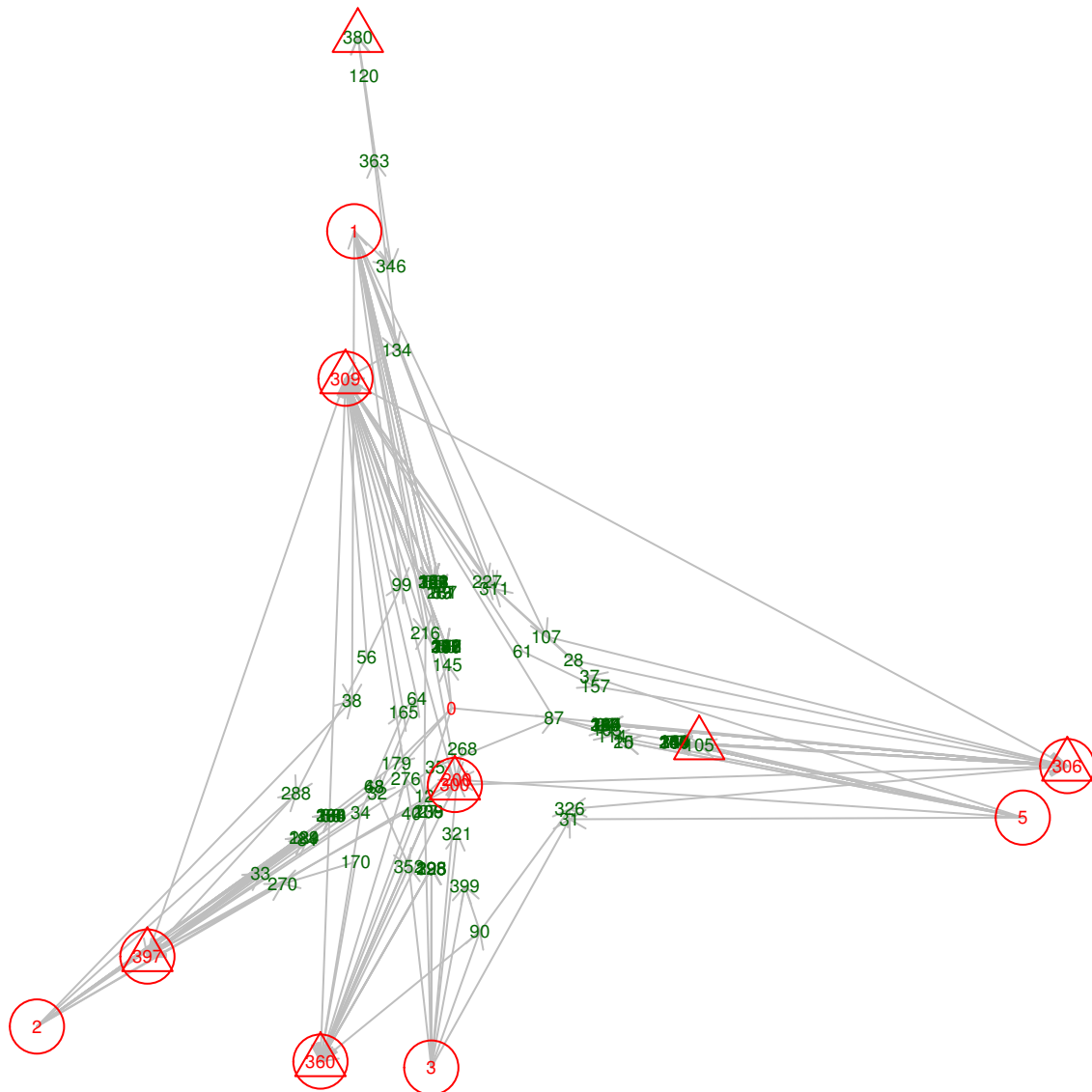


Figure 3.4: A spectral cluster plot of individuals who were in contact with the anomalous individuals identified at the 95% level from the 7th day to the 8th day. Nodes identified by Ye et al. (2008) as being involved in the anomalous activity are highlighted in red; nodes identified as anomalous here are circled in red for the incoming call data and have a red triangle around them for the outgoing call data.

Algorithm 2 Approximate Gibbs sampler

```

1: for  $i = 1 : m_0$  do
2:   Draw  $s_{1:\ell}^{(i)}$  from  $\pi_{(0,S]}(s_{1:\ell}|y^{1:L}([0, T]))$  (2.15)
3: end for
4: Let  $\ell^* = \underset{\ell}{\operatorname{argmax}}(\ell|y^{1:L}([0, T]))$  and  $s_{1:\ell^*}^* = \underset{s_{1:\ell}^{(i)}}{\operatorname{argmax}}\pi(s_{1:\ell}^{(i)}|y^{1:L}([0, T]))\mathbb{I}_{\ell^{(i)}}(\ell^*)$ 
5: Set  $(s_{1:\ell}^*, \mu_{0:\ell}^*) = (s_{1:\ell^*}^*, \mathbb{E}_{\pi(\mu|s_{1:\ell^*}^*)}(\mu))$ 
6: for  $i = 1 : m$  do
7:   for  $j = 1 : L$  do
8:     Draw  $\tau_{1:k}^{j,(i)}$  from  $\pi(\tau_{1:k}^j|s_{1:\ell}^*, \mu_{0:\ell}^*, y^j([0, T]))$  (2.6)
9:     Draw  $\lambda_{0:k}^{j,(i)}$  from  $\pi(\lambda_{0:k}^j|s_{1:\ell}^*, \mu_{0:\ell}^*, \tau_{1:k}^{j,(i)}, y^j([0, T]))$  (2.10)
10:   end for
11: end for

```

3.2 Approximate inference

The tightness of the posterior distribution for the seasonal changepoints around the starting values, and the similarity of the distributions for both the incoming and outgoing call analysis with that of the initial sample (Figure 3.2), suggests that no further information is gained by conditioning on individuals' changepoints. Using the approximate model whereby the individuals' changepoints are assumed to be independent of the seasonal changepoints provides a simplified algorithm detailed in Algorithm 2. Due to the low variance of the seasonal changepoint distribution, the MAP of a sample obtained from the approximate distribution of the seasonal changepoints given the event times across the network, (2.15), can provide an adequate summary of the seasonal effects. The full conditional distributions for each individual's changepoints then conditions on this MAP of the independent seasonal changepoint sample rather than iterating between the full conditionals for the seasonal and individuals' changepoints.

The benefits of this approach are that computation speed will be increased, inference will extend nicely in the sequential setting, the algorithm will be completely parallelisable once the seasonal changepoints are determined and such a model allows inference to be done on the total communications for entities within the network.

If $\lambda_{0:k}$ is regarded as a nuisance parameter then Line 9 in Algorithm 2 can be eliminated.

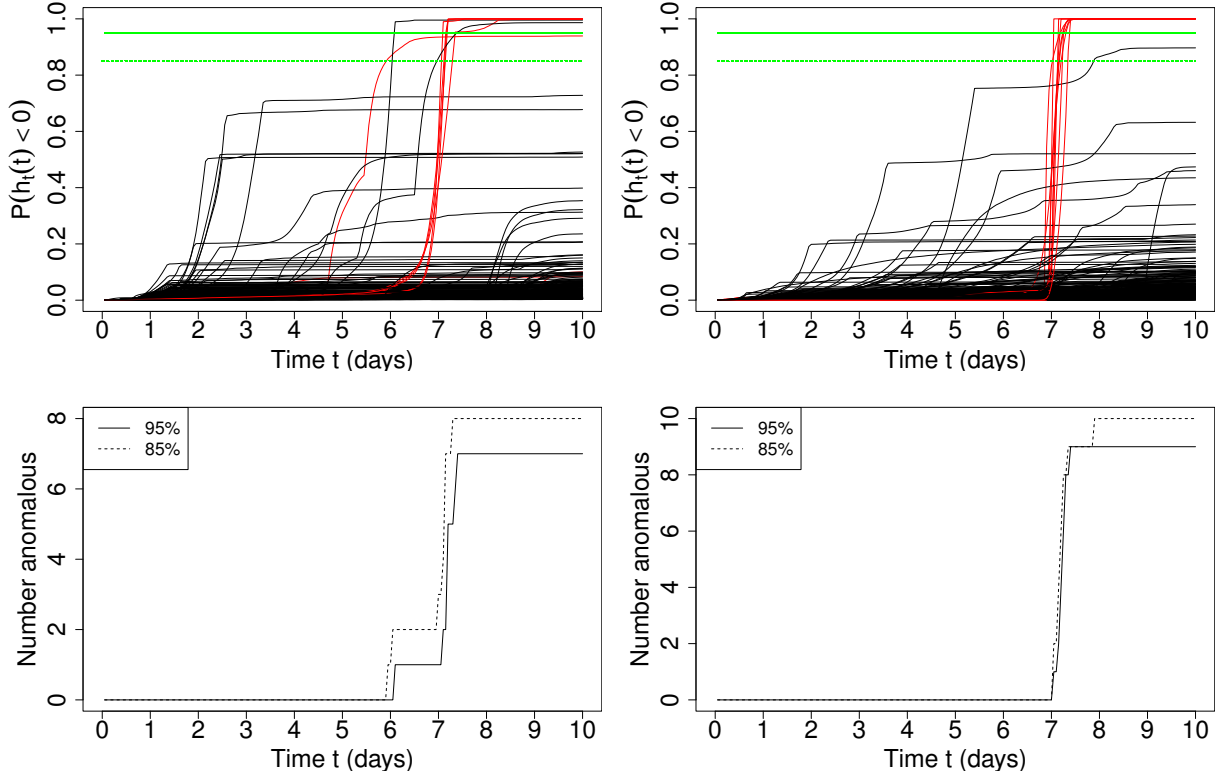


Figure 3.5: Top: $\hat{\mathbb{P}}_{\pi}(h_t(t) < 0)$, known anomalous caller ID's are in red. Bottom: Number of caller IDs considered anomalous at each time t for two different significance levels. Left: Outgoing call analysis. Right: Incoming call analysis.

3.2.1 VAST data

To make comparable inference to Algorithm 1 the same prior parameters are chosen and the same RJMCMC settings are used. The MAP for the seasonal changepoints was taken from the sample obtained in the analysis in Section 3.1.1 from the approximate seasonal model used to provide an initial starting value for the Gibbs sampler given in Algorithm 1.

The counts of anomalous individuals identified under the same criteria used in Section 3.1.1 as well as the Monte Carlo estimate of (2.18) for each individual are shown in Figure 3.5. The same individuals were identified as anomalous as those from the inference above, with one of the individuals for the outgoing call analysis now being identified slightly earlier at the 95% significance level.

As can be seen from comparing Figures 3.3 and 3.5 this method is a good approximation to the full inference performed above in Section 3.1.

3.3 Sample size allocation for rival samplers

In Algorithms 1 and 2 the sample size allocated to each RJMCMC sampler for the changepoint posteriors for each node in the network is assumed to be fixed and the same for all L nodes. Consider that fixed computational resources exist that translate to a restriction on the total number of observations, m^* , that can be sampled across all L distributions. The default choice in this case would be to split up the samples evenly between the target distributions. However, this simple strategy does not take into account the complexity of the different target distributions, which could be arbitrarily different.

Consider L target probability distributions $\pi_1, \dots, \pi_L$, from which random samples are to be drawn for the purpose of approximating the true distributions as in Algorithms 1 and 2. Given a fixed computational resource corresponding to a maximum amount of samples that can be drawn from all L distributions, Heard and Turcotte (2013a) addresses how best to divide the total number of samples between the L samplers of the different distributions based on their relative complexities. An error criterion to assess the progress of each sampler is provided with the aim of choosing which samplers should be allocated more sampling effort. The next section will outline the methodology of Heard and Turcotte (2013a); that methodology will then be applied to the VAST data, with $\pi_1, \dots, \pi_L$ corresponding to the changepoint posteriors for each of the L individuals in the network.

3.3.1 Theory

The empirical distribution estimate of a probability distribution π given m samples with support $\mathcal{X}$ is given by

$$\hat{\pi}_m(B) = \frac{1}{m} \sum_{i=1}^m \delta_{X_i}(B), \quad B \subseteq \mathcal{X}.$$

In assessing whether m samples are sufficient to approximate π across $\mathcal{X}$ it is necessary to understand how well $\hat{\pi}_m$ estimates π . To aid this assessment, a distribution-free Monte Carlo divergence diagnostic is derived in Heard and Turcotte (2013a) motivated as a limit of the Jensen-Shannon divergence (Lin, 1991) of empirical measures $\hat{\pi}_m$ from repeated sampling.

When the target distribution relates to that of a continuous random variable a discretisation of the empirical measure (and notionally the target) is performed, as without this the overlap of any two sets of independent samples would be empty. For variable-dimensional mixture distributions, such as the distribution on an unknown number of changepoints, a large but finite grid with fixed spacing is used to partition the support of $\mathcal{X}$ for each dimension, such that there is an infinite number of bins.

Monte Carlo divergence

Let $\hat{\pi}^{(1)}, \dots, \hat{\pi}^{(R)}$ be binned empirical distribution estimates of a probability distribution π obtained from $R \geq 2$ independent runs of the chosen sampler of π , where the sample size of each run is itself a realisation of a random variable M corresponding to a random stopping rule for the sampler. The Jensen-Shannon divergence (JSD) of $\hat{\pi}^{(1)}, \dots, \hat{\pi}^{(R)}$ is first proposed as an empirical measure of Monte Carlo divergence for this collection of estimates,

$$JSD(\hat{\pi}^{(1)}, \dots, \hat{\pi}^{(R)}) = H\left(\frac{1}{R} \sum_{j=1}^R \hat{\pi}^{(j)}\right) - \frac{1}{R} \sum_{j=1}^R H(\hat{\pi}^{(j)}), \quad (3.1)$$

where $H(p)$ is the Shannon entropy of the probability mass function p . Recall that if $p = (p_1, \dots, p_K)$ is a probability mass function with K support points with non-zero mass $\{p_i\}$,

$$H(p) = - \sum_{i=1}^K p_i \log(p_i).$$

The Jensen-Shannon divergence (3.1) measures the difference in the distribution estimates obtained from R sampling repetitions by calculating their average Kullback-Leibler divergence from the closest dominating measure, which is their average.

The error criterion given in Heard and Turcotte (2013a), e_{KL} , for assessing how well the empirical sample represents the target distribution based on a single run of a sampler is the limit as $R \rightarrow \infty$ of (3.1),

$$e_{\text{KL}} = H(\pi) - \mathbb{E}[H(\hat{\pi})]. \quad (3.2)$$

This can be seen using the strong law of large numbers,

$$\lim_{R \rightarrow \infty} \frac{1}{R} \sum_{j=1}^R \hat{\pi}^{(j)} = \pi,$$

whilst

$$\lim_{R \rightarrow \infty} \frac{1}{R} \sum_{j=1}^R H(\hat{\pi}^{(j)}) = \mathbb{E}[H(\hat{\pi})],$$

the expected entropy of a Monte Carlo estimate from one of the runs. Hence it follows that the Jensen-Shannon divergence given in (3.1) is a finite sample, consistent estimate of the proposed Monte Carlo divergence criterion given in (3.2). An interpretation for this divergence criterion, from the definition of the Jensen-Shannon divergence, is the expected Kullback-Leibler divergence between the empirical distribution of a sample and the target.

Estimating the error criterion

In Grassberger (2003) a sample-based estimate for the Shannon entropy $H(\pi)$ is derived correcting for the bias in the maximum likelihood estimator $H(\hat{\pi})$. Let K denote the number of non-empty bins in $\hat{\pi}_m$ after drawing m samples. If m_i is the number of samples in the i th non-empty bin, such that $\sum_{i=1}^K m_i = m$, then the entropy estimator given in Grassberger (2003) is

$$H_G(\pi) = \log(m) - \frac{1}{m} \sum_{i=1}^K m_i \psi(m_i),$$

where

$$\psi(x) = \frac{d}{dx} \log \left(\int_{t=0}^{\infty} e^{-t} t^{x-1} dt \right).$$

Substituting this estimate for the entropy into (3.2) gives the following estimate for e_{KL} (Heard and Turcotte, 2013a) based on a single run of a sampler,

$$\hat{e}_{\text{KL},m} = \frac{1}{m} \sum_{i=1}^K \phi(m_i), \tag{3.3}$$

where,

$$\phi(m_i) = m_i \{\log(m_i) - \psi(m_i)\}.$$

Calculation of (3.3) during sampling can be updated at each iteration very quickly. Let i' be the bin in which the m th observation falls. Then

$$\hat{e}_{\text{KL},m} = \frac{(m-1)\hat{e}_{\text{KL},m-1} + \Delta^1 \phi(m_{i'} - 1)}{m}. \quad (3.4)$$

where Δ^j is the j th forward difference operator.

Rival samplers

Now consider L target distributions $\pi_1, \dots, \pi_L$. Let m^* be the maximum total number of random samples that can be drawn across these distributions given a fixed computational resource. The choice of how to balance the samples sizes between the samplers is made according to minimising a loss function on the resulting error levels implied for each sampler by (3.3).

Let m^j be the number of the current total m samples assigned to the sampler for π_j and $\hat{e}_{\text{KL},m^j}$ the corresponding Monte Carlo error for that process. In Heard and Turcotte (2013a) two natural loss functions for combining the individual errors into an overall performance error are considered.

The first loss function is the average Monte Carlo error across the samplers,

$$\mathcal{L}_{\text{ave}}(m^1, \dots, m^L) = \frac{1}{L} \sum_{j=1}^L e_{\text{KL},m^j}^j,$$

and the second is the maximum error of the L samplers,

$$\mathcal{L}_{\text{max}}(m^1, \dots, m^L) = \max_{j \in \{1, \dots, L\}} e_{\text{KL},m^j}^j.$$

Minimising either of these two loss functions will imply different sample size strategies.

Determining which samplers should be afforded more computational effort is considered in

a sequential setting, conditioning on the information learnt so far about the target distributions. Having taken $m < m^*$ samples, with m^j of these allocated to the j th sampler, the decision problem is to choose from which sampler to draw the $(m + 1)$ th sample such that the chosen loss function is minimised.

For analysing the VAST data, concern will be focused on minimising the maximum error, $\mathcal{L}_{\max}(m^1, \dots, m^L)$. It is clear that the optimal decision for allocating one more sample under this loss function is to allocate it to the sampler with the highest estimated error,

$$\operatorname{argmax}_j \hat{e}_{\text{KL}, m^j}^j.$$

Heard and Turcotte (2013a) similarly defines a strategy to minimise $\mathcal{L}_{\text{ave}}(m^1, \dots, m^L)$.

As the sequential allocation scheme described above depends on random draws from the samplers, this leads to a random stopping rule for the number of samples allocated to the different target distributions. Stopping rules can introduce bias into Monte Carlo estimates (Mendo and Hernando, 2006). Here this bias will arise if the first few samples taken from a target distribution have a particularly low estimated Monte Carlo error according to (3.3). This will arise, for example, if the first random samples happen to fall into the same bin. Any Monte Carlo estimates of interest will be biased towards estimates of this character. To eradicate this bias, Heard and Turcotte (2013a) suggest a minimum number of samples ℓ^j should be drawn from each of the L target distributions to prevent degenerate sample sizes. Algorithm 3 describes the algorithm of Heard and Turcotte (2013a) for sequentially allocating samples to rival samplers when the chosen loss function for combining Monte Carlo errors across samplers is $\mathcal{L}_{\max}$.

3.3.2 VAST data

The variable sample size allocation method is now demonstrated on the VAST incoming call data, where the distributions that are considered as rivals for sampling effort are the change-point posteriors, using the approximate model for seasonality, for each of the 400 individuals in the network. Note that this is equivalent to using Algorithm 2 with Lines 6:10 replaced by

Algorithm 3 Sample allocation for rival samplers

```

1: for  $j = 1 : L$  do
2:   Draw  $\ell^j$  samples from  $\pi_j$  and calculate  $\hat{\pi}_j$ , the binned empirical estimate of  $\pi_j$ ; let
      $K_j$  be the number of non-empty bins in  $\hat{\pi}_j$ , and  $m_1^j, \dots, m_{K_j}^j$  be the corresponding
     bin counts; set  $m^j = \ell^j$ , and  $m = \sum_{j=1}^L m^j$ 
3:   Calculate the divergence estimate for the  $j$ th sampler,  $\hat{e}_{\text{KL},m^j}^j$ , using (3.3)
4: end for
5: Iterate until  $m \leq m^*$ 
6:   Set  $j^* = \text{argmax}_j \hat{e}_{\text{KL},m^j}^j$ 
7:   Sample one new observation from sampler  $j^*$ 
8:   Let  $i$  be the bin into which the new observation falls. If bin  $i$  was previously empty,
     set  $K_{j^*} = K_{j^*} + 1$  and let  $m_i^{j^*} = m_i^{j^*} + 1$ 
9:   Set  $m^{j^*} = m^{j^*} + 1$  and  $m = m + 1$ 
10:  Update  $\hat{e}_{\text{KL},m^j}^j$  using (3.4)

```

Algorithm 3, where $\pi_j = \pi(\tau_{1:k}^j | s_{1:l}^*, \mu_{0:l}^*, y^j([0, T]))$. RJMCMC, as described in Section 3.1, was used to sample from the distributions with 10,000 burnin samples discarded and then one in every fifty samples being retained.

To discretise the distributions, for each dimension, corresponding to the the number of changepoints, the interval $[0, 10]$ was divided into 100 equally sized bins. This choice distinguished well the varying complexity of the different samplers. If the bin sizes are too large then the strategy would simplify to working with marginal posterior distributions of the number of changepoints, as in the limit of the bin width becoming arbitrarily large all samples within the same dimension would fall into the same bin. Whereas, in the limit of the bin sizes becoming arbitrarily small the binned empirical distribution after m samples would have m non-empty bins; therefore when calculating the Monte Carlo error, (3.3) would be indistinguishable between the different samplers and the samplers would all be allocated equal sample sizes.

It is assumed for this analysis that the computational limit corresponds to a maximum number of samples $m^* = 400,000,000$. If an equal size sample strategy were to be used this would correspond to the sample size for each individual $m^{(j)} = 1,000,000$. The initial sample size for each individual $\ell^{(j)}$ was chosen to be 5,000. This choice performed well in that the final sample sizes allocated to each sampler were stable.

To measure the performance of the sample allocation strategy given in Algorithm 3, the

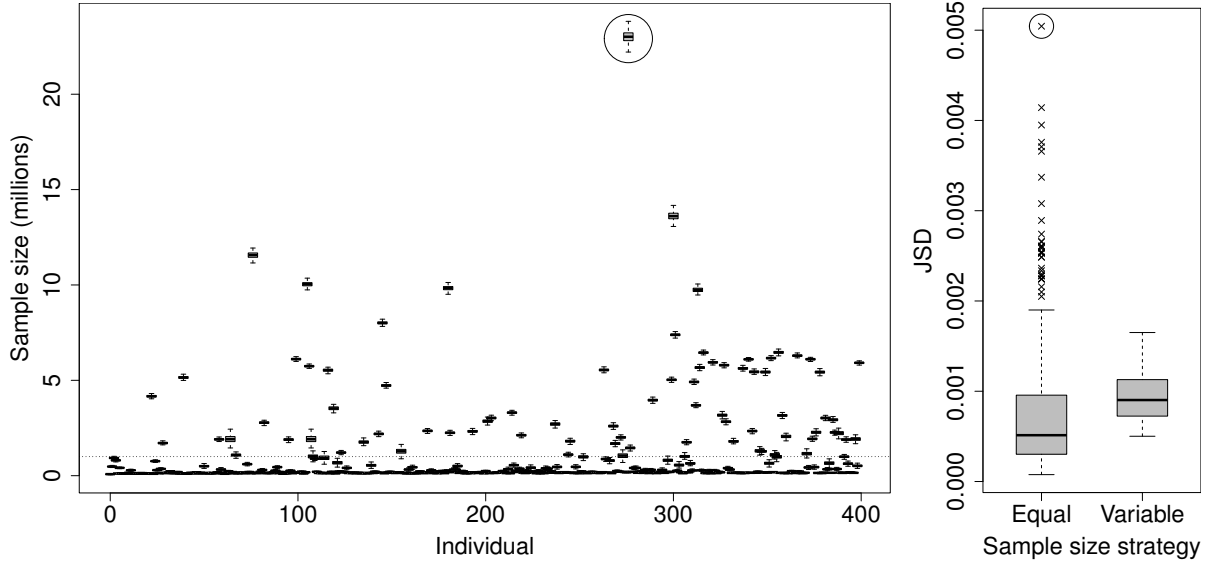


Figure 3.6: Left: Box plot of the sample sizes $m^{(j)}$ allocated to each individual over the 200 runs. Right: Box plot of the Jensen-Shannon divergence (JSD) under an equal sample size strategy and a variable sample size strategy for each individual.

Jensen-Shannon divergence (3.1) is evaluated over the empirical distributions

$$\pi(\tau_{1:k}^j | s_{1:l}^*, \mu_{0:l}^*, y^j([0, T])), \quad j = 1, \dots, 400$$

over $R = 200$ runs. This provides a good estimate of the true value of the Monte Carlo divergence error e_{KL} , given in (3.2). This dynamic allocation strategy is compared against the equal sample size strategy where $m^{(j)} = 1,000,000$ for $j = 1 \dots, 400$.

Figure 3.6 shows boxplots for both the samples sizes for each individual and the overall Jensen-Shannon divergence under the equal sample size and variable sample size strategies. The circled points correspond to the individual obtaining the maximum error under the fixed sample size strategy. Clearly the variable sample size strategy performs better according to the loss function $\mathcal{L}_{\text{max}}$. The maximum error is significantly lower than that of the fixed sample size strategy, and the variance of the errors across the individuals is also smaller under the variable sample size strategy.

3.3.3 Discussion

In Heard and Turcotte (2013a) the methodology described in Section 3.3.1 was demonstrated on a piecewise constant Poisson process changepoint problem. The variable sample size strategy using the error criterion given by (3.3) was compared against some alternative novel dynamic sample size methods based on other measures of Monte Carlo error. One method included in that comparison utilised the average of the variance of the function of interest $g_t(t)$ given by (2.16) over a grid of reference points across a single run of a RJMCMC sampler,

$$\hat{e}_{g(t),m} = \frac{1}{G} \sum_{t=1}^G \hat{\mathbb{V}}_{\pi}(g_t(t)),$$

as a measure of Monte Carlo error in place of $\hat{e}_{\text{KL},m}$ in Algorithm 3. In Heard and Turcotte (2013a) the error criterion $\hat{e}_{\text{KL},m}$ was shown to perform better than using $\hat{e}_{g(t),m}$ when using the Jensen-Shannon divergence as a measure of performance.

However, if interest was in minimising the maximum variance of the function of interest $g_t(t)$ across the L processes then using $\hat{e}_{g(t),m}$ as an error criterion should perform better. However, if the precise nature of the functionals of the distribution that will eventually be of interest are unknown or there are several functions of interest, then understanding how well $\hat{\pi}_m$ estimates π would be preferable. For example, if the random sample to be drawn is to be used in an intermediary step of a sequential Monte Carlo sampler, as will be the case in Chapter 4.

Chapter 4

Real time inference

Consider the situation where the data arrive as a stream, which will be analysed over some sequence of increasing times $t_0 < t_1 < t_2 < \dots$. This will provide a sequence of changepoint posteriors over the intervals $\{[t_0, t_n]\}$ for $n = 1, 2, \dots$, and interest will focus on the online estimation of the changepoint posteriors. Using MCMC methods to sample from the sequence of posteriors sequentially would have a computational complexity increasing at least linearly with n and would not exploit the similarity of the distributions over time. Much of the literature concerning sequential inference for both the discrete and continuous changepoint problems uses sequential Monte Carlo (SMC) methods. In this chapter, after a review of SMC sampling methods, a new SMC strategy for changepoint problems is presented that is computationally fast and performs well.

4.1 Sequential Monte Carlo

SMC methods are Monte Carlo numerical methods used to sample from a sequence of target distributions of increasing dimension known pointwise up to a normalising constant and have gained much interest in recent years; in particular, they are used in the context of sequential Bayesian inference (Doucet et al., 2001). Much of the research in this area has been related to particle filters for state space models, see Doucet and Johansen (2009).

In the context of changepoint detection Fearnhead and Liu (2007) extend the method in

Fearnhead (2006) to sequential inference where recursions are used to sample from the distribution of the changepoints in discrete time exactly; the cost of exact simulation increases with time so an approximation using particle filtering is presented. Chopin (2007) and Fearnhead and Clifford (2003) look at discrete changepoint detection in time series by reformulating the changepoint problem as a hidden Markov model and use particle filters to propagate forward the distribution of the time since the most recent changepoint.

More relevant to the continuous time model of interest for this work are the SMC samplers of Del Moral et al. (2006), which form a more general methodology for SMC and are used to sample sequentially from a sequence of target distributions defined on a common space. This methodology was demonstrated on a similar changepoint problem to the one considered here, where the target distributions are defined on a sequence of nested trans-dimensional spaces. Whiteley et al. (2011) uses this methodology for making sequential inference on piecewise deterministic processes.

Following Del Moral et al. (2006) a very brief introduction is given below to importance sampling and sequential importance sampling, which are the foundations of SMC methods, for the case when the sequence of distributions are defined on a common space.

4.1.1 Importance sampling

For $n > 0$, let $\pi_{[t_0, t_n]}(x_n)$ be a target density at time t_n for a corresponding random variable X_n defined on a common measurable space E , such that

$$\pi_{[t_0, t_n]}(x_n) = \frac{\gamma_{[t_0, t_n]}(x_n)}{Z_{[t_0, t_n]}}$$

where $\gamma_{[t_0, t_n]} : E \rightarrow \mathbb{R}^+$ is known pointwise but $Z_{[t_0, t_n]}$, which is the normalising constant of $\pi_{[t_0, t_n]}$, may be unknown.

Importance sampling (IS) is a fundamental Monte Carlo method that introduces an importance density $q_{[t_0, t_n]}(x_n)$ so that expectations with respect to $\pi_{[t_0, t_n]}$ can be regarded instead as

expectations with respect to the importance density $q_{[t_0, t_n]}(x_n)$. Specifically,

$$\begin{aligned}\mathbb{E}_{\pi_{[t_0, t_n]}}[g(x_n)] &= \frac{1}{Z_{[t_0, t_n]}} \int g(x_n) w_n(x_n) q_{[t_0, t_n]}(x_n) dx_n \\ &= \frac{1}{Z_{[t_0, t_n]}} \mathbb{E}_{q_{[t_0, t_n]}}[g(x_n) w_n(x_n)],\end{aligned}\quad (4.1)$$

where $g(x_n)$ is a function of interest mapping $E \rightarrow \mathbb{R}$ and

$$w_n(x_n) = \frac{\gamma_{[t_0, t_n]}(x_n)}{q_{[t_0, t_n]}(x_n)} \quad (4.2)$$

is the unnormalised importance weight.

Given a sample, $\{x_n^{(i)}\}_{i=1}^N$, drawn from the importance density, the Monte Carlo approximation of $q_{[t_0, t_n]}(x_n)$,

$$\hat{q}_{[t_0, t_n]}(x_n) = \frac{1}{N} \sum_{i=1}^N \delta_{x_n^{(i)}}(x_n), \quad (4.3)$$

is used to obtain a numerical approximation for the expectations in (4.1):

$$\hat{\mathbb{E}}_{\pi_{[t_0, t_n]}}[g(x_n)] = \frac{1}{Z_{[t_0, t_n]} N} \sum_{i=1}^N g(x_n^{(i)}) w_n(x_n^{(i)}).$$

If $Z_{[t_0, t_n]}$ in (4.1) is unknown, this can also be approximated using (4.3) by noting that $Z_{[t_0, t_n]}$ can be rewritten as an integral w.r.t. $q_{[t_0, t_n]}$,

$$Z_{[t_0, t_n]} = \int_E w_n(x_n) q_{[t_0, t_n]}(x_n) dx_n.$$

This gives the following Monte Carlo estimate

$$\hat{\mathbb{E}}_{\pi_{[t_0, t_n]}}[g(x_n)] = \sum_{i=1}^N g(x_n^{(i)}) W_n(x_n^{(i)}),$$

where $W_n(x_n^{(i)})$ correspond to the normalised weights.

4.1.2 Sequential importance sampling

Sequential importance sampling (SIS) extends the idea of IS when sampling from a sequence of related target distributions. The importance density is then evolved sequentially over time, utilising the sampled particles from the previous time point to exploit the similarity of the target distributions over time.

At time t_{n-1} assume there is a weighted sample $\{x_{n-1}^{(i)}, w_{n-1}^{(i)}\}_{i=1}^N$ targeting $\pi_{[t_0, t_{n-1}]}$, where the sample $\{x_{n-1}^{(i)}\}_{i=1}^N$ was drawn from an importance distribution $q_{[t_0, t_{n-1}]}(x_{n-1})$ and weighted according to (4.2). Then later at t_n the particles are propagated forward using a chosen transition kernel, K_n , and then reweighted, so that the implied importance density is now

$$q_{[t_0, t_n]}(x_n) = \int_E q_{[t_0, t_{n-1}]}(x_{n-1}) K_n(x_n | x_{n-1}) dx_{n-1}. \quad (4.4)$$

In many cases this integral cannot be evaluated pointwise and Del Moral et al. (2006) presents methodology for these scenarios where the state space is extended by a sequence of backward kernels giving an artificial joint target distribution $\pi_{[t_0, t_n]}(x_{1:n})$ on the extended space E^n . In this case the sequence of augmented target probability distributions are defined on dimensions that are increasing over time; SIS can then be performed over these extended spaces using the ‘standard’ SMC framework in Doucet et al. (2001). As $\pi_{[t_0, t_n]}(x_{1:n})$ admits $\pi_{[t_0, t_n]}(x_n)$ as a marginal, SIS provides estimates of this distribution and its normalising constants. Comparison with this methodology will be made in Section 4.4.3.

Finally, most SMC algorithms introduce resampling to deal with the problem of weight degeneracy, which is an increase in the variance of the weights that can occur over time. Resampling is a mechanism by which the weighted sample is approximated with an unweighted sample. This will eliminate particles with low weights whilst replicating those with larger weights. Standard resampling schemes include multinomial, residual, stratified or systematic resampling (Robert and Casella, 2005). Stratified, systematic and residual resampling schemes are lower variance approaches than multinomial resampling and are more commonly used.

Resampling is often carried out when the effective sample size (ESS) (Liu, 1996),

$$\text{ESS} = \frac{1}{\sum_{i=1}^N (W^{(i)})^2} \quad (4.5)$$

drops below a predetermined threshold, ESS_T , which is usually taken to be half of the total number of particles. After resampling the unnormalised weights are then reset to 1.

Extensions to the above algorithm, such as using Markov kernels to jitter the particles at each time step have been explored to improve the efficiency of the algorithm and deal with problems such as sample impoverishment; see Doucet et al. (2001) and the references therein.

4.2 Sequential Monte Carlo algorithm for changepoint detection

The motivation behind the SMC algorithm presented here is for detecting changepoint in continuous time piecewise processes as considered in Whiteley et al. (2011) and introduced in Section 1.2, where the parameters of the model have been marginalised out. The posterior distributions for the unknown changepoints are defined over a sequence of time indices t_n , so that (1.4) is now written

$$\pi_{[t_0, t_n]}(\tau_{1:k_n}, k_n | y([t_0, t_n])) = \frac{\gamma_{[t_0, t_n]}(\tau_{1:k_n}, k_n, y([t_0, t_n]))}{Z_{[t_0, t_n]}},$$

for $\tau_{1:k_n} \in \mathbb{T}_{n, k_n}$ where $\mathbb{T}_{n, k_n} = \{\tau_{1:k_n} : t_0 < \tau_1 < \dots < \tau_{k_n} < t_n\}$.

This sequence of distributions has support over the nested trans-dimensional spaces

$$E_n = \bigcup_{k_n=0}^{\infty} \{k_n\} \times \mathbb{T}_{n, k_n},$$

where $E_n \subset E_{n+1}$.

It will be useful to also introduce the following notation: let $\tilde{k}_n$ be a random variable for the number of changepoints in an interval $(t_{n-1}, t_n]$, and let

$$\pi_{(t_{n-1}, t_n]}(\tilde{\tau}_{1:\tilde{k}_n}, \tilde{k}_n | y((t_{n-1}^*, t_n])) = \frac{\gamma_{(t_{n-1}, t_n]}(\tilde{\tau}_{1:\tilde{k}_n}, \tilde{k}_n, y((t_{n-1}^*, t_n]))}{Z_{(t_{n-1}, t_n]}} \quad (4.6)$$

denote the posterior distribution of changepoints on the space

$$\tilde{E}_n = \bigcup_{\tilde{k}_n=0}^{\infty} \{\tilde{k}_n\} \times \{\tilde{\tau}_{1:\tilde{k}_n} : t_{n-1} < \tilde{\tau}_1 < \dots < \tilde{\tau}_{\tilde{k}_n} < t_n\}$$

given the data observed in the extended interval $(t_{n-1}^*, t_n]$ for some time point t_{n-1}^* satisfying $t_0 \leq t_{n-1}^* \leq t_{n-1}$, where

$$Z_{(t_{n-1}, t_n]} = \int_{\tilde{E}_n} \gamma_{(t_{n-1}, t_n]}(\tilde{\tau}_{1:\tilde{k}_n}, \tilde{k}_n, y((t_{n-1}^*, t_n])) d\tilde{\tau}_{1:\tilde{k}_n}, \quad (4.7)$$

which is the marginal likelihood of the data $y((t_{n-1}^*, t_n])$ under a changepoint model that only admits changepoints inside $(t_{n-1}, t_n]$.

When referring to distributions from $[t_0, t_n]$ the dependence on k_n , the number of changepoints, and the data will be omitted as in previous chapters so that $\pi_{[t_0, t_n]}(\tau_{1:k_n}, k_n | y([t_0, t_n])) = \pi_{[t_0, t_n]}(\tau_{1:k_n})$. However, when referring to distributions over an interval $(t_{n-1}, t_n]$ the dependency on the data and $\tilde{k}_n$ will be retained.

To provide a computationally fast SMC algorithm, the proposal distributions for the sequence of target distributions $\pi_{[t_0, t_n]}$, for $n = 1, 2, \dots$, will sample the changepoints for each update interval $(t_{n-1}, t_n]$ independently from the posterior distribution for previous intervals but will condition on the data observed since an earlier time $t_{n-1}^* \in [t_0, t_{n-1}]$.

At time t_{n-1} assume that there is a set of weighted particles $\{\tau_{1:k_{n-1}}^{(i)}, w_{n-1}^{(i)}\}_{i=1}^N$ approximating $\pi_{[t_0, t_{n-1}]}(\tau_{1:k_{n-1}})$. To allow the changepoints for $(t_{n-1}, t_n]$ to be sampled efficiently and independently from the existing set of particles at t_{n-1} , those particles are first summarised by constructing an estimate t_{n-1}^* of the last changepoint before t_{n-1} (*cf.* Chopin, 2007; Fearnhead and Clifford, 2003). See Section 4.2.2 for a discussion on how to choose t_{n-1}^* . Conditional on t_{n-1}^* , new sub-particles can be sampled from $\pi_{(t_{n-1}, t_n]}(\tilde{\tau}_{1:\tilde{k}_n}, \tilde{k}_n | y((t_{n-1}^*, t_n]))$, (4.6), which restricts changepoints to lie inside $(t_{n-1}, t_n]$ but fits them to the extended interval $(t_{n-1}^*, t_n]$. The resulting importance distribution $q_{[t_0, t_n]}(\tau_{1:k_n})$ is only known up to proportionality through

$q_{[t_0, t_n]}(\tau_{1:k_n}) = \bar{q}_{[t_0, t_n]}(\tau_{1:k_n})/Q_{[t_0, t_n]}$, where

$$\begin{aligned}\bar{q}_{[t_0, t_n]}(\tau_{1:k_n}) &= \gamma_{[t_0, t_1]}(\tau_{1:k_1}) \prod_{j=2}^n \gamma_{(t_{j-1}, t_j]}(\tau_{k_{j-1}+1:k_j}, k_j - k_{j-1}, y((t_{n-1}^*, t_n])) \\ &= \bar{q}_{[t_0, t_{n-1}]}(\tau_{1:k_{n-1}}) \gamma_{(t_{n-1}, t_n]}(\tau_{k_{n-1}+1:k_n}, k_n - k_{n-1}, y((t_{n-1}^*, t_n))).\end{aligned}$$

Note that this corresponds to SIS where the transition kernel in (4.4) is given by

$$K_n(\tau_{1:k_n} | \tau'_{1:k_{n-1}}) \propto \delta'_{\tau'_{1:k_{n-1}}}(\tau_{1:k_{n-1}}) \gamma_{(t_{n-1}, t_n]}(\tau_{k_{n-1}+1:k_n}, k_n - k_{n-1}, y((t_{n-1}^*, t_n))).$$

In practice, to obtain a weighted sample $\{\tau_{1:k_n}^{(i)}\}_{i=1}^N$ at time t_n , a sample $\{\tilde{\tau}_{1:\tilde{k}_n}^{(i)}\}_{i=1}^N$ obtained from $\pi_{(t_{n-1}, t_n]}(\tilde{\tau}_{1:\tilde{k}_n}, \tilde{k}_n | y((t_{n-1}^*, t_n)))$ is augmented with the vectors of changepoints from the existing weighted sample at t_{n-1} giving a new sample:

$$\{\tau_{1:k_n}^{(i)} = (\tau_1^{(i)}, \dots, \tau_{k_{n-1}}^{(i)}, \tilde{\tau}_1^{(i)}, \dots, \tilde{\tau}_{\tilde{k}_n}^{(i)})\}_{i=1}^N. \quad (4.8)$$

The weights then need to be updated to account for the discrepancy between the importance distribution and the changepoint posterior distribution, which doesn't assume that the distribution of changepoints on $\tilde{E}_n$ is independent of previous intervals. For $n > 1$ the weight for the i th particle is given by

$$w_n(\tau_{1:k_n}^{(i)}) = \frac{\gamma_{[t_0, t_n]}(\tau_{1:k_n}^{(i)})}{\bar{q}_{[t_0, t_n]}(\tau_{1:k_n}^{(i)})} = w_{n-1}(\tau_{1:k_{n-1}}^{(i)}) \bar{w}_n(\tau_{1:k_n}^{(i)}), \quad (4.9)$$

where

$$\bar{w}_n(\tau_{1:k_n}^{(i)}) = \frac{\gamma_{[t_0, t_n]}(\tau_{1:k_n}^{(i)})}{\gamma_{[t_0, t_{n-1}]}(\tau_{1:k_{n-1}}^{(i)}) \gamma_{(t_{n-1}, t_n]}(\tau_{k_{n-1}+1:k_n}^{(i)}, k_n^{(i)} - k_{n-1}^{(i)}, y((t_{n-1}^*, t_n)))}, \quad (4.10)$$

is the incremental weight. At $n = 1$, $t_{n-1}^* = t_0$ and the unnormalised weights are trivially one. For $n > 1$ the incremental weight is computationally quick to calculate and in the special case of $t_{n-1}^* = t_{n-1}$ is almost equivalent to calculating the probability of accepting a reversible jump

death move at t_{n-1} .

Note that if RJMCMC is used to sample from $\pi_{(t_{n-1}, t_n]}(\tilde{\tau}_{1:k_n}, \tilde{k}_n | y((t_{n-1}^*, t_n]))$ in each update interval implying the samples will be correlated, then before joining the particles together the sample from the new interval should be permuted each time to break auto-correlation of the combined particles.

4.2.1 Normalising constant estimates

Both the target and the proposal distributions are only known up to proportionality. Recall

$$Z_{[t_0, t_n]} = \int_{E_n} \gamma_{[t_0, t_n]}(\tau_{1:k_n}) d\tau_{1:k_n},$$

which is the marginal likelihood of the data $y([t_0, t_n])$, and let

$$Q_{[t_0, t_n]} = \int_{\tilde{E}_n} \bar{q}_{[t_0, t_n]}(\tau_{1:k_n}) d\tau_{1:k_n}$$

be the normalising constant for the proposal distribution. Given a sample $\{\tau_{1:k_n}^{(i)}\}_{i=1}^N$ from the proposal distribution at t_n , the Monte Carlo approximation given in (4.3) yields the following estimate for the ratio of the normalising constants:

$$\frac{\widehat{Z_{[t_0, t_n]}}}{Q_{[t_0, t_n]}} = \frac{1}{N} \sum_{i=1}^N w_n^{(i)}. \quad (4.11)$$

Note that if $t_{n-1}^* = t_{n-1}$, then

$$Q_{[t_0, t_n]} = Q_{[t_0, t_{n-1}]} Z_{(t_{n-1}, t_n]}, \quad (4.12)$$

where $Z_{(t_{n-1}, t_n]}$ is defined in (4.7). This result will be utilised in Section 4.2.2.

4.2.2 Choices of t_{n-1}^*

The question addressed here is how to efficiently generate samples from the posterior distribution over the new update window $(t_{n-1}, t_n]$ given the weighted particles from the target distribution over $[t_0, t_{n-1}]$. Although the proposal distribution for changepoints over the update interval conditions on the data from t_{n-1}^* changepoints are only sampled from $(t_{n-1}, t_n]$. For example, in the case of Poisson process data with conjugate priors for the intensity levels between changepoints as considered in Section 2.1, this is equivalent to proposing changepoints from $\pi_{(t_{n-1}, t_n]}(\tilde{\tau}_{1:\tilde{k}_n}, \tilde{k}_n | y((t_{n-1}, t_n]))$ with a revised prior distribution for the first intensity level in $(t_{n-1}, t_n]$ given by

$$\text{Gamma}(\alpha + r_{n-1}^*, \beta + t_{n-1} - t_{n-1}^*),$$

where r_{n-1}^* is the number of events in $(t_{n-1}^*, t_{n-1}]$.

One option would be to define t_{n-1}^* as an estimate of the time of the most recent changepoint before t_{n-1} , $\tau_{k_{n-1}}$. This is desirable as it provides information about how much of the data before t_{n-1} should be considered as part of the current regime on entering $(t_{n-1}, t_n]$. Given a weighted set of particles $\{w_{n-1}^{(i)}, \tau_{1:k_{n-1}}^{(i)}\}_{i=1}^N$ from $\pi_{[t_0, t_{n-1}]}(\tau_{1:k_{n-1}})$ a Monte Carlo estimate of the posterior expectation of $\tau_{k_{n-1}}$ is

$$\hat{\tau}_{k_{n-1}} = \sum_{i=1}^N \tau_{k_{n-1}}^{(i)} W_{n-1}^{(i)}. \quad (4.13)$$

Other statistical summaries, such as taking the MAP of the weighted set of particles, could be considered as an estimate for $\tau_{k_{n-1}}$.

One other option would be to simply set $t_{n-1}^* = t_{n-1}$. If several processes are being observed simultaneously over time, it could be that you often get intervals where there are identical data. In this case having $t_{n-1}^* = t_{n-1}$ would be desirable as a set of particles targeting $\pi_{(t_{n-1}, t_n]}(\tau_{1:\tilde{k}_n}, \tilde{k}_n | y((t_{n-1}, t_n]))$ for some arbitrary interval $(t_{n-1}, t_n]$, where for example $\int_{t_{n-1}}^{t_n} y(t)dt = 0$ in a Poisson process, could be pre-sampled and then shared in real time when sampling for intervals where identical data are observed.

The limitation of this choice of t_{n-1}^* is that when extending the particles for the interval $(t_{n-1}, t_n]$ the assumption of independence between update intervals of the proposal distribution prevents the possibility of a changepoint near t_{n-1} being properly explored. The following solution utilising Bayes factors is added to the SMC algorithm when t_{n-1}^* is chosen to be t_{n-1} .

Bayes factors for changepoint detection near t_{n-1}

At t_n let $\mathcal{M}_1$ be the model described by $\pi_{[t_0, t_n]}(\tau_{1:k_n})$, and $\mathcal{M}_2$ be the model $\tilde{\pi}_{[t_0, t_n]}(\tau_{1:k_n}) = \pi_{[t_0, t_{n-1}]}(\tau_{1:k_{n-1}})\pi_{(t_{n-1}, t_n]}(\tau_{k_{n-1}+1:k_n}, k_n - k_{n-1} | y((t_{n-1}, t_n]))$.

Then the Bayes Factor in favour of $\mathcal{M}_1$ over $\mathcal{M}_2$ at t_n is

$$\text{BF}_n = \frac{Z_{[t_0, t_n]}}{Z_{[t_0, t_{n-1}]} Z_{(t_{n-1}, t_n]}} = \frac{Z_{[t_0, t_n]}}{Q_{[t_0, t_n]}} \frac{Q_{[t_0, t_{n-1}]}}{Z_{[t_0, t_{n-1}]}} ,$$

from (4.12).

Given $\{\tau_{1:k_n}^{(i)}, w_n^{(i)}\}_{i=1}^N$, the estimate from (4.11) leads to an estimate for this Bayes Factor of

$$\widehat{\text{BF}}_n = \frac{\sum_{i=1}^N w_n^{(i)}}{\sum_{i=1}^N w_{n-1}^{(i)}} . \quad (4.14)$$

If $\widehat{\text{BF}}_n$ is less than some threshold BF_T , then the particles need to be modified in a targeted way to reflect the knowledge that there is a high probability that a changepoint was missed near t_{n-1} .

To target this area, a small number of RJMCMC moves are applied to each particle, where for the birth move changepoints are generated from $N(t_{n-1}, \sigma^2(t_n - t_{n-1})^2)$ with a reflective boundaries at $[t_0, t_n]$. A good choice of $\sigma^2 = 1/9$, say, this corresponds to 99.7% of the proposed changepoints falling in the interval $(t_{n-2}, t_n]$. Due to the invariance of the reversible jump Markov Kernel with respect to $\pi_{[t_0, t_n]}$, the same particle weights are retained following these RJMCMC moves.

The threshold BF_T controls how much evidence is required in favour of $\mathcal{M}_1$ over $\mathcal{M}_2$. Choosing $\text{BF}_T = 0.1$, which corresponds to *strong evidence* (Kass and Raftery, 1995) against $\mathcal{M}_1$, performed well in the example presented in Section 4.4.1.

4.3 Adaptive sequential Monte Carlo

Standard SMC practice would be to fix a number of particles N to sample from the proposal distribution for changepoints over each new interval observed and then join them with the previous sample according to (4.8). In many practical applications the intervals $\{(t_{n-1}, t_n]\}$ will be sufficiently small that *a priori* the probability of having only 0 or 1 changepoints is high. A smaller number of samples, $M \leq N$, could instead be drawn from $\pi_{(t_{n-1}, t_n]}(\tau_{1:\tilde{k}_n}, \tilde{k}_n | y((t_{n-1}^*, t_n]))$. However, it still might be necessary to have a larger number of particles N to represent the overall distribution from $[t_0, t_n]$. In this case, one could then consider increasing the particle set sampled over $(t_{n-1}, t_n]$ by making $N - M$ copies of the M particles sampled from $\pi_{(t_{n-1}, t_n]}$ and then joining it with the N overall particles, so that (4.8) is now

$$\{\tau_{1:k_n}^{(i)} = (\tau_1^{(i)}, \dots, \tau_{k_{n-1}}^{(i)}, \tilde{\tau}_1^{(1+\text{mod}(i-1, M))}, \dots, \tilde{\tau}_{\tilde{k}_n}^{(1+\text{mod}(i-1, M))})\}_{i=1}^N. \quad (4.15)$$

for $n > 1$. If RJMCMC is used to obtain the samples over each new interval then sampling would be computationally fast and quick to converge over the small interval $(t_{n-1}, t_n]$ and again permuting the M samples would break autocorrelation.

For many applications there will be multiple processes arriving as a stream on which to make inference, such as the case when doing anomaly detection on networks as is the interest for this thesis. Rather than fixing the number of particles to be sampled on the posterior over each new interval observed for each data stream; it would be desirable to make use of the variable sample size allocation strategy given in Algorithm 3 so that proposal distributions on $(t_{n-1}, t_n]$ with greater complexity are allocated more samples since these proposal distributions are intended to resemble the posterior distributions for the changepoints on $(t_{n-1}, t_n]$. To create an adaptive SMC algorithm, a variable number of particles determined according to Algorithm 3 can be sampled on each new update interval.

Assume at t_{n-1} , the j th process has N^j weighted samples approximating $\pi_{[t_0, t_{n-1}]}$ and after sampling particles according to Algorithm 3 from the proposal distributions for the L processes over the interval $(t_{n-1}, t_n]$ the j th process has been assigned M^j samples.

If $M^j < N^j$ then the N^j particles from $[t_0, t_{n-1}]$ would be joined with the M^j particles from $(t_{n-1}, t_n]$ according to (4.15). However, if $M^j > N^j$ then $M^j - N^j$ copies are needed from the weighted sample at t_{n-1} .

Rather than randomly resampling particles from the N^j particle set according to their weights $\{w_{n-1}^{j,(i)}\}_{i=1}^{N^j}$ it would be preferable to increase the particle set in a deterministic way to maximise the ESS (4.5), which is equivalent to minimising the sum of the squared weights. The following section describes how to increase a weighted particle set from N to M , say, in such a way as to minimise the sum of the squared weights. Note that the maximum number of particles a process will ever be assigned will correspond to the total maximum number of particles m^* allowed across all L processes in Algorithm 3.

4.3.1 Increasing the weighted particle set

Since this section will be concerned with duplicating particles from the weighted particle set, it will be useful to note that there may be already some duplicated particles. For continuous time changepoints this would be the particles with no changepoints. Also note that any duplicated particles will all have the same weight.

For simplicity of notation, assume now that the weighted particle set has been labelled such that the first $N' < N$ particles are unique. For $1 \leq i \leq N$, let $m_0^{(i)}$ be the number of replicates of $\tau_{1:k_{n-1}}^{(i)}$ in the N particles, and define $\bar{w}^{(i)} = w_{n-1}^{(i)} m_0^{(i)}$. Then note that $\{\tau_{1:k_{n-1}}^{(i)}, \bar{w}^{(i)}\}_{i=1}^{N'}$ is an equivalent representation of the full weighted particle set, since

$$\sum_{i=1}^{N'} \bar{w}^{(i)} \delta_{\tau_{1:k_{n-1}}^{(i)}} (\tau_{1:k_{n-1}}) \equiv \sum_{i=1}^N w_{n-1}^{(i)} \delta_{\tau_{1:k_{n-1}}^{(i)}} (\tau_{1:k_{n-1}}).$$

Since an algorithm is required to further duplicate some of the weighted particle set, it is necessary to work with this reduced representation. Otherwise, the algorithm would admit the possibility of making different numbers of copies of the same particle.

Assume that each unique particle i will be replicated $m^{(i)}$ times, so that $\sum_{i=1}^{N'} m^{(i)} = M$. Then in order to minimise the sum of the squared weights and ensure that any Monte Carlo

Algorithm 4 Increasing particle set from N to M

```

1: Set  $m^{(i)} = m_0^{(i)}$  and  $\bar{w}^{(i)} = m_0^{(i)} w^{(i)}$  for  $i = 1, \dots, N'$ . Let  $m = \sum_{i=1}^{N'} m^{(i)} = N$ 
2: Calculate  $\delta_j = \frac{(\bar{w}^{(j)})^2}{(m^{(j)}+1)m^{(j)}}$  for  $j = 1, \dots, N'$ 
3: Let  $j_{N'} = \operatorname{argmax}_j \{\delta_j : j = 1, \dots, N'\}$ 
4: while  $m < M$  do
5:   Let  $j_{N'-1} = \operatorname{argmax}_j \{\delta_j : j = 1, \dots, N', j \neq j_{N'}\}$ 
6:   Let

$$x_{N'} = \min \left( M - m, \operatorname{ceil} \left[ -0.5 - m^{(j_{N'})} + \sqrt{\frac{(\bar{w}^{(j_{N'}))^2}{\delta_{j_{N'-1}}} + 0.25} \right] \right)$$

7:   Let  $m^{j_{N'}} = m^{j_{N'}} + x_{N'}$  and  $m = m + x_{N'}$ 
8:   Let  $\delta_{j_{N'}} = \frac{(\bar{w}^{(j_{N'})})^2}{(m^{(j_{N'})}+1)m^{(j_{N'})}}$ 
9:   Let  $j_{N'} = j_{N'-1}$ 
10: end while
11: Let  $i' = 1$ 
12: for  $i = 1 : N'$  do
13:   for  $j = 1 : m^{(i)}$  do
14:      $\tau_{1:k_{n-1}}^{*(i')} = \tau_{1:k_{n-1}}^{(i)}$ 
15:      $w_{n-1}^{*(i')} = \frac{\bar{w}^{(i)}}{m^{(i)}}$ 
16:      $i' = i' + 1$ 
17:   end for
18: end for
19:  $(\tau_{1:k_{n-1}}^{(i)}, w_{n-1}^{(i)}) \leftarrow (\tau_{1:k_n}^{*(i)}, w_{n-1}^{*(i)})$  for  $i = 1, \dots, M$ 

```

estimates are the same after the particle set has been increased the revised weight is $\bar{w}^{(i)}/m^{(i)}$.

The sum of the squared weights after replicating the sample would then be:

$$\sum_{i=1}^{N'} \frac{(\bar{w}^{(i)})^2}{m^{(i)}}$$

Choosing optimal values $\{m^{(i)}\}_{i=1}^{N'}$ for increasing the particle set from N to M particles so that the resulting sum of squared weights is minimised is a complex optimisation problem, and solving this directly would add too much computational burden to the overall SMC algorithm. So instead, Algorithm 4 presents a sequential optimisation method.

The quantity δ_j , calculated in Line 2, represents the decrease in the sum of the squared

weights if the j th particle was replicated once:

$$\delta_j = \sum_{i=1}^{N'} \frac{(\bar{w}^{(i)})^2}{m^{(i)}} - \sum_{i \neq j} \frac{(\bar{w}^{(i)})^2}{m^{(i)}} - \frac{(\bar{w}^{(j)})^2}{m^{(j)} + 1} = \frac{(\bar{w}^{(j)})^2}{(m^{(j)} + 1)m^{(j)}}.$$

$x_{N'}$, calculated in Line 6, is the number of replicates made of the optimally chosen particle $j_{N'}$ at the current iteration, which is the largest integer solving the following inequality:

$$\frac{(\bar{w}^{(j_{N'})})^2}{(m^{(j_{N'})} + x_{N'} + 1)(m^{(j_{N'})} + x_{N'})} < \delta_{j_{N'-1}}.$$

4.3.2 Algorithm

A summary of the fully adaptive SMC algorithm is given in Algorithm 5. The computational effort of this algorithm does not increase over time and is parallelisable when used across multiple data streams.

Other standard techniques for improving SMC performance, such as the Resample-Move algorithm, Gilks and Berzuini (2001), where RJMCMC moves are applied to the particle set after ESS resampling to introduce diversity can also be applied to Algorithm 5.

Finally, in the simplest case where $t_{n-1}^* = t_{n-1}$, note that when the BF drops the ESS will often also drop below the specified threshold. This is due to a small proportion of the sampled particles possibly having a changepoint near the interval boundary t_{n-1} and therefore being much higher weighted than the other particles. If this is the case, then the RJMCMC moves should be applied to the particle set *after* resampling is carried out as this will provide a larger number of distinct particles to approximate the target distribution (Doucet and Johansen, 2009).

4.4 Examples

Some illustrative examples are now considered, where the data generating process is assumed to be a Poisson process with a conjugate prior for the intensity so that $\pi_{[t_0, t_n]}(\tau_{1:k_n})$ is known up to

Algorithm 5 SMC algorithm for changepoint detection

```

1:  $n = 1$ 
2: Sample  $\{\tau_{1:k_1}^{(i)}\}_{i=1}^N \sim \pi_{[t_0, t_1]}(\tau_{1:k_1})$  and set  $w_1^{(i)} = 1$  for  $i = 1, \dots, N$  where  $N$  is either fixed
   or chosen according to Algorithm 3
3:  $n \leftarrow n + 1$ 
4: Sample  $\{\tilde{\tau}_{1:\tilde{k}_n}^{(i)}\}_{i=1}^M \sim \pi_{(t_{n-1}, t_n]}(\tilde{\tau}_{1:\tilde{k}_n}, \tilde{k}_n | y((t_{n-1}^*, t_n]))$  where  $M$  is either fixed or chosen
   according to Algorithm 3
5: if  $M > N$  then
6:   Let  $N'$  be the number of unique particles at  $t_{n-1}$  and replicate the particles ac-
   cording to Algorithm 4 giving a weighted set of  $M$  particles  $\{\tau_{1:k_{n-1}}^{(i)}, w_{n-1}^{(i)}\}_{i=1}^M$  at
    $t_{n-1}$ .
7:   Let  $N = M$ 
8: end if
9:  $\{\tau_{1:k_n}^{(i)} = (\tau_1^{(i)}, \dots, \tau_{k_{n-1}}^{(i)}, \tilde{\tau}_1^{(1+\text{mod}(i-1, M))}, \dots, \tilde{\tau}_{\tilde{k}_n}^{(1+\text{mod}(i-1, M))})\}_{i=1}^N$ 
10: Calculate  $w_n^{(i)}$  for  $i = 1, \dots, N$  according to (4.9)
11: if  $t_{n-1}^* = t_{n-1}$  then
12:   Calculate  $\widehat{\text{BF}}_n$  according to (4.14)
13: end if
14: if  $\text{ESS} < \text{ESS}_T$  then
15:   Resample  $\{\tau_{1:k_n}^{(i)}, w_n^{(i)}\}_{i=1}^N$  to obtain equally weighted particles  $\{\tau_{1:k_n}^{(i)}, 1\}_{i=1}^N$ 
16: end if
17: if  $t_{n-1}^* = t_{n-1}$  then
18:   if  $\widehat{\text{BF}}_n < \text{BF}_T$  then
19:     Sample  $\{\tau_{1:k_n}^{*(i)}\}_{i=1}^N \sim K(\tau_{1:k_n} | \tau_{1:k_n}^{(i)})$  and set  $\tau_{1:k_n}^{(i)} \leftarrow \tau_{1:k_n}^{*(i)}$  for  $i = 1, \dots, N$ 
20:   end if
21: end if
22: goto 3

```

proportionality through (2.3). First, a simulated example is considered with fixed changepoints, using $t_{n-1}^* = t_{n-1}$ in Algorithm 5, to demonstrate the use of BF-based resampling. Second, a large run of 100 simulated data sets with random changepoints is used to demonstrate the general performance of the algorithm in terms of accuracy in changepoint detection. Third, the algorithm is demonstrated on the coal-mining disaster data analysed in Green (1995) and Raftery and Akman (1986) amongst others and compared with the SMC samplers algorithm of Del Moral et al. (2006). Finally a network example is considered using the VAST data set (*cf.* Section 3.1.1 and Section 3.2.1).

For all of the examples, the ESS threshold is $N/2$ and when the ESS drops below the threshold the systematic resampling approach is used. Additionally, five RJMCMC moves with a uni-

form birth proposal are applied to the particle set after resampling when the ESS drops below the threshold. RJMCMC is used to obtain a sample from each of the update intervals for all the examples, where the moves employed are those specified in Section 3.1.

As a comparison in Sections 4.4.2 and 4.4.3, the RJMCMC algorithm as described in Section 3.1 is used as an alternative method to sample from the sequence of posteriors $\pi_{[t_0, t_n]}(\tau_{1:k_n})$ over time. The MAP obtained from the sample for $\pi_{[t_0, t_{n-1}]}(\tau_{1:k_{n-1}})$ is used as a starting value when sampling from $\pi_{[t_0, t_n]}(\tau_{1:k_n})$. This can be considered as the best way to generate samples from the posterior given the data from $[t_0, t_n]$, although carrying a much higher computational cost. This approach will be referred to as sequential MCMC (SMCMC).

Given $\{\tau_{1:k_n}^{(i)}, w_n^{(i)}\}_{i=1}^N$ the Monte Carlo estimate for $\mathbb{P}_\pi(h_\delta(t) < 0)$ (see Section 2.3) is

$$\hat{\mathbb{P}}_\pi(h(t) < 0) = \sum_{i=1}^N I(h_\delta^{(i)}(t) < 0) W_n^{(i)}.$$

For the examples below, approximations calculated sequentially on each interval $t_{n-1} < t \leq t_n$ over time using the posterior density estimate at t_n are referred to as *online* estimates. It may be desirable to revise the estimates of $\mathbb{P}_\pi(h_\delta(t) < T)$ over $t_0 < t \leq t_n$ in light of the updated sample, as in the examples in Sections 4.4.2 and 4.4.4; such estimates will be referred to as *retrospective* estimates.

All the stated run times in this section are based on C++ code run on an HP Z400 workstation with a single hex-core Intel Xeon W3680 CPU, clocked at 3.33 GHz.

4.4.1 Simple example

To demonstrate the BF-based resampling performance, Algorithm 5 is applied to a single simulated Poisson process data set with two changepoints over a period of 100 time units where $t_{n-1}^* = t_{n-1}$. The SMC algorithm is run with incremental time steps of two, which corresponds to 50 densities over time, given up to proportionality by (2.3). The changepoints are fixed so that one changepoint, at 30, falls on an update interval boundary and would be the type of changepoint the algorithm might typically miss when $t_{n-1}^* = t_{n-1}$ without BF resampling,

whilst the other changepoint, at 60.7, falls $1/3$ of the way into an update interval. The intensities are simulated from the gamma prior with $\alpha = 1$ and $\beta = 0.04$, see Section 2.1.

The number of overall SMC particles $N = 5,000$ and the number of particles on each update interval $M = 500$, see Section 4.3. The prior parameters for the intensities are set equal to those from which the data were marginally simulated from, and the prior parameter for the changepoints $\nu = 0.02$. Using uninformative priors for the intensities similar to the ones chosen in Section 3.1.1 gives comparable performance. The BF threshold is chosen to be 0.1 and RJMCMC moves with a targeted Gaussian birth proposal (see Section 4.2.2) are applied to the particle set when the BF drops below the threshold.

Figure 4.1 shows the ESS (rescaled by a factor of 1,500), BF (on the log scale) and the online estimate of the expected number of changepoints. After joining the particles from $[0, 30]$ and $(30, 32]$ according to (4.15) and reweighting, the BF and the ESS estimate both drop due to the changepoint at that interval boundary. After resampling and applying targeted RJMCMC moves to the particle set there is a corresponding jump in the expected number of changepoints.

Figure 4.1 also shows the online estimate of $\mathbb{P}_\pi(h_\delta(t) < 0)$ where $\delta = \min(t, 2)$ and both changepoints are clearly identified.

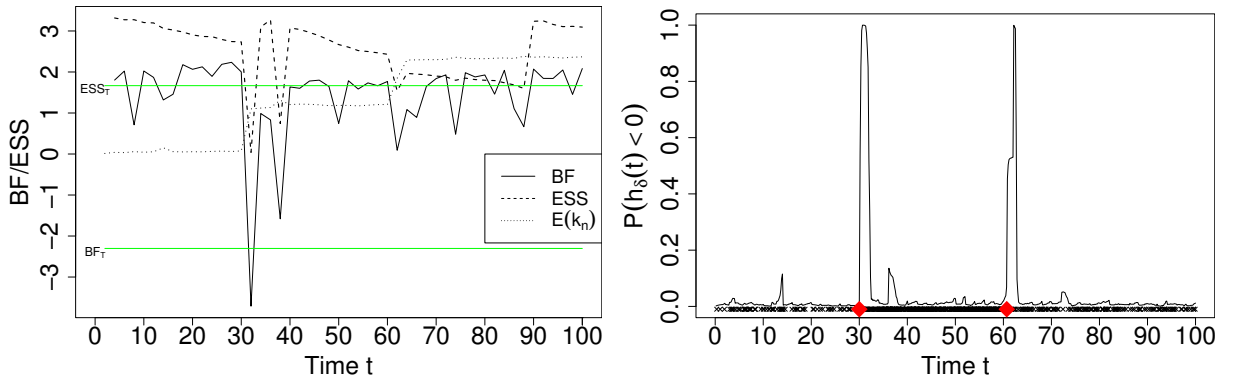


Figure 4.1: Left: ESS, BF and the online estimate of the expected number of changepoints. Right: Online estimate of $\mathbb{P}_\pi(h_{\min(t,2)}(t) < 0)$.

4.4.2 Multiple simulated data sets

To test the general performance of the algorithm in terms of accuracy of changepoint detection, it is now run on a set of 100 simulated inhomogeneous Poisson process data sets. Each data set is simulated over 100 time units, with the intensities simulated from the prior using the same parameters as Section 4.4.1. The changepoints are also simulated from the prior with $\nu = 0.02$, so each data set will have on average two changepoints, with a constraint that they have to be a distance of 5 time units apart.

The SMC algorithm is run with incremental time steps of two corresponding to 50 densities. The overall number of particles $N = 5,000$ and $t_{n-1}^* = \hat{\tau}_{k_{n-1}}$, given by (4.13). The same choices of prior parameters as Section 4.4.1 are used. The algorithm is run twice to vary the number of particles sampled on each update interval, first using $M = 500$ and then $M = 1,000$. At each time point, t_n , $\mathbb{P}_\pi(h_\delta(t) < 0)$ with $\delta = \min(t, 2)$ is recalculated retrospectively over the interval $[t_0, t_n]$ using the estimated posterior density at t_n . A changepoint is considered detected when the retrospective estimate of $\mathbb{P}_\pi(h_{\min(t,2)}(t) < 0)$ exceeds the specified significance level; further details are given below.

As a rival method of SMC, SMCMC is used to sample from each of the 50 posteriors, $\pi_{[t_0, t_n]}$, over time, with a sample size of 5,000. For both the SMC algorithm and SMCMC algorithm when using RJMCMC to sample from the intervals $(t_{n-1}, t_n]$ and $[t_0, t_n]$ respectively, an initial burn-in of 5,000 iterations is discarded, and then subsequently one in every ten samples is retained. The algorithms are run 200 times for the different methods to compare the sample variance of the estimates obtained. Finally as a comparison for performance an extra run of SMCMC with 1,000,000 samples is used. This provides a more accurate representation of the posteriors over time and can be considered as close to the optimal performance that could be achieved, albeit at a much higher computational cost.

The criteria used to assess performance are as follows.

- **Detected changepoints.** As the changepoints are continuous random variables, an assumption has to be made about what is considered a detected changepoint. For these

simulations a simulated changepoint, t_c , is considered detected if $\exists t$ s.t. $\hat{\mathbb{P}}_\pi(h_{\min(t,2)}(t) < 0) > 1 - p$ and $\hat{\tau}_{i^*}(t) \in [t_c - 3, t_c + 3]$, where $\hat{\tau}_{i^*}(t)$ is the corresponding estimate at time t for the location of the nearest changepoint as defined in (2.19).

- **False detections.** An occurrence where $\hat{\mathbb{P}}_\pi(h_{\min(t,2)}(t) < 0) > 1 - p$ and the corresponding estimate $\hat{\tau}_{i^*}(t) \notin [t_c - 3, t_c + 3]$ for any simulated changepoint t_c .
- **Average change in height of intensity** from either side of those changepoints which are not detected.
- **Average update intervals until detection.** For this statistic the changepoints are treated as if they were at the end of the update interval in which they lie, as that is the earliest time at which they could be detected. Then for a single detected changepoint at $t_c \in (t_{n-1}, t_n]$, the number of update intervals until detection would be $d - n$ where $t_d \geq t_n$ is the first time that $\exists t$ s.t. both $\hat{\mathbb{P}}_\pi(h_{\min(t,2)}(t) < 0) > 1 - p$ and the corresponding estimate $\hat{\tau}_{i^*}(t) \in [t_c - 3, t_c + 3]$.
- **Average squared distance from simulated changepoint.** For a single detected changepoint at t_c let the squared distance estimation error be $(t_c - \hat{\tau}_{i^*}(t))^2$, where $\hat{\tau}_{i^*}(t)$ is the estimate for the location of the changepoint at the first time when it is detected.

Table 4.1 shows the results for two different changepoint detection significance levels for the function of interest, 70% and 90%. To enable fair comparisons across the methods, the last two statistics of the criteria listed above are calculated only for the set of changepoints that are detected at both significance levels for all of the runs for the SMC algorithm, for both $M = 500$ and $M = 1,000$, and the SMC MC algorithms. A total number of 129 out of 185 originally sampled changepoints were detected for all these methods across all the runs.

The performance of the SMC algorithm with $M = 1,000$ and $M = 500$ is seen to be comparable with SMC MC with $N = 5,000$. The sample standard deviation for the various statistics is only slightly smaller with SMC MC than the SMC algorithm showing that SMC is giving near optimal performance in terms of Monte Carlo error. As would be expected, for

all of the methods it is observed that as the changepoint detection significance level increases, there are more missed changepoints but fewer false detections; similarly, when looking at the changepoints detected by all methods across the runs, the detection delay increases with the significance level while the average squared distance from the simulated changepoint decreases.

Table 4.1: Results from the SMC and SMCMC algorithms for the 100 simulations. For the first three rival methods the summary statistics are the averages across all the runs, and standard deviations are given in parentheses.

	SMC		SMCMC		SMCMC	
Number of particles	$N = 5000, M = 500$		$N = 5000, M = 1000$		$N = 5000$	
Significance level	70%	90%	70%	90%	70%	90%
Detected changepoints	144.2 (1.1)	134.0 (1.6)	144.3 (1.0)	134.4 (1.5)	144.7 (0.6)	135.2 (0.7)
Average change in height of intensity	3.47	4.55	3.41	4.51	3.38	4.39
False detections	1.24 (.52)	0.10 (.30)	1.12 (.32)	0.07 (.25)	1 (.00)	0.01 (.07)
Average update intervals until detection	0.31 (.010)	0.49 (.057)	0.31 (.008)	0.50 (.054)	0.31 (.006)	0.47 (.021)
Average squared distance from changepoint	0.11 (.011)	0.05 (.004)	0.11 (.012)	0.05 (.005)	0.11 (.004)	0.04 (.001)
Average times resampled per dataset	2.17		2.21		-	-
Run times (for one run)	~ 1min 55secs		~ 2min 14secs		~ 3min 54secs	-

Both SMC algorithms have faster run times than SMCMC and for SMC, choosing $M = 500$ further reduces the run time at very little cost in performance. The SMC algorithms with the two choices of M has performance similar to that of SMCMC with 1,000,000 samples, albeit with the probability of a changepoint being slightly overestimated at the 90% level with fewer samples.

4.4.3 Coal data

The coal mining disaster data consist of the times of coal-mining disasters in the UK between 1851 and 1962 and are a popular data set for changepoint analysis. In a sequential time frame this data set has been analysed in Del Moral et al. (2006) and a comparison will be made with results from the SMC samplers (SMCs) algorithm in that paper using the code provided for this example therein. It is assumed that the disasters follow a Poisson process with piecewise constant intensity, and the interest is in estimating this piecewise constant intensity sequentially. In Del Moral et al. (2006) the prior on the number and location of changepoints is also assumed to be a Poisson process; however, non conjugate priors are chosen for the intensities so that

$$p_{\text{SMCs}}(\lambda_{0:k_n} | \tau_{1:k_n}, k_n) = p(\lambda_0) \prod_{i=1}^{k_n} p(\lambda_i | \lambda_{i-1})$$

where $\lambda_0 \sim \text{Gamma}(\alpha_{\text{SMCs}}, \beta_{\text{SMCs}})$ and $\lambda_i | \lambda_{i-1} \sim \text{Gamma}(\lambda_{i-1}^2/\chi, \lambda_{i-1}/\chi)$.

To make comparable inference using the proposed SMC algorithm, which assumes conjugate priors for the intensity levels, the following particle re-weighting is proposed. Given a weighted sample of changepoints $\{\tau_{1:k_n}^{(i)}\}_{i=1}^N$ obtained from the SMC algorithm proposed here (for comparison purposes this will be referred to as SMCcp), $\lambda_{0:k_n}^{(i)}$ can first be sampled directly from the independent posterior distributions of $\{\lambda_i\}$, (2.4), to give a weighted sample of changepoints and intensities at each time point. Second, this augmented sample can then be reweighted to give an approximate sample from the model in Del Moral et al. (2006) with the non-conjugate prior for the intensities. At t_n given $\{\tau_{1:k_n}^{(i)}, \lambda_{0:k_n}^{(i)}, w_n^{(i)}\}_{i=1}^N$, the new weights $\bar{w}_n^{(i)}$

are

$$\bar{w}_n^{(i)} = w_n^{(i)} \frac{p_{\text{SMCs}}(\lambda_{0:k_n}^{(i)} | \tau_{1:k_n}^{(i)}, k_n^{(i)})}{p(\lambda_{0:k_n}^{(i)} | \tau_{1:k_n}^{(i)}, k_n^{(i)})},$$

where $p(\lambda_{0:k_n}^{(i)} | \tau_{1:k_n}^{(i)}, k_n^{(i)})$ is the distribution for the independent gamma priors (2.4) and $w_n^{(i)}$ is given in (4.9).

As in Del Moral et al. (2006) it is assumed that inference is of interest annually, which defines 112 densities over time where the n th density is defined over $[1851, n + 1851]$. The chosen prior parameter for the number of changepoints $\nu = 2/112$. For the priors for the non-conjugate model, $\alpha_{\text{SMCs}} = 4.5$ and $\beta_{\text{SMCs}} = 1.5$ as in Del Moral et al. (2006) and $\chi = 5$. For the conjugate model uninformative priors are chosen so that $\alpha = 0.1$ and $\beta = 0.1$, where $\lambda_i \sim \text{Gamma}(\alpha, \beta)$. As in Del Moral et al. (2006), the overall number of particles $N = 10,000$ and the ESS resampling threshold is set to be 3000 for both algorithms. For SMCcp the number of particles sampled on each interval $M = 10,000$ and $t_{n-1}^* = \hat{\tau}_{k_{n-1}}$, given by (4.13).

As a comparison to both SMC methods, a modification of the SMCMC algorithm is also applied to these data. Firstly, RJMCMC is used to sample from the densities over time for the conjugate model, with 1,000,000 samples drawn from each posterior over time. Secondly, the intensities are then drawn from their independent posterior distributions under the conjugate model and the sample of changepoints and intensities is then weighted so that the sample is approximately from the posterior for the non-conjugate model.

Figure 4.2 shows the online or filtered intensity function for the non-conjugate model, estimated at each time point using SMCcp, SMCs and SMCMC. Initially SMCcp performs much better than SMCs and by 1900 the performance of both algorithms are comparable. Moreover, the SMCcp method perfectly tracks the SMCMC curve, which represents the best possible inference. Figure 4.3 shows the ESS at each time point for both SMCcp and SMCs. The ESS for the SMCcp method shows good stability across the whole of the 112 year period, whereas, SMCs initially suffers from persistent weight degeneracy. In total SMCs performs resampling due to the ESS dropping below the threshold 26 times, where 19 of the times are before 1900, compared with SMCcp performing resampling only 8 times. This could partly be due to the

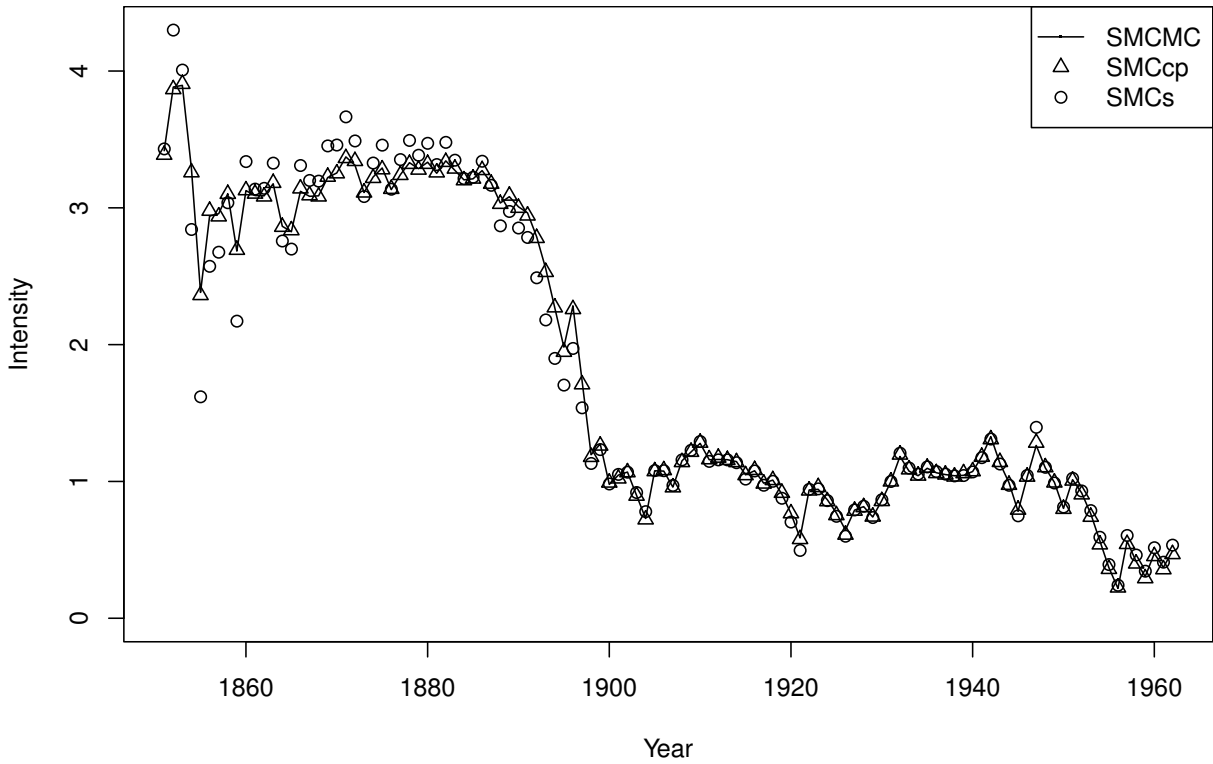


Figure 4.2: Online estimated intensity function for the coal-mining disaster data

initialisation of the particles from the prior in SMCs and the high uncertainty contained in the target distributions in the first few years.

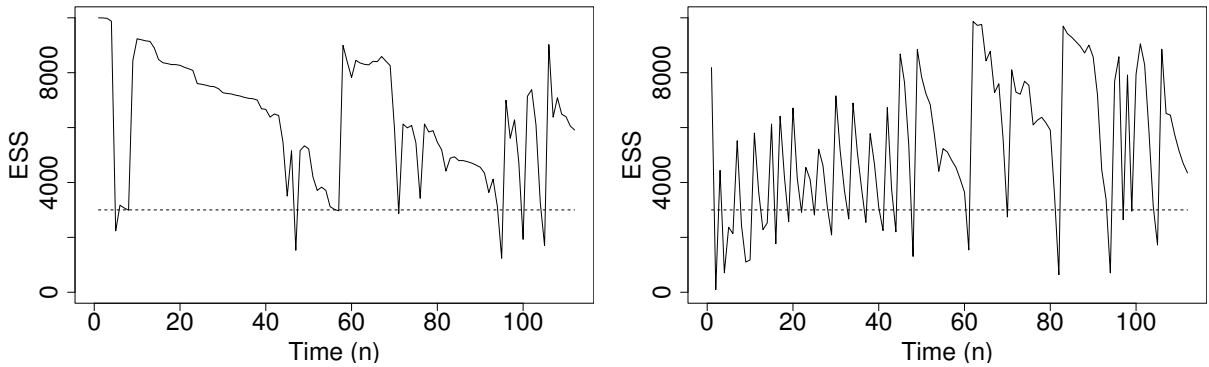


Figure 4.3: Effective sample size. Left: SMCcp Right: SMC samplers

The posterior distribution for the number of changepoints and their locations using the final weighted sample from the non-conjugate model over $[t_1, t_{112}]$ for each of the algorithms is shown in Figure 4.4. In this respect the performance of both SMCcp and SMCs is very similar to that observed when using RJMCMC to sample from the posterior over $[t_1, t_{112}]$.

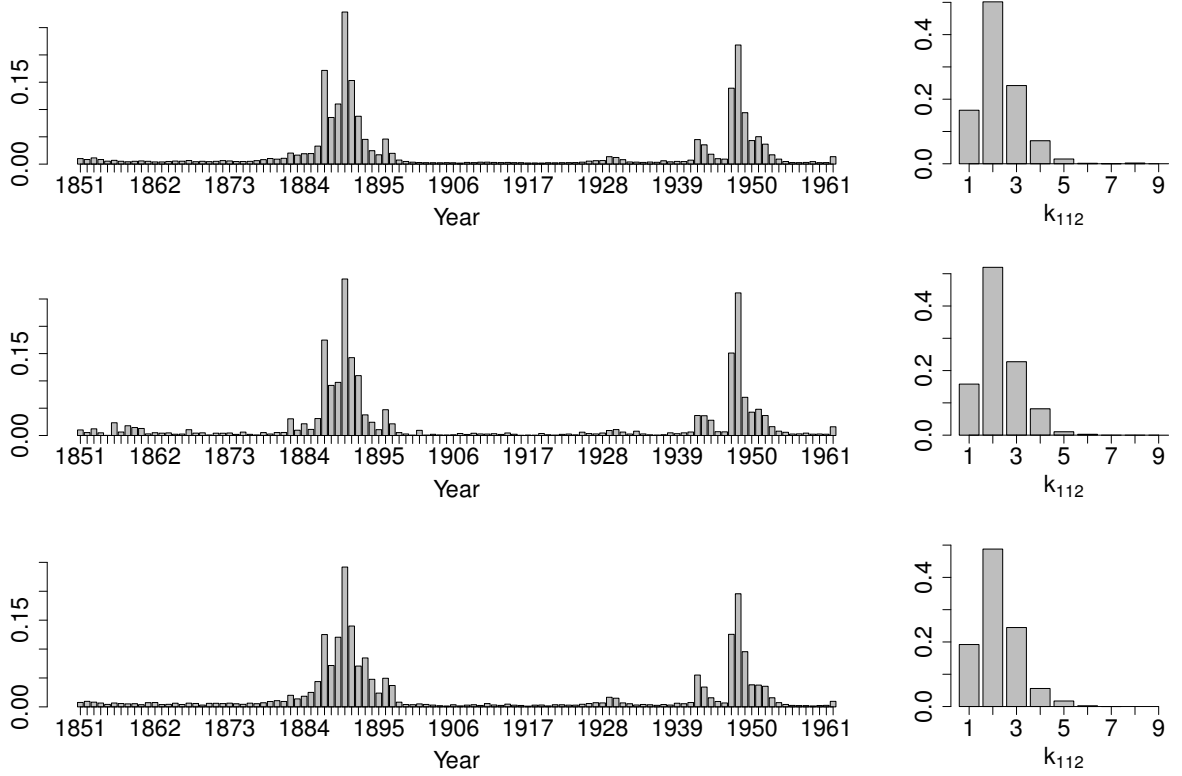


Figure 4.4: Left: Binned one-dimensional projection of the target changepoint posterior where each bar shows the probability of a changepoint falling in that bin. Right: Posterior probability of the number of changepoints. Top: Sequential MCMC. Middle: SMCcp. Bottom: SMC samplers.

4.4.4 VAST data

Finally the VAST data (Section 1.3.1) are reanalysed in a sequential time frame. To incorporate seasonality within the model in the sequential setting, the approximation used in Section 3.2.1 is implemented, where inference on the individuals' changepoints conditions on a single set of seasonal changepoints and intensity multipliers obtained from the approximate seasonal model. The MAP estimate for the seasonal changepoints obtained in Section 3.1.1 from the approximate seasonal model, which was used in the analysis for the VAST data in Section 3.2.1, is applied here. Note that this means the analysis is not strictly sequential as all of the data from the whole ten days are used to infer seasonality. Section 6.1 discusses sequential learning of seasonality, but otherwise in principle seasonality could be imagined to have been inferred beforehand using training data.

From the analyses in Sections 3.1.1 and 3.2.1, the incoming call data are known to carry more signal, with more true detections; furthermore, all the true anomalous individuals detected in the outgoing call analysis are also detected in the incoming call analysis.

Hence Algorithm 5 with $t_{n-1}^* = \hat{\tau}_{k_{n-1}}$ is run for the incoming call data for the individuals in the network with 1 hour time increments, which corresponds to 240 update intervals or target densities

$$\pi_{[t_0, t_n]}(\tau_{1:k_n}^j | s_{1:l}^*, \mu_{0:l}^*, y^j([t_0, t_n])), \quad n = 1, \dots, 240$$

over the ten days of data. Note that implicitly the proposal distribution for the SMC algorithm is also conditioning on the seasonal changepoints.

For the VAST data analysis the adaptive SMC algorithm from Section 4.3 will be used where a variable number of particles are assigned to each individual in each update window according to Algorithm 3. The maximum number of samples allowed for all individuals on each interval $m^* = 4,000,000$. If an equal size sampling strategy were to be used this would correspond to the number of particles allocated to each individual $N^j = 10,000$. The initial sample size for each individual $\ell^{(j)} = 500$. When RJMCMC sampling from each interval, a burnin of 5,000 samples was discarded and then subsequently one in every fifty samples was retained. For comparable inference, the same prior parameters used in Sections 3.1.1 and 3.2.1 are retained for the model.

The left hand side of Figure 4.5 shows the online estimate of the number of anomalous individuals found over time under the criteria used in Sections 3.1.1 and 3.2.1. The online estimate for $\mathbb{P}_\pi(h_t(t) < 0)$ for each individual is also shown. The right hand side of Figure 4.5 shows the retrospective estimate of the number of anomalous individuals found in each sub-interval and the retrospective estimate of $\mathbb{P}_\pi(h_t(t) < 0)$ for each individual, obtained from using the final weighted sample at the end of the 10 day period ($n = 240$). The retrospective plots agree with the earlier analysis of the incoming call VAST data in Section 3.2.1 using the approximate Gibbs sampler, with all the same anomalous ID's being found for the incoming call analysis (cf. Figure 3.5). In real time there is a natural small delay in the time at which the anomaly is flagged for some of the individuals.

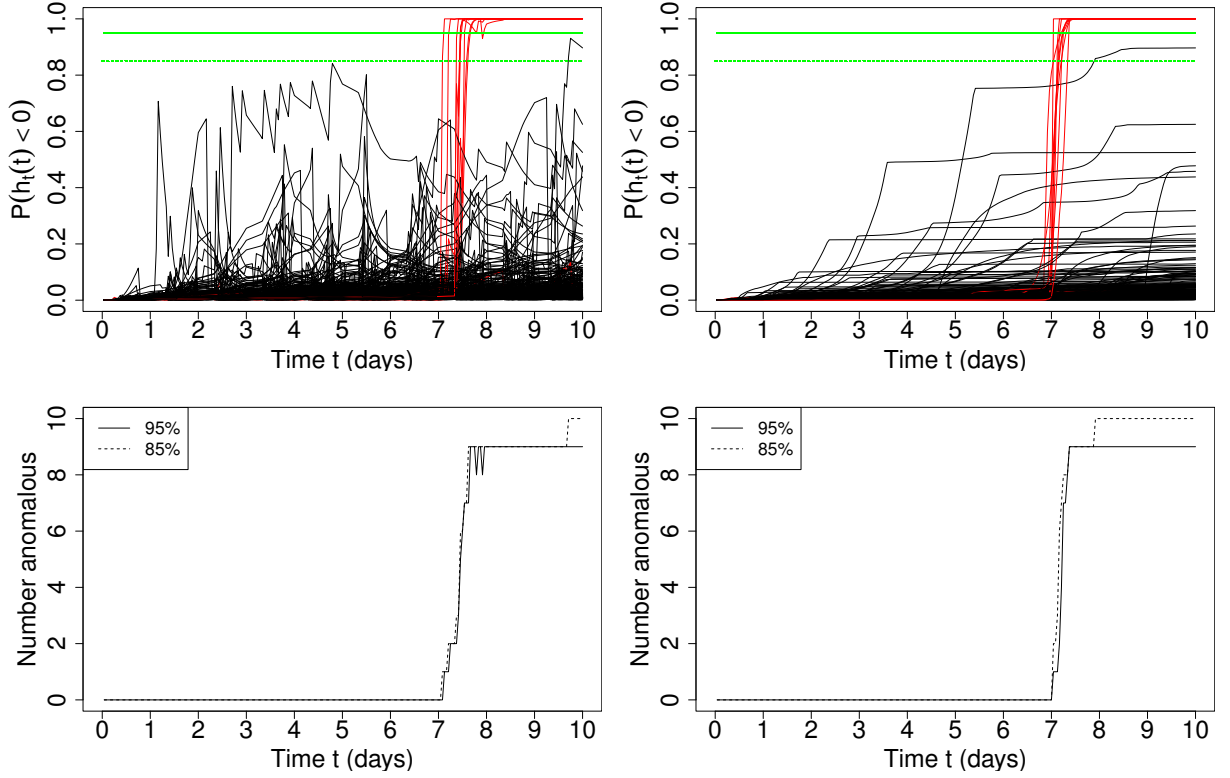


Figure 4.5: Top: $\hat{\mathbb{P}}_{\pi}(h_t(t) < 0)$, known anomalous caller ID's are in red. Bottom: Number of caller IDs considered anomalous at each time t for two different significance levels. Left: Online estimate. Right: Retrospective estimate.

For the online estimate of $\mathbb{P}_{\pi}(h_t(t) < 0)$ there is more uncertainty in the curve. This will be due to the low intensities of call data for some of the individuals, which provide a relatively weak signal; in particular, the sparse data are often not evenly spread out over the whole ten days. Hence in sequential analysis with 1 hour time increments an increase in the probability of a changepoint can temporarily be observed only to later be retracted once more data have been observed. This phenomenon is demonstrated in Figure 4.6, which plots the online estimate of $\mathbb{P}_{\pi}(h_t(t) < 0)$ for the incoming call data of ID 392. For a while this individual's estimated probability curve increases due to a slightly extended break in calls being received, until they eventually resume; then, as in the fixed time frame analysis, the presence of a changepoint becomes less clear-cut.

Figure 4.7 shows a box plot of the sample sizes $N^{(j)}$ allocated to each of the 400 individuals over each update window. The dotted line shows the sample size that would be allocated to

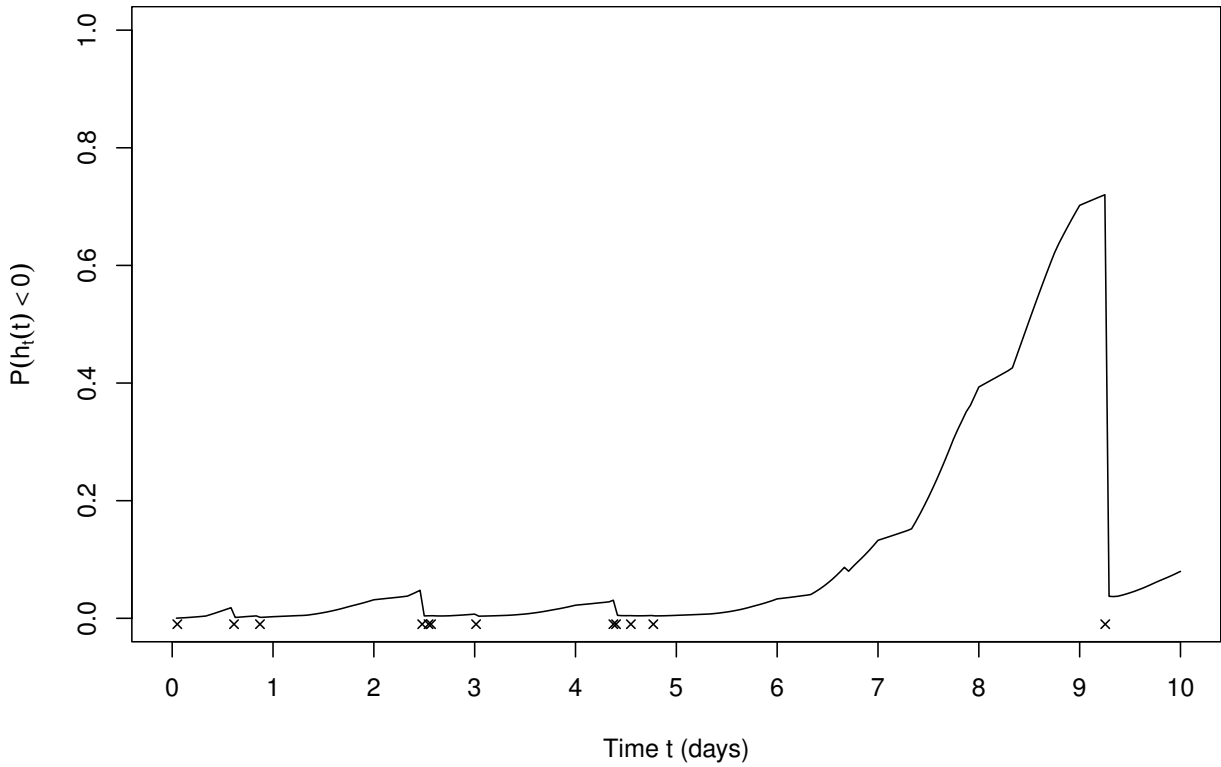


Figure 4.6: Online estimate of $\mathbb{P}_\pi(h_t(t) < 0)$ for the individual with ID 392.

each individual under a fixed sample size strategy, where $N^{(j)} = 10,000$. It is apparent that at night, when most of the individuals are quiet, the sample sizes allocated are similar to that of an equal sample size strategy; whereas during the day the sample sizes allocated are far more varied as individuals become more active, to varying degrees. The few individuals that receive larger sample sizes at night most often correspond to those that surprisingly become active at night, suggesting a more complicated distribution during that update interval.

As mentioned in Section 3.1.1, the major anomalous activity which could be detected from the incoming call analysis found individuals who usually communicated with ID's 1, 2, 3 and 5 switching to communicating with ID's 309, 397, 360 and 306 on the eighth day. The impact of the locally anomalous behaviour on sample size allocation can be seen clearly in Figure 4.8, which shows the allocated sample sizes for these 8 individuals over each of the update intervals.

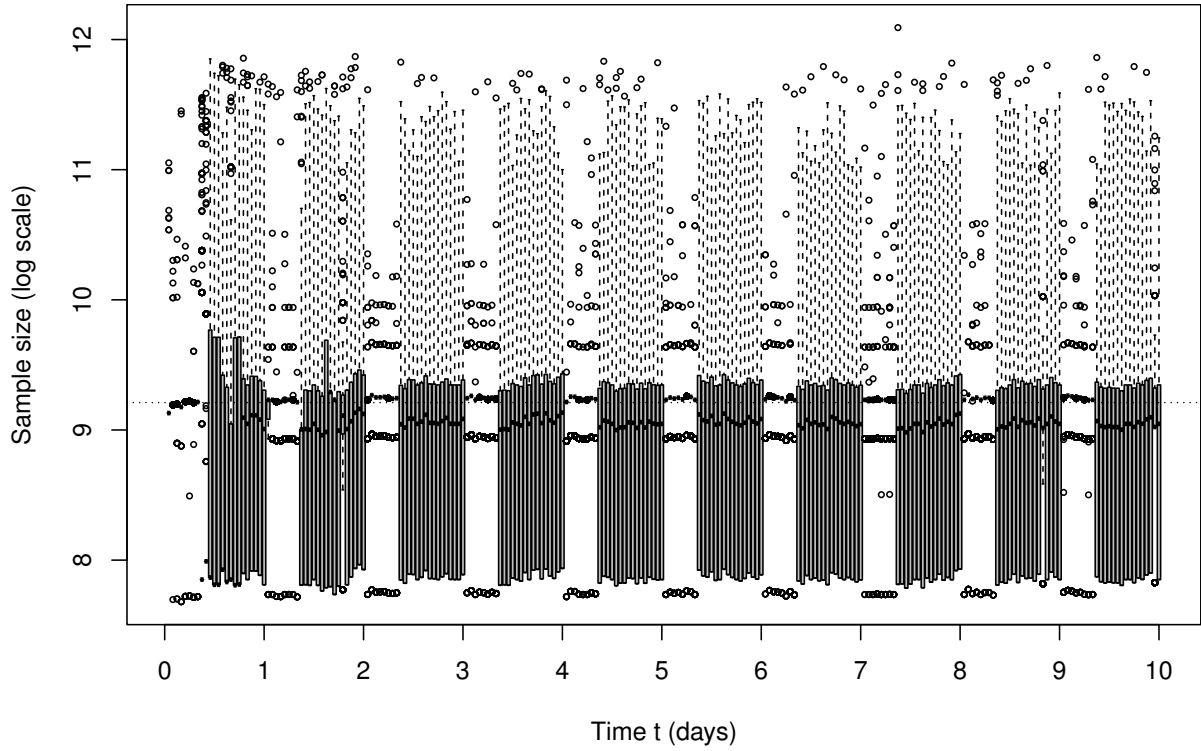


Figure 4.7: Box plot of the sample sizes $N^{(j)}$ allocated to each individual on each of the 240 update intervals. Dotted line corresponds to $N^{(j)} = 10,000$.

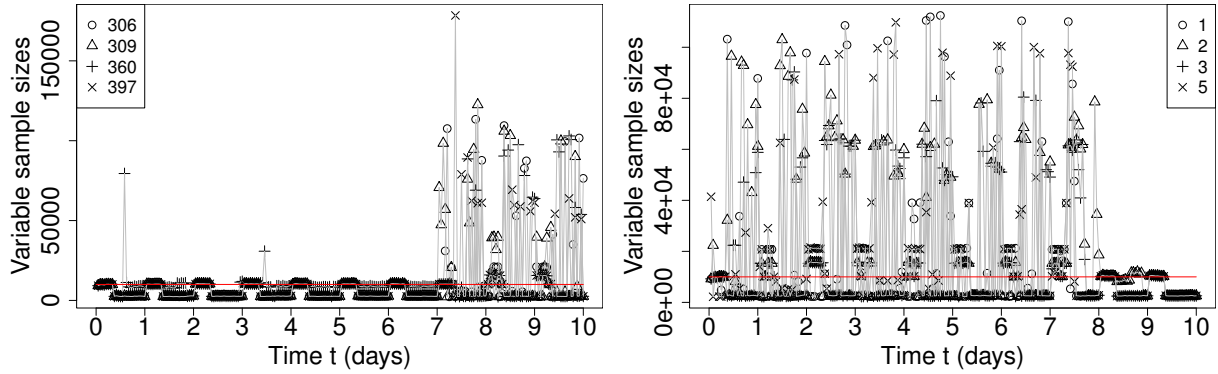


Figure 4.8: Allocated sample sizes for each update interval for 8 of the individuals identified as potentially anomalous. The red line corresponds to the fixed sample size strategy of $N^{(j)} = 10,000$.

Chapter 5

Computer network anomaly detection

This chapter addresses the problem of real time intrusion detection on an internal computer network. Specifically, detecting the presence of an intruder intending to conduct malicious activity, such as extracting valuable information or disrupting the network once they have gained access. Whilst most modern networks have security systems that try to prevent intruders from gaining access to machines in the network, these systems are notoriously permeable and it is inevitable that some intruders will succeed in penetrating the network. Once an attacker has gained access to a host on the network, they will typically then traverse the network, moving from machine to machine to establish persistent “footholds” in the network (Neil et al., 2013a); this could be to escalate their privileges within the network or gain access to machines that hold more valuable information. It is this type of anomalous movement pattern within the network, as the intruder attempts to compromise multiple machines, that this work aims to identify. Neil et al. (2013a) gives examples of the types of movements of intruders which are commonly observed on computer networks once they have been compromised.

There are two broad approaches to intrusion detection: the first is searching for evidence of attacks based on attack signatures accumulated from previous types of attacks, commonly known as signature-based detection; the second is anomaly based detection where any deviations from a model of the normal state of the system is sought. Meza et al. (2009) gives a nice overview of work that has recently been done in both the statistical and computer science com-

munities using both of these approaches. The anomaly based approach has a clear advantage in that new types of attacks that haven't been observed previously can be identified; Patcha and Park (2007) and Lazarevic et al. (2003) provide an overview of work done in this area as related to network intrusion. It is this approach that this present work is concerned with.

One area of research is in detecting anomalous substructures of the computer network graph by examining various graph heuristics, (Noble and Cook, 2003). Scan statistics have been used for analysing the e-mail communications of users in Priebe et al. (2005). In Sexton et al. (2013), graph structured hypothesis testing is used to find anomalous structures in the network based on anomaly scores for nodes in the network. The focus of this work is to model the time series of communication levels along edges or pairs of computers, whereas none of the work above takes into account edge behaviour. The usage patterns of communications between pairs of computers in a network are very bursty in nature and so the independent increment continuous time models given for communication networks in Chapter 2 are not appropriate for these types of data. For tractable inference, discrete time models will be used to model communications along edges in the network (*cf.* Lambert and Liu, 2006; Heard et al., 2010; Neil et al., 2013a).

As was the case from the continuous time perspective, the model developed in this chapter treats the activity status (the binary status of whether any communications are observed or none at all) of all nodes/computers in the network as statistically independent entities; the validity of this independence assumption partly relies on learning the seasonal behaviour of each node from some training data, and this forms a component of the baseline probability model. The seasonal behaviour of each node is treated as a changepoint problem acting over some finite seasonal period, similar to the approach taken in Section 2.2, except here discrete time models are used. In Lambert and Liu (2006) a different set of parameters are used for discrete time count models for each interval over a discretised seasonal time period. Those seasonal patterns are assumed to repeat over each seasonal period and the parameters of the model are updated over time using a EWMA approach.

Next, for those periods in which a node is active, the activity status of each of the edges emanating from that node are also treated as being conditionally independent. Together, these

two components provide probability models for the activity levels along each edge in the network. These models can be considered as an extension of Heard et al. (2010), where edges in a network are also treated as independent entities and conjugate Bayesian discrete time counting process models are used to describe the connection frequencies along edges in a network. In this work we incorporate both seasonality into the models and have a hierarchical model for the nodes and edges to capture some of the dependence between two edges emanating from the same node.

Statistical anomaly detection monitors behaviour along each edge in the network looking for outlying behaviour with respect to the fitted probability model. Whilst an edge continues to behave normally, the data observed should be used to further refine the probability model in a coherent updating scheme. Otherwise, edges can be flagged as anomalous if their current behaviour deviates significantly from past behaviours.

A second stage can then be performed to look across seemingly anomalous edges within sliding windows for subgraphs that could indicate an intruder moving around the network. This helps to filter out the noise within the network by combining anomalous edge events together to search for anomalous patterns within the network. This aggregating of anomalous edges together over sliding windows is similar to the methodology of Neil et al. (2013a) who looks for anomalous edges within the network that form a path, which is defined as a sequence of edges where the destination node of the current edge is the source node for the next edge. By looking at paths, Neil et al. (2013a) aims to detect traversal of an intruder and Figure 5.1 shows an example of how a traversal attack appears in the network. In the present work, rather than looking only for traversal through the network initiating from a single infected node, the proposed method aims to detect more general shapes.

The remainder of this chapter will be organised as follows: Section 5.1 will describe the computer network data set from Los Alamos National Laboratory (LANL) used for analysis in this chapter; Section 5.2 will give a description of the Markov chain changepoint model used to capture seasonal node behaviour; Section 5.3 will give a description of the model for the edges that comprises initially of a Markov chain model for edge activity conditional on the node being

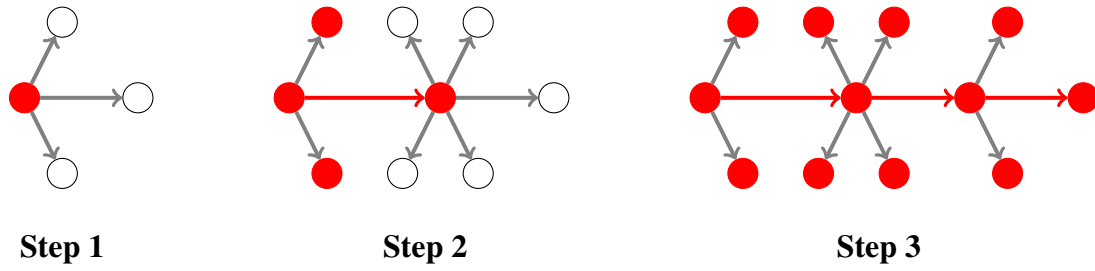


Figure 5.1: An example of a traversal attack. Filled in circles indicate compromised hosts. All edge activities are potentially anomalous. Neil et al. (2013a)

active, and then further to the edge being active, a Negative Binomial model for the edge counts (note that seasonality could in practice be modelled on an edge level rather than a node level and all the models described easily extend to this case). Section 5.4 will give a description of the second stage of the analysis whereby an anomaly graph is formed of edges that are potentially anomalous. Finally in Section 5.5, the results will be shown when the proposed method is run on the LANL data. In this chapter, demonstrations of modelling techniques of relevance to the LANL data will be given as the method is described; however the general methodology is developed with the aim of being usable on any internal computer network.

5.1 LANL data

The motivating data set used in this chapter was obtained from Los Alamos National Laboratory's large internal computer network. The data consist of NetFlow records (Sperotto et al., 2010) of the communications made between individual computers (hosts) on LANL's network, where each connection has an associated direction from a source IP address to a destination IP address. In particular, the data which will be used for analysis is a discrete time-series of summary counts of communications made between pairs of computers over consecutive 10 minute windows, following the approach of Neil et al. (2013a).

The data comprise four weeks of NetFlow records when there are no known infected hosts on the network; this will be referred to for the rest of the chapter as the training data. Further to this there are data recorded for ten days following a significant malicious attack, which will

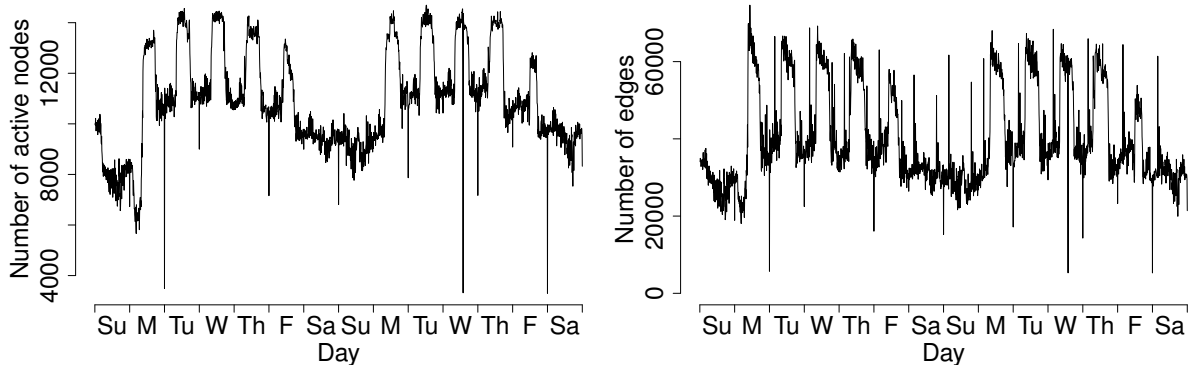


Figure 5.2: Left: Number of active nodes in the training data. Right: Number of observed edge events in the training data.

be referred to as the test data. Analysts believe that the attack started early on in the test data, and six hosts were known to be infected since they were observed being used for malicious behaviour. Note that in such cases there is no absolute ground truth, and further hosts may have been infected which were never discovered. Within the training and test data there are a total of 57,459 unique IP addresses (nodes) observed and on average there are approximately 10,000 active nodes and 40,000 observed edge events in each 10 minute window, as can be seen in Figure 5.2. Although any source computer can connect to any other computer in the network, analysis is restricted to communications between nodes that were observed in both the training and the test data as the types of anomalies sought require the intruder to make connections between computers in order to traverse around the network.

The seasonality inherent in this type of user driven data can be seen clearly in Figure 5.2. Additionally the LANL data, and other computer networks in general, show very sparse activity along edges; for the LANL data, 80% of the edges are active for less than 10% of the time. As mentioned previously, these types of data sets can be very bursty in nature and Figure 5.3 shows the activity status from an example edge for the LANL data set. Another feature of these types of data sets are volatile connection counts; it is common for edges to be quiet for long periods of time and then have extremely large one off counts. For the LANL data this volatility is demonstrated in Figure 5.4, which plots the maximum count observed against the median count for a random collection of edges in the network over the training period.

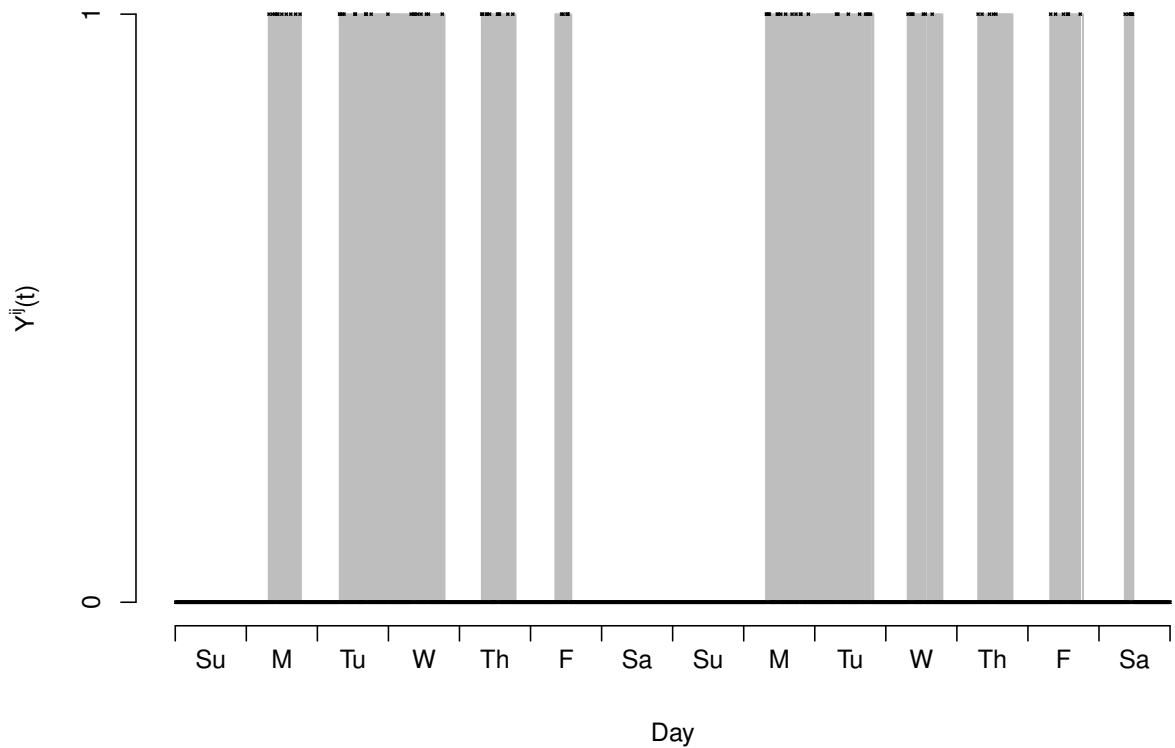


Figure 5.3: Activity status of an edge. The areas shaded in grey are the intervals where the source node is active.

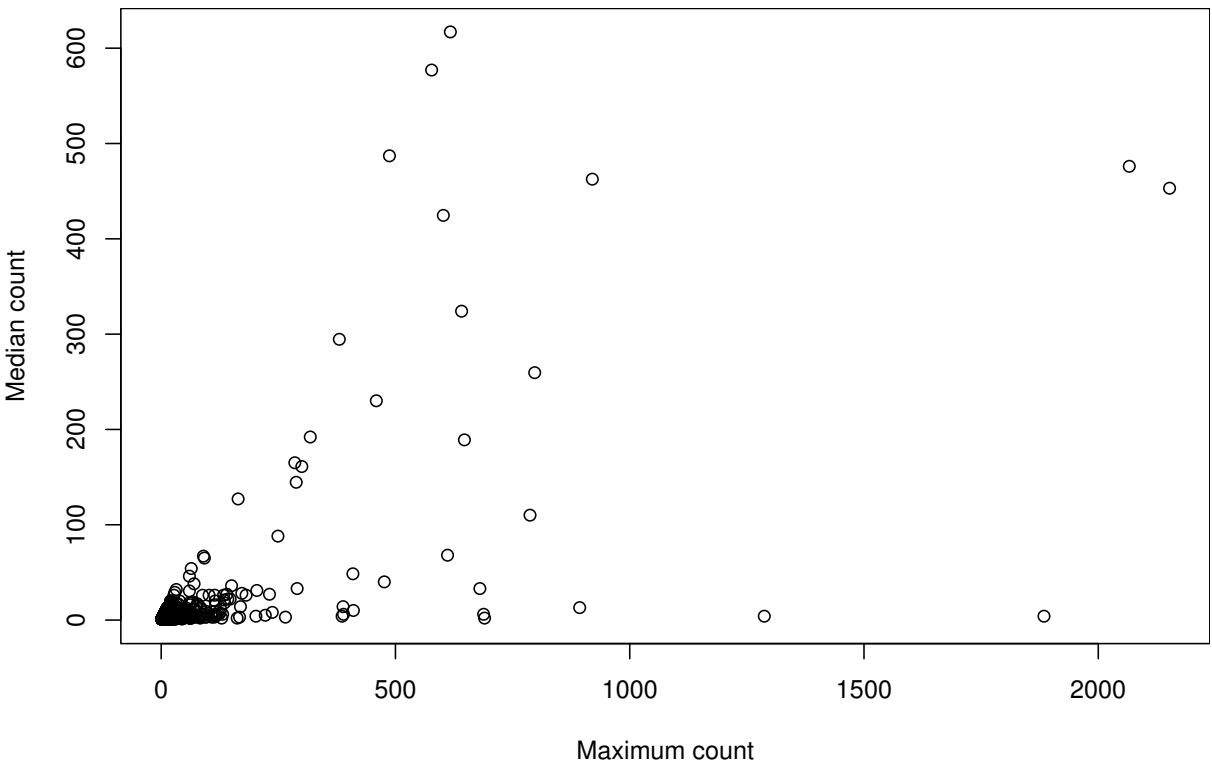
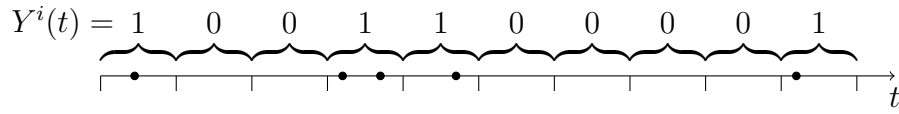


Figure 5.4: Maximum count versus the median count for the observed counts in the training period for a random sample of 3,000 edges.

5.2 Markov chain changepoint model for seasonal node behaviour

Let $N^i(t)$ be the number of connections made from node i at the t th time point, where the t th time point refers to some suitably discretised time interval, for $t = 0, 1, \dots$. For a binary perspective, let $Y^i(t) \in \{0, 1\}$ be the indicator variable for whether node i is idle or active at time t ; so

$$Y^i(t) = 1 \iff N^i(t) > 0.$$



The random variables $\{Y^i(t)\}$ are modelled as a discrete time 2-state Markov chain with transition probability matrix

$$P^i(t) = \begin{pmatrix} 1 - \psi^i(t) & \psi^i(t) \\ 1 - \phi^i(t) & \phi^i(t) \end{pmatrix}$$

where

$$\phi^i(t) = \mathbb{P}(Y^i(t) = 1 | Y^i(t-1) = 1),$$

$$\psi^i(t) = \mathbb{P}(Y^i(t) = 1 | Y^i(t-1) = 0).$$

5.2.1 Seasonal changepoints

For each node in the network seasonality is then learnt as a changepoint problem acting on the time-dependent transition matrix $P^i(t)$ over some finite season S that repeats every S observations. For ease of notation for the remainder of this section the dependency on node i will be dropped, but it is assumed for the LANL data that seasonality is learnt separately for each node. There may be some computer networks where it would be appropriate to assume that seasonality is shared across all nodes in the network as in Chapters 2, 3 and 4 and the models described can easily extend to that scenario.

Seasonal variability in behaviour is captured by dividing the seasonal period into segments and fitting separate homogeneous Markov chains for transition behaviour within each seasonal segment. In discrete time the vector of ℓ ordered seasonal changepoints $s_{1:\ell}$ take values from a discrete set of points $\{1, 2, \dots, S - 1\}$ and let $s_0 = 0$ and $s_{\ell+1} = S$. Within each seasonal segment the model assumes a fixed but unknown parameter pair (ψ_j, ϕ_j) for $j = 0, \dots, \ell$.

At $t = T$, where T can be assumed to be the training period, define the set B_j to be the time intervals for which the data observed on $\{0, \dots, T\}$ share the common transition probability matrix for the seasonal period $\{s_j, \dots, s_{j+1} - 1\}$ so that

$$B_j = \{t : s_j \leq r(t, S) < s_{j+1}\}$$

where $r(t, S) \in \{s_j, s_j + 1, \dots, s_{j+1} - 1\}$ s.t. $\exists c \in \mathbb{N}_0$ s.t. $t = cS + r(t, S)$.

The likelihood of the observed data conditional on $Y(0)$ is then

$$\mathcal{L}(Y(\{1, \dots, T\}) | s_{1:\ell}, \ell, \psi_{0:\ell}, \phi_{0:\ell}) = \prod_{j=0}^{\ell} \psi_j^{n_{01}^j} (1 - \psi_j)^{n_{00}^j} \phi_j^{n_{11}^j} (1 - \phi_j)^{n_{10}^j},$$

where n_{ab}^j records the number of transitions from a to b in $Y(B_j)$, for $a, b \in \{0, 1\}$.

The presence of a changepoint at each of the time points can be viewed as independent Bernoulli($\tilde{\nu}$) trials so that the prior for the number and location of the changepoints is

$$p(s_{1:\ell}, \ell) = \tilde{\nu}^{\ell} (1 - \tilde{\nu})^{S-\ell-1}.$$

Conjugate beta distribution priors,

$$\phi_j, \psi_j \stackrel{\text{iid}}{\sim} \text{Beta}(\tilde{\alpha}, \tilde{\beta}),$$

allow the unknown transition probability matrices to be integrated out to give the posterior distribution for the seasonal changepoints as in (1.4). This posterior is known up to proportionality

through

$$\gamma(s_{1:\ell}|Y(\{0, \dots, T\})) = \tilde{\nu}^\ell (1 - \tilde{\nu})^{S-\ell-1} \prod_{j=0}^{\ell} \frac{B(\tilde{\alpha} + n_{01}^j, \tilde{\beta} + n_{00}^j)}{B(\tilde{\alpha}, \tilde{\beta})} \frac{B(\tilde{\alpha} + n_{11}^j, \tilde{\beta} + n_{10}^j)}{B(\tilde{\alpha}, \tilde{\beta})},$$

where

$$B(\tilde{\alpha}, \tilde{\beta}) = \frac{\Gamma(\tilde{\alpha})\Gamma(\tilde{\beta})}{\Gamma(\tilde{\alpha} + \tilde{\beta})}.$$

RJMCMC sampling can then be performed directly on the number and location of the changepoints according to the posterior $\pi(s_{1:\ell}|Y(\{0, \dots, T\}))$. The moves employed are similar to Section 3.1 where the transitions are: (a) ‘birth’ of a new changepoint uniformly chosen over $\{1, 2, \dots, S-1\} \setminus \{s_1, \dots, s_\ell\}$; (b) ‘death’ of a randomly chosen changepoint; and (c) ‘move’ a randomly chosen changepoint uniformly over a discrete time window around the changepoint.

The MAP of a sample of size m obtained via RJMCMC will be used as an estimate for the seasonal changepoints for each node. First the MAP number of changepoints is obtained,

$$\ell^* = \operatorname{argmax}_{\ell} p(\ell|Y(\{0, \dots, T\}))$$

and then conditional on $\ell = \ell^*$, the MAP changepoints are obtained,

$$s_{1:\ell^*}^* = \operatorname{argmax}_{s_{1:\ell^*}^{(i)}} \pi(s_{1:\ell^*}^{(i)}|Y(\{0, \dots, T\})) \mathbb{I}_{\ell^*}(\ell^*).$$

The seasonal changepoints can initially be learnt on a batch of training data, and then updated periodically.

Within each seasonal period between the MAP changepoints, the Markov chain probabilities have a known Beta posterior distribution at time T ,

$$\begin{aligned} [\psi_j|Y(\{0, \dots, T\})] &\equiv \text{Beta}(\tilde{\alpha} + n_{01}^j, \tilde{\beta} + n_{00}^j), \\ [\phi_j|Y(\{0, \dots, T\})] &\equiv \text{Beta}(\tilde{\alpha} + n_{11}^j, \tilde{\beta} + n_{10}^j). \end{aligned} \tag{5.1}$$

These posterior distributions can be obtained given a training data set and will serve as the priors

for the unknown seasonal Markov chain parameters when doing inference in real time on a test data set. Further, when performing inference, these distributions can be updated as each new data point is observed over time.

Note that as an alternative, $Y(\{0, \dots, T\})$ could be modelled as a Bernoulli process and then seasonality would be learnt as a changepoint problem acting on a single parameter $\phi(t)$ where,

$$\begin{aligned}\phi(t) &= \mathbb{P}(Y(t) = 1) \\ 1 - \phi(t) &= \mathbb{P}(Y(t) = 0).\end{aligned}$$

5.2.2 Application to LANL data

For analysis of the LANL data, the counts are aggregated into 10 minute intervals following Neil et al. (2013a). The seasonal period for this data set is chosen to be two weeks, which corresponds to $S = 2016$: LANL have a work scheme whereby employees can take every other Friday off as can be seen by Figure 5.5, which shows the activity status of a typical node from the network with respect to a fortnightly season, and so choosing a seasonal period of a fortnight will capture this behaviour. The prior expectation of the number of changepoints according to the model is $(S - 1)\tilde{\nu}$, and therefore setting $\tilde{\nu} = 0.009$ will correspond *a priori* to an average of 18 changepoints per seasonal period. This value is chosen based on the hypothesis that it is reasonable to expect on average 2 changepoints per work day and then none on the weekends. The parameters $\tilde{\alpha}$ and $\tilde{\beta}$ for the conjugate Beta priors on the transition probabilities will be set to a default value of 1, as very small values for $\tilde{\alpha}$ and $\tilde{\beta}$ can lead to overfitting of changepoints.

Figure 5.5 shows the MAP seasonal changepoints and posterior expectation of the transition probabilities for one individual learned from the 4 weeks of training data.

As a comparison of the Markov and Bernoulli process models, the marginal likelihood of the ten days of test data for each node in the network was calculated based on the seasonal changepoints obtained under each choice of model. 88% of nodes had a larger marginal like-

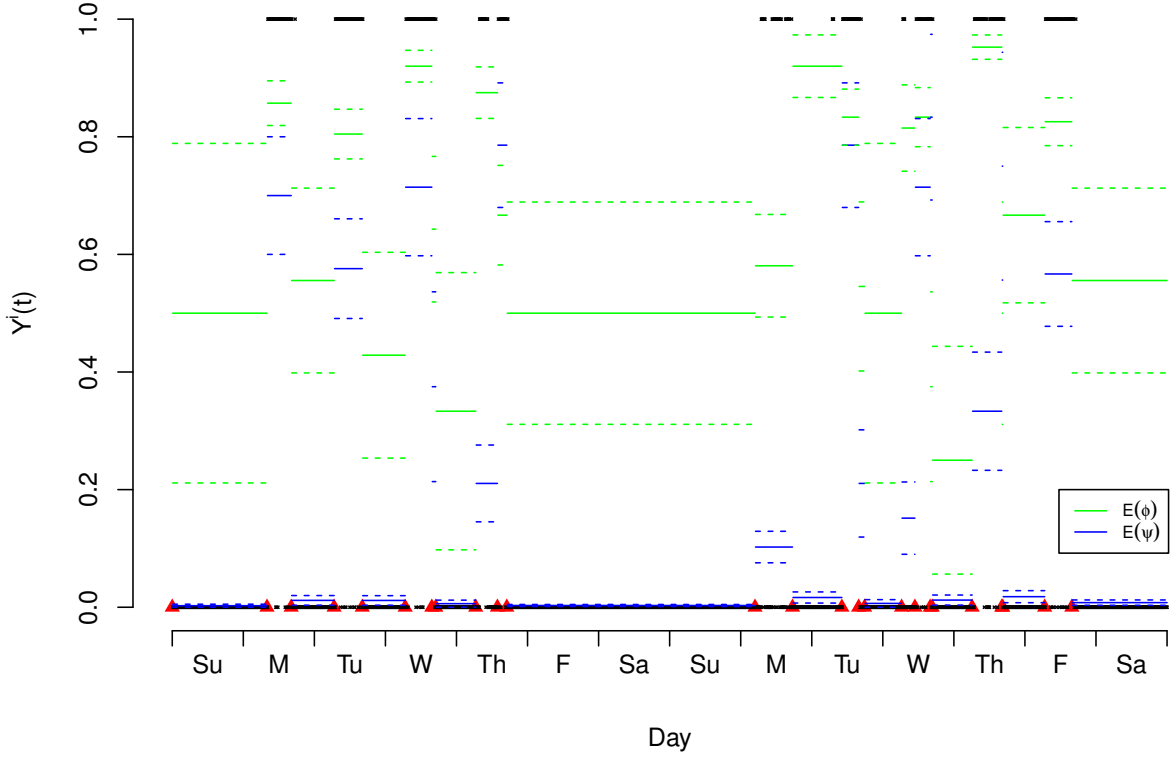


Figure 5.5: Activity status of a node from the LANL computer network data with the MAP seasonal changepoints indicated by triangles. The dotted lines are the posterior expectations (5.1): $\mathbb{E}_\pi(\phi) \pm \text{s.d.}_\pi(\phi)$ in green and $\mathbb{E}_\pi(\psi) \pm \text{s.d.}_\pi(\psi)$ in blue.

likelihood under the Markov model and the average log Bayes factor across all nodes in favour of the Markov model is 258.81, whereas the average log Bayes factor for the nodes that favour the Bernoulli process is 10.34. This should be expected, as the likelihood of a user making a connection in the next 10 minutes is generally affected by whether the user is currently active. Since the Bernoulli process is the simpler model, this choice also tends to lead to a higher number of changepoints being fitted. The nodes for which the Bernoulli process had a higher marginal likelihood were most often quiet nodes with not much observed activity.

5.3 Edge model

Conditional on the source node being active in any particular interval t , the activity status for a particular edge is assumed to be independent of the time of day. Even when only looking at the time periods when the node is active, connections along the edge are still very bursty, as

illustrated in Figure 5.3. To capture this burstiness an additional 2–state discrete Markov chain for the on-off activity status of the edge is applied, but restricted to those time points where the source node is active.

5.3.1 Markov chain for activity status

Similarly let $N^{ij}(t)$ be the number of connections from node i to node j at the t^{th} time point and let $Y^{ij}(t) \in \{0, 1\}$ be the indicator variable for whether the edge is active at t :

$$Y^{ij}(t) = 1 \iff N^{ij}(t) > 0.$$

Fitting a 2–state Markov chain model on the activity status gives a similar transition probability matrix for each edge:

$$P^{ij} = \begin{pmatrix} 1 - \psi^{ij} & \psi^{ij} \\ 1 - \phi^{ij} & \phi^{ij} \end{pmatrix}.$$

Conjugate beta distribution priors,

$$\phi^{ij}, \psi^{ij} \stackrel{\text{iid}}{\sim} \text{Beta}(\bar{\alpha}, \bar{\beta}),$$

allow the unknown transition probability parameters to be integrated out. The Beta posterior distributions for the unknown parameters ψ^{ij} and ϕ^{ij} at time T are,

$$\begin{aligned} [\psi^{ij} | Y^{ij}(\{0, \dots, T\})] &\equiv \text{Beta}(\bar{\alpha} + n_{01}^{ij}, \bar{\beta} + n_{00}^{ij}), \\ [\phi^{ij} | Y^{ij}(\{0, \dots, T\})] &\equiv \text{Beta}(\bar{\alpha} + n_{11}^{ij}, \bar{\beta} + n_{10}^{ij}). \end{aligned} \tag{5.2}$$

As on the node level these will act as a prior when doing inference in real time and can be updated with each new observed data point over time.

Alternatively, modelling could begin directly at the edge level rather than the node level, and seasonal changepoints would be learnt separately for each edge. This would be necessary if seasonality variability is different for the edges with the same source node. However, this isn't

preferable if data on the edges are very sparse.

Application to LANL Data

As with the prior parameters for the transition probabilities for the nodes, $\bar{\alpha}$ and $\bar{\beta}$ will be set to be 1. Note that decreasing $\bar{\alpha}$ will increase how anomalous “new” edges are perceived, such as edges that are quiet for the whole training period and start making connections in the test period. However, p-values obtained from observing extreme counts (see Section 5.4.1) tend to be smaller than those obtained from observing new edges even when $\bar{\alpha}$ is very small. An extension would be to incorporate a separate model for the formation of new edges, see Section 6.2.

As in Section 5.2.2, a comparison was made between the Markov model and the Bernoulli process for each edge in the test period. As the data are very sparse for the edges only 46% of the edges perform better under the more complex Markov model. However, the average log Bayes factor across all the edges in favour of the Markov model is 61.43 and across the edges in favour of the Bernoulli process is 1.47; this shows that for the more interesting edges, the Markov model performs much better than for the Bernoulli process.

5.3.2 Negative Binomial distribution for counts

For those periods in which an edge (i, j) is active then communication counts $\{N^{ij}(t)\}$ will be considered to be independent realisations from a fixed probability model. The form of this distribution will correspond to the typical distribution of the counts between that pair of nodes for that edge under regular conditions when that region of the network is not infected. A good justification is given in Lambert and Liu (2006) for using a Negative Binomial distribution as it does not suffer from the equidispersion property of the Poisson distribution.

If $Y^{ij}(t) = 1$, then it is assumed that

$$N^{ij}(t) - 1 \sim \text{NegBin}(r^{ij}, \theta^{ij}).$$

The conjugate prior for θ^{ij} is the Beta distribution, $\theta^{ij} \sim \text{Beta}(\alpha, \beta)$, and the conditional posterior distribution at t given r^{ij} is

$$\left[\theta^{ij} \middle| \sum_{u=0}^t N^{ij}(u) = n^{ij}, \sum_{u=0}^t Y^{ij}(u) = a^{ij}, r^{ij} \right] \equiv \text{Beta}(\alpha + n^{ij} - a^{ij}, \beta + r^{ij} a^{ij}). \quad (5.3)$$

Full Bayesian inference would require marginalising over r^{ij} using numerical integration or Monte Carlo simulation. Instead, here r^{ij} is determined by maximising the joint posterior of r^{ij} and θ^{ij} using the counts obtained from the training data, via a root finding algorithm.

Application to LANL Data

The priors for the Beta distribution are chosen to be 1, which is equivalent to assuming a uniform distribution on θ^{ij} . Figure 5.6 shows the empirical count distributions for four of the edges within the network during the training period, as well as their fitted Negative Binomial distribution with r^{ij} and θ^{ij} set to the MAP values.

An obvious alternative choice for modelling the count data would be a Poisson distribution with a conjugate Gamma prior on the rate parameters λ^{ij} , $\lambda^{ij} \sim \text{Gamma}(\alpha, \beta)$. The posterior distribution for λ^{ij} at time t is

$$\left[\lambda^{ij} \middle| \sum_{u=0}^t N^{ij}(u) = n_{ij}, \sum_{u=0}^t Y^{ij}(u) = a_{ij} \right] \equiv \text{Gamma}(\alpha + n^{ij} - a^{ij}, \beta + a^{ij}).$$

Figure 5.6 also shows the fitted Poisson distributions with λ_{ij} set as the posterior mean, where $\alpha = 0.01$ and $\beta = 0.01$.

It is clear to see that the Negative Binomial distribution fits the counts much better than the Poisson distribution in these examples; it is more flexible as there are two parameters controlling the shape and location, and the heavier tails capture the over dispersion that is apparent in some of the edges.

For computational ease the counts $N^{ij}(t)$ for active periods are assumed to be independent and identically distributed, after the autocorrelation function lag 1 was estimated to give an

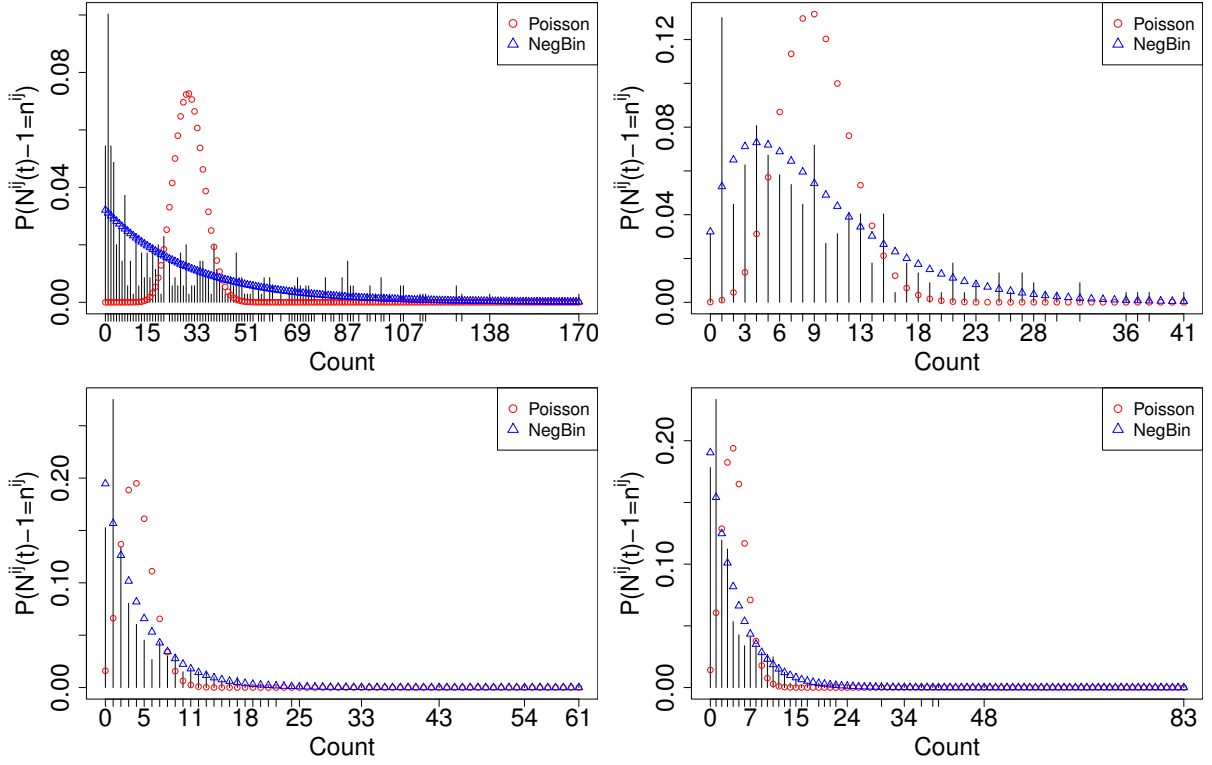


Figure 5.6: The empirical count distribution with the corresponding fitted Negative Binomial distribution in blue triangles and Poisson distribution in red circles.

indication of how much dependence is exhibited between non-zero counts in the data. Figure 5.7 shows the empirical distribution of the autocorrelation for the edges in the network during the training period that were active for more than 50 time points. There are a collection of edges for which the counts are constant throughout the training period, meaning the variance of $N^{ij}(t)$ is zero, and these were also excluded. Approximately 80% of the edges have a lag 1 autocorrelation estimated to be between $[-.30, .30]$. Some autocorrelation is to be expected, but this result shows that in general it is fairly low. The edges for which the counts exhibit stronger dependence are mostly edges that originate from or connect to servers or very other central nodes in the network and are almost always active. Therefore in practice p-values obtained from these edges should be treated with caution. Ideally separate models should be fitted to this set of edges and this is a subject of further work, see Section 6.2.

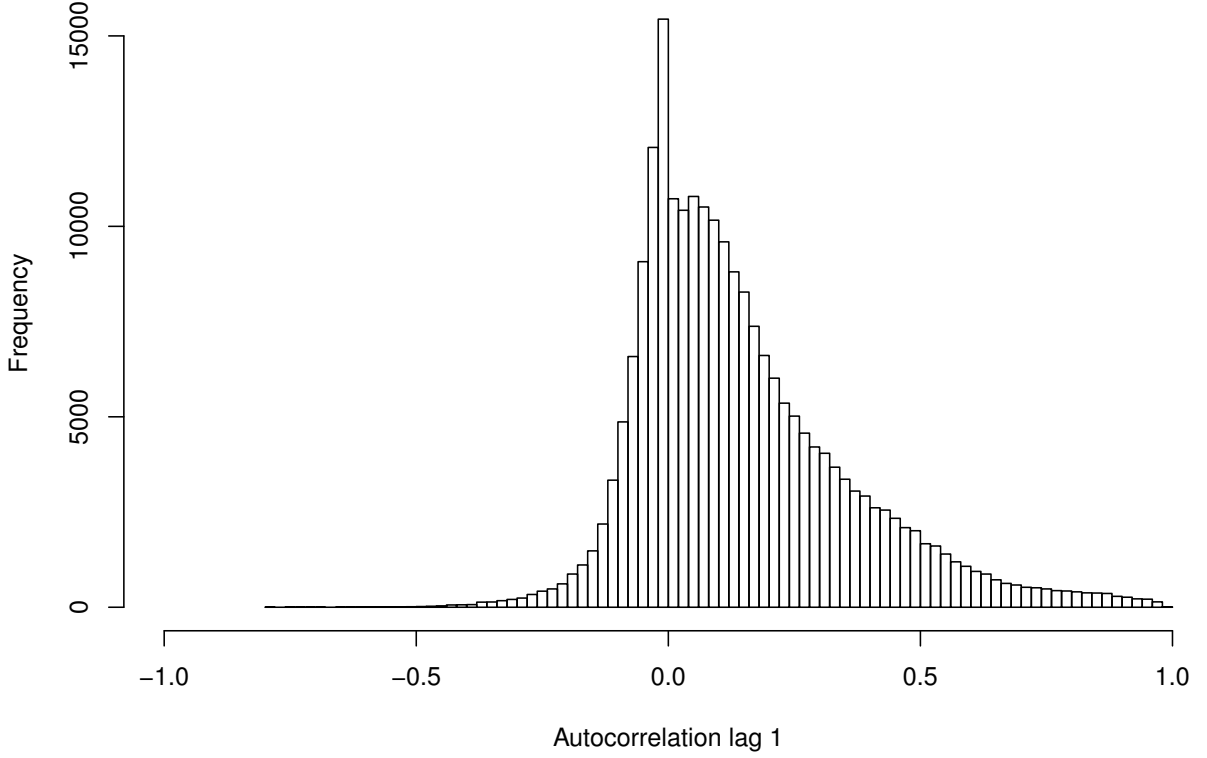


Figure 5.7: Empirical distribution of the autocorrelation function lag 1.

5.4 Monitoring

At time t , let $G_t = (V_t, E_t)$ be the current graph consisting of all the communicating nodes, V_t , and edges, E_t , active during the recent time window $\{t - w, \dots, t - 1\}$, where w is the length of the window. The window length w can be chosen to suit the concerns of the analyst, but realistically should be small relative to the history of the graph. For each edge $(i, j) \in E_t$, a predictive p-value, p_t^{ij} , is obtained from the counts observed over $\{t - w, \dots, t - 1\}$ signifying how far the edge has deviated from its usual behaviour, see Section 5.4.1. An anomaly subgraph of the network is then formed of all edges that have a p-value below a threshold and further inference is conducted on the anomaly subgraph, see Section 5.4.2. This can be considered as the second stage of analysis similar to that discussed in Heard et al. (2010).

The above steps are then repeated at subsequent time points, producing a time series of anomaly graphs over a sliding window of size w . The use of sliding windows to monitor network graphs over time is detailed in Neil et al. (2013a).

5.4.1 Predictive distributions

At any point in time we want to test whether the observed communications between edges, $\{N^{ij}(t)\}$ are a draw from their respective probability distributions or if the relationship has changed to some degree that is significant, i.e a draw from some other distribution.

In particular at time t , let $\bar{N}^{ij}(t, w) = \sum_{u=1}^w N^{ij}(t - u)$. The quantity of interest is the posterior predictive p-value, $p_t^{ij} = \mathbb{P}(\bar{N}^{ij}(t, w) \geq k)$, where k is the sum of the observed communication counts in the past w consecutive intervals. Note that we are only interested in one-sided p-values as when intruders move around the network they will always increase communication counts between edges.

Let $[\bar{Y}^i(t, w) = \sum_{u=1}^w Y^i(t - u)] \in \{0, \dots, w\}$ be the number of subintervals in which a node is active in w windows. The pmf $\mathbb{P}(\bar{Y}^i(t, w) = y_i)$ can be obtained by summing over all possible activity status w -tuple combinations. For example if $Y^i(t - 4) = 0$ then

$$\mathbb{P}(\bar{Y}^i(t, 3) = 2) = (1 - \psi_j^i)\psi_j^i\phi_j^i + \psi_j^i(1 - \phi_j^i) + \psi_j^i\phi_j^i(1 - \phi_j^i),$$

given t is in the j th seasonal period.

Similarly let $[\bar{Y}^{ij}(t, w)|\bar{Y}^i(t, w) = y^i] \in \{0, \dots, y^i\}$ be the number of subintervals in which an edge is active given the source node is active for y^i windows. Then the pmf of $\bar{Y}^{ij}(t, w)$ can be obtained in a similar way. The unknown transition probabilities are integrated out using their posterior distributions using the data observed so far given in (5.2) and (5.1). Note that when integrating out ϕ^i and ψ^i the pmf will depend on what seasonal interval the window of time falls in, and further there will be times when the window $\{t - w, \dots, t - 1\}$ may cross over a seasonal interval.

For $k = 0$ the p-value is trivially one, and for $k > 0$ the p-value is calculated as,

$$p_t^{ij} = \sum_{y^i=1}^w \sum_{y^{ij}=1}^{y^i} [\mathbb{P}(\bar{N}^{ij}(t, w) \geq k | \bar{Y}^{ij}(t, w) = y^{ij}) \mathbb{P}(\bar{Y}^{ij}(t, w) = y^{ij} | \bar{Y}^i(t, w) = y^i) \mathbb{P}(\bar{Y}^i(t, w) = y^i)] . \quad (5.4)$$

If $k \leq y^{ij}$ then trivially $\mathbb{P}(\bar{N}^{ij}(t, w) \geq k | \bar{Y}^{ij}(t, w) = y^{ij}) = 1$. Otherwise let $\bar{N}_-^{ij}(t, w) = \bar{N}^{ij}(t, w) - \bar{Y}^{ij}(t, w)$ and $k' = k - y^{ij}$, where $k' > 0$. Then under the Negative Binomial model for the counts,

$$[\bar{N}_-^{ij}(t, w) | \bar{Y}^{ij}(t, w) = y^{ij}, r^{ij}, \theta^{ij}] \sim \text{NegBin}(r^{ij} y^{ij}, \theta^{ij}),$$

where θ^{ij} is unknown but has a distribution that will have been learnt using the data observed so far given by (5.3). The compounded distribution of a NegBin whose rate parameter is a random variable following a Beta distribution is a Beta negative binomial distribution (BNB). Hence

$$[\bar{N}_-^{ij}(t, w) | \bar{Y}^{ij}(t, w) = y^{ij}, r^{ij}] \sim \text{BNB}(y^{ij} r^{ij}, \alpha + n^{ij} - a^{ij}, \beta + r^{ij} a^{ij}),$$

and the predictive p-values have closed form solutions.

5.4.2 Anomaly graph

Classifying an anomaly for single edge p-values that fall below a threshold would result in a lot of false alarms. Setting a threshold extremely low to limit the number of false alarms obtained could result in missed true anomalous events. As intruders in the network move around the network to either escalate their privileges or to obtain access to computers with valuable information the idea with the anomaly graph is to combine p-values to look for anomalous substructures within the network indicating this type of movement.

For a p-value threshold $T_p \in (0, 1)$, an anomaly subgraph of the network, $S_t = (V_t^S, E_t^S)$, is formed from edges that have a p-value below the threshold,

$$E_t^S = \{(i, j) \in E_t | p_t^{ij} < T_p\},$$

$$V_t^S = \{i \in V_t | \exists j \neq i \in V_t \text{ s.t. } (i, j) \in E_t^S \text{ or } (j, i) \in E_t^S\}.$$

In practice, the threshold T_p can be chosen so that, over a training period, the average or the median of the number of nodes in the anomaly graph $\{|V_t^S|\}$ does not exceed a desired number.

A graph is said to be connected if every vertex is reachable from every other vertex, (Kolaczyk, 2000). A weakly connected component of a graph is a maximally connected subgraph with the property that if all the directed edges were replaced with undirected edges, the resulting subgraph would be connected. Each of the weakly connected components of S_t can be considered as a potentially anomalous attack. Let $A_{k,t} = (V_{k,t}, E_{k,t})$ denote the k^{th} weakly connected component.

Under the null hypothesis of normal behaviour the p-values $p_{A_{k,t}} = \{p_t^{ij} | (i, j) \in E_{k,t}\}$ obtained along the edges for each weakly connected component $A_{k,t}$ are independent and approximately uniformly distributed on $(0, T_p)$. Note that they are only approximately uniformly distributed due to the discreteness of the counts, see Heard and Turcotte (2013b) for details on continuity corrections for discrete p-values.

The p-values can then be combined to give an overall anomaly score for each connected component. Fisher's method is commonly used to combine p-values obtained from independent tests into a single test statistic. Using this method, for each connected component a test statistic is obtained using the formula,

$$X_{k,t}^2 = -2 \sum_{(i,j) \in E_{k,t}} \log \frac{p_t^{ij}}{T_p}.$$

When the null hypotheses, which give rise to the p-values, are correct then $X_{k,t} \sim \chi_{2|E_{k,t}|}^2$. Outlying behaviour will correspond to large values of $X_{k,t}$ and a single p-value,

$$p_{k,t}^F = \mathbb{P}(X_{k,t} > x_{k,t}) \quad (5.5)$$

can be obtained from the chi-squared distribution to give a measure of surprise for each connected component in the anomaly graph.

An alternative method that can be used to combine the p-values is Stouffer's Z-score method. Letting $Z_t^{ij} = \Phi^{-1}(1 - p_t^{ij}/T_p)$, under the null hypothesis the realised values $\{Z_t^{ij}\}$ independently follow a standard normal distribution. Then for each connected component a Z-score, $Z_{k,t}$, can

be obtained for the overall level of surprise, where in the simplest case

$$Z_{k,t} = \frac{\sum_{(i,j) \in E_{k,t}} Z_t^{ij}}{\sqrt{|E_{k,t}|}}.$$

The Z-score follows a standard normal distribution under the null hypothesis.

The advantage of using Stouffer's Z-score method to Fisher's method is that it becomes straightforward to weight the p-values. In a computer network there are nodes that are inherently more connected and hence would naturally appear in the anomaly graph more often. At the other extreme there are nodes that are rarely active, and hence edges in the anomaly graph originating from these nodes are more rare. The p-values don't take into account the level of overall connectivity in the network and weighting them when combining the p-values would add an extra level of information. Given weights for each edge in the anomaly graph, w_t^{ij} , more generally define

$$Z_{k,t} = \frac{\sum_{(i,j) \in E_{k,t}} w_t^{ij} Z_t^{ij}}{\sqrt{\sum_{(i,j) \in E_{k,t}} w_t^{ij^2}}}.$$

Outlying behaviour will correspond to large values of $Z_{k,t}$ and a single p-value

$$p_{k,t}^Z = \mathbb{P}(Z_{k,t} > z_{k,t}) \quad (5.6)$$

can be obtained from the standard normal.

Ultimately how to set the weights would be dependent on the computer network, and could be guided by an analyst. The weights on each edge could depend on how often the source node and the receiving node appear in the anomaly graph over time. A reasonable weighting mechanism could be

$$w_t^{ij} = \left(1 - \frac{\sum_{t'=0}^t \mathbb{I}((i,k) \in S_{t'}, k \neq i)}{t+2}\right) \times \left(1 - \frac{\sum_{t'=0}^t \mathbb{I}((k,j) \in S_{t'}, k \neq j)}{t+2}\right), \quad (5.7)$$

corresponding to a linearly increasing interaction in the weights according to how often the

source and destination nodes appear in the anomaly graph.

As a final note, the p-values given by (5.4) originating from the same node depend on each other through $\mathbb{P}(\bar{Y}^i(t, w) = y^i)$. So when combining the p-values consider conditioning on the activity status of the source node so that, $p_t^{ij} = \mathbb{P}(\bar{N}^{ij}(t, w) \geq k | \bar{Y}^i(t, w) = y^i)$. Using Stouffer's Z-score method to combine the p-values and weighting the edges according to how often the source node appears in the anomaly graph retains some level of surprise from the source node. For the LANL data most of the surprise from the p-values comes from the counts along the edges, so conditioning on the node activity level when combining the p-values would have little effect.

5.5 Results

For the LANL computer network data, the counts are aggregated into 10 minute time periods. The parameters of the model given in (5.1), (5.2) and (5.3) were learnt from the four weeks of training data and then updated recursively throughout the test period. Sections 5.2.2, 5.3.1 and 5.3.2 discussed the choices of prior parameters for the model.

The window size used to monitor the network is chosen to be 30 minutes, which corresponds to $w = 3$. Analysts at LANL suggested this to be an appropriate choice for a window size, as it approximately corresponds to the time required to see intruders moving around the network. Neil et al. (2013a), who also analyse LANL NetFlow data, also scan the network over a 30 minute window.

Many edges in the LANL network data exhibit occasional extreme counts as part of normal behaviour. This can be seen in Figure 5.4, which shows the maximum count observed in a 10 minute window versus the median count, for a sample of edges. Standard exponential family models will not capture these heavy tails well; further work related to modelling the edges better is discussed in Section 6.2. As a result, the p-values obtained in this analysis are highly unstable in the tails of the distribution and this would severely distort the Fisher score when combining p-values in the anomaly graph, since they will be far from uniformly distributed

under normal behaviour. Hence for the analysis, the empirical cdf of all p-values obtained less than the threshold T_p up until the current time are used to recalibrate the present p-values. This empirical cdf is evaluated at the current p-values to provide stabilised p-values that are used for obtaining the Fisher scores or Stouffer's Z-scores of the components in the anomaly graph as described in Section 5.4.2.

The threshold, T_p , for the anomaly subgraph was chosen so that the median of the number of nodes in the anomaly graphs in the training period was 10. The median was considered rather than the average as the empirical distribution of the number of nodes in the anomaly graphs has very heavy tails.

Figure 5.8 shows the p-values less than 0.05 from the components of the anomaly graphs over the 10 day test period using both the Fisher method, (5.5), and the Z-score Stouffer's method, (5.6), where the p-values are weighted according to (5.7). The points in this plot which are close together in time are often the same event being repeatedly detected in sequential time windows, in the sense that the majority of the actors of the connected components are the same. For example, the points in Figure 5.8 joined together by black lines would all constitute one event, detected over several windows.

The most anomalous component detected on the first day contains four of the known anomalous hosts (labelled 277, 1115, 3584 and 4299). Figure 5.9 shows the anomaly graph of this event at the window in which it was first detected, just 1 hour after the initial infection was known to have occurred. Analysis of the attack suggest that the infected computers did not become active until an hour after the time of the initial infection. The same event is then detected in two subsequent time windows thereafter and then again half an hour later. Some of the four anomalous nodes plus two additional anomalous nodes (130 and 10580) appear in much larger connected components in the later detections as shown in Figure 5.10. The strength of the detection of this anomaly is such that a threshold could be set to detect this genuine anomaly such that there are no false alarms when using Fisher's method of combining p-values, and just one false alarm using Stouffer's Z-score method. Although Stouffer's Z-score method flags a component as being more anomalous later on in the test period, on the first and second day, the

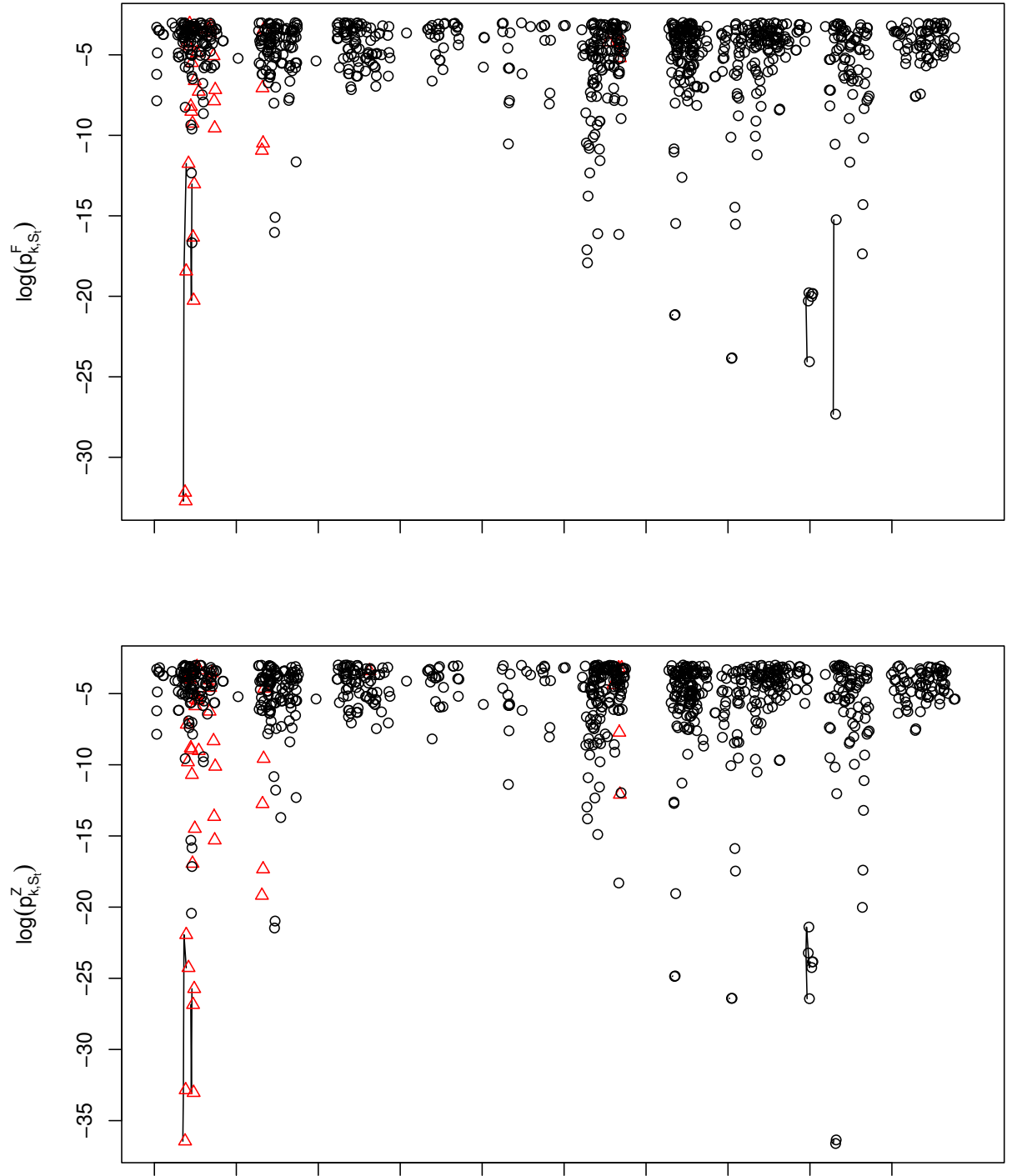


Figure 5.8: Anomaly scores for components of the anomaly graphs over the 10 day test period using Fisher's method(top) and Stouffer's Z-score method(bottom). The triangles indicate components that contain known infected nodes.

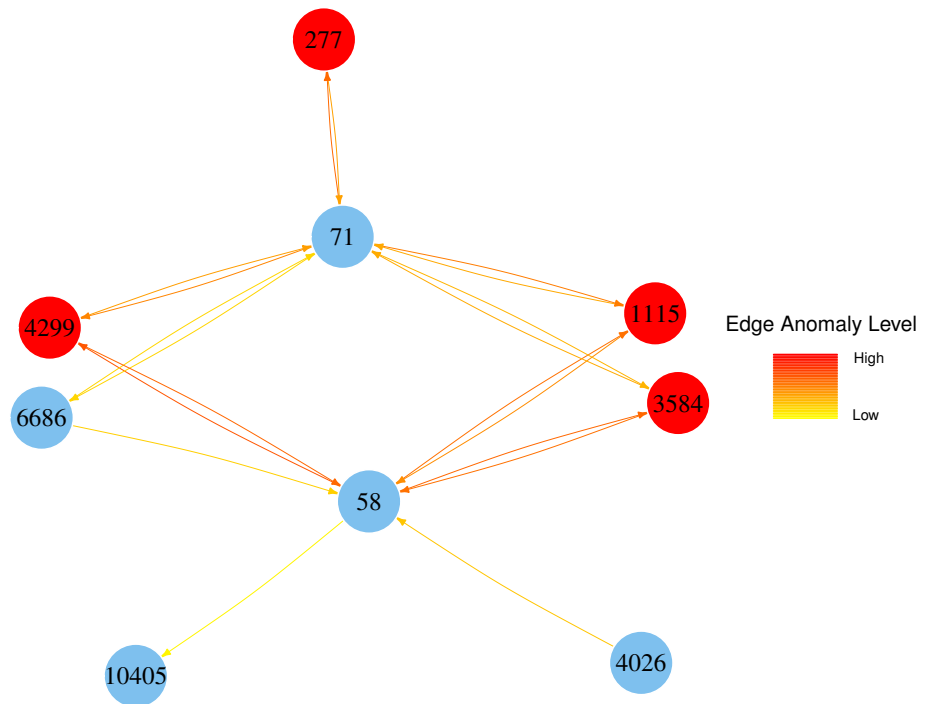


Figure 5.9: Heat map of the anomalous component that was first detected, red nodes are the known infected hosts.

anomalous event has lower p-values after the initial detection relative to other components than under Fisher's method.

The central nodes (71 and 58) in Figure 5.9 are core servers in the LANL network and thus connected to a large portion of the network, as evidenced by Figure 5.11 which shows the indegree of the core servers over the training period. The four anomalous hosts were connecting to these core servers simultaneously for data mining purposes. Combining anomalous activity along edges to look for coordinated anomalous movement in the network allows the anomaly to be detected even though it is centered around busy hubs of the network. When thresholding is based on the single p-values of edges there are many false alarms. For example, a single edge p-value threshold that would detect just the lowest edge p-value in Figure 5.9 would raise 57 false alarms.

As a comparison, the methodology of Neil et al. (2013a), which aims to detect 3-paths of anomalous behaviour, was also applied to this data set. Detection of the initial intrusion into

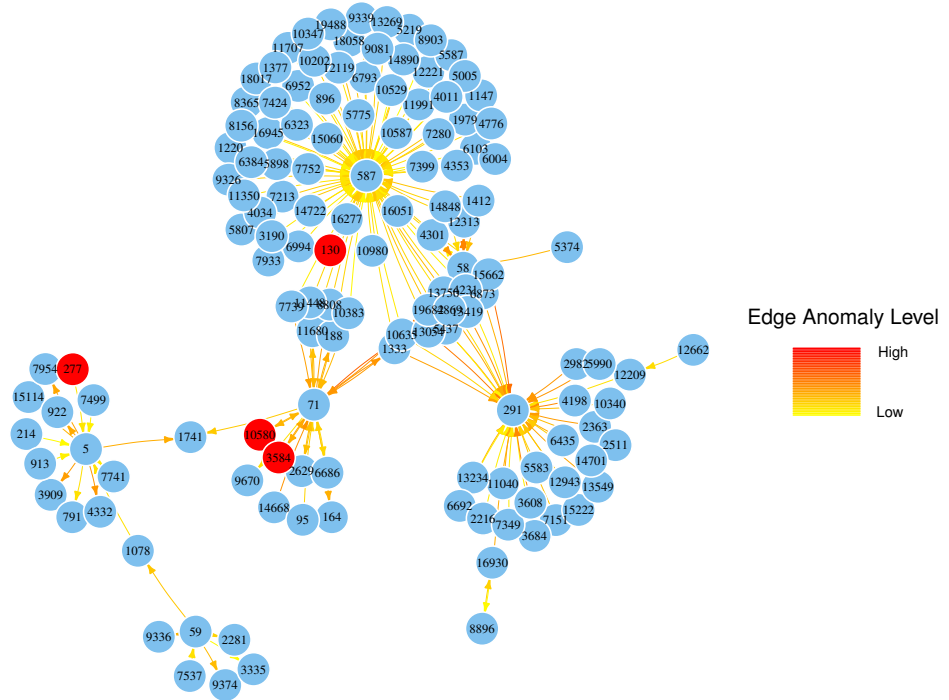


Figure 5.10: Heat map of component containing anomalous hosts in a detection in subsequent time windows, red nodes are the known infected hosts.

the network on the first day of the test data required an unacceptably high false alarm rate. In contrast, the methodology of Neil et al. (2013) was instead able to detect a genuine anomaly on the seventh day of the test data, where traversal was occurring and new edges were being formed. This traversal behaviour was not detected by the methods presented here, but the earlier detection of infection from this proposed method would have had considerable value.

The anomaly graph analysis was done offline. Calculating the p-values along each edge for all nodes in the network over the 10 days of test data took 53 minutes based on C++ code run on an Poweredge T300 workstation with single quad-core Intel Xeon X5470 CPU, clocked at 3.33 GHz. This is approximately 3 seconds run time for evaluating the predictive p-values across the network for every 10 minute window. As mentioned in Section 5.1, there are approximately 40,000 observed edges in each 10 minute window. The methodology developed in this Chapter would therefore be quick enough to be implemented in real time on extremely large networks.

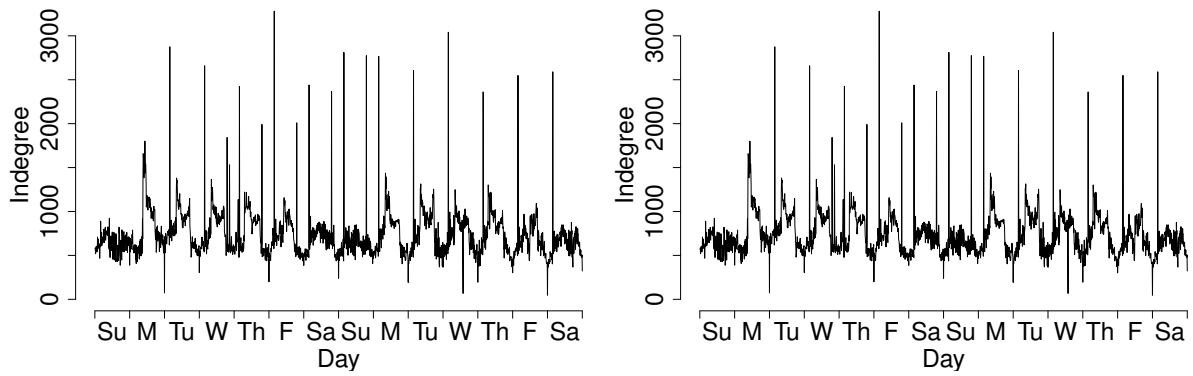


Figure 5.11: Indegree of the two core servers(Left: 71, Right: 58) for each 10 minute window over the training period

Chapter 6

Conclusion and Future Work

6.1 Anomaly detection for telecommunication networks

From a continuous time perspective, a Bayesian changepoint model was introduced for anomaly detection where communications across the network were modelled independently for each entity as inhomogeneous Poisson processes. As some networks exhibit seasonal behaviour that is common across the network, the model was adapted to include a set of seasonal changepoints common to all entities in the network. This assumption could be relaxed so that each entity in the network has their own set of seasonal changepoints. Algorithms were given for both the fixed time frame and sequential inference to sample from the changepoint posterior for each entity in the network. The algorithms were demonstrated on the VAST dataset with results comparable to those in Heard et al. (2010) and Ye et al. (2008).

The continuous time changepoint Bayesian literature is somewhat lacking useful definitions for characterising the estimated locations of changepoints; construction of such a definition is complicated by the number of changepoints being unknown. The function of interest specified in Section 2.3 provided a novel measure for continuous time changepoint detection; for discrete time models Fearnhead and Liu (2007), Chopin (2007) and Fearnhead and Clifford (2003) look at the distribution of the time until the most recent changepoint over discrete time points. The function of interest in Section 2.3 also provides the user with tunable parameters, such as the

significance level desired at which the changepoint (anomaly) will be flagged and the period of time that would be perceived as anomalous after the changepoint has been detected.

In a fixed time frame a Gibbs sampler was presented to sample from the joint posterior of the seasonal changepoints and the individual changepoints for entities within the network. An approximation to the full model was presented that speeds up computation, is fully parallelisable and extends nicely to the sequential case.

The contribution to continuous time changepoint detection is the SMC algorithm described in Section 4.2. Although the algorithm is primarily for changepoint detection in continuous time models it could also be used for discrete time models where the observations arrive in discrete time. However, for discrete time scenarios where the data are processed one observation at a time the algorithm may work less well than other standard particle filtering methods. All of the examples presented in Section 4.4 assumed the data followed an inhomogeneous Poisson process, but the algorithm has also been successfully applied to both an AR process of changing model order and a piecewise constant regression model using datasets given in Punska et al. (1999) and Donoho and Johnstone (1994) respectively, obtaining similar results to those papers.

The SMC algorithm presented was for conjugate models or other models where the parameters can be marginalised out. In Section 4.4.3, where the algorithm was demonstrated on the coal mining disaster data, importance sampling from a conjugate model was performed to give a weighted sample from the non-conjugate model used in Del Moral et al. (2006) to get estimates for the intensity of the process. In general, for more complex non-conjugate models there may not be a suitable conjugate model to exploit in this way. Therefore an important extension would be to generalise the algorithm to non-conjugate models and a brief outline of the required extension is given in Appendix B. However, for the motivating application of real time network anomaly detection considered in this thesis, the extra computational complexity implied by this extension could be problematic.

When using the approximate model for seasonality, where the seasonal changepoints were assumed to be independent of the individual changepoints (Sections 4.4.4 and 3.2.1), data from the whole 10 days of the VAST data set were used to learn the seasonality patterns in the net-

work. To learn seasonality sequentially, at every incremental time step t_n the seasonal changepoints and intensity multipliers could be updated given the new observed data across the network. However, updating the seasonal changepoints sequentially increases the computational complexity of Algorithm 5 as the simplicity of calculating the incremental weights, (4.10), is lost. This is because the target distribution in the numerator of (4.10) would have to be recalculated for the vector of changepoints for each of the particles due to the revised set of seasonal changepoints. The algorithm applied to a network dataset would also now only be partially parallelisable. Learning seasonality sequentially may not be necessary for most cases as usually there will be some training or historical data from which the seasonality could be learnt beforehand; then going forward, the seasonal behaviour would not be expected to vary greatly over time. As a compromise, one could consider learning the seasonality from data currently available and then updating this periodically rather than at each incremental time step.

In both the fixed time and real time inference settings an algorithm was also presented for adaptive sample size allocation. A novel error criterion was introduced using information theoretic criteria for assessing the Monte Carlo convergence of a sample based on a single run. A loss function was then specified to combine the errors of the different samplers into an overall performance criteria. This enables adaptive sample size choices to be made in parallel when sampling from a collection of distributions. The method was applied in a fixed time frame for a collection of processes and as an intermediary step for the SMC algorithm presented in Section 4.2. Results showed that using a variable sample size strategy significantly outperforms an equal sample size strategy.

Finally from a network analysis perspective, rather than treating each edge or node as a univariate counting process of call times, each edge or node could be considered to have an associated multivariate stochastic process describing behavioural aspects of that node or edge. For telecommunications data this could be the location of where the call was made or the duration of the call, both of which were available for the VAST data set. A richer class of models incorporating the various attributes available could enhance anomaly detection capabilities.

6.2 Computer network anomaly detection

For anomaly detection in computer networks, an overall anomaly detection system was developed for detecting intruders on internal computer networks. Discrete time hierarchical Bayesian models were used to model node and then edge behaviour. As in the continuous time frame, seasonality was incorporated through a changepoint model acting on a Markov chain transition probability matrix modelling the on/off behaviour of nodes in the network. A second stage of analysis was developed whereby predictive p-values obtained from the edge models were combined to look for locally anomalous components in the network that are indicative of an intruder moving around the network. The system was demonstrated on Los Alamos National Laboratory's large internal network and a genuine attack was detected soon after the attack was known to have occurred.

The method developed assumed that the discrete count models for all edges was a Negative Binomial distribution. However; highly variable behaviour is observed along different edges across the network. For example, the servers or central machines in the network are usually constantly active and the distribution of counts observed along edges connecting to these central machines is very different from that observed along edges that are more user driven. A more complete modelling effort is needed, where edges are categorised according to the different types of observed behaviour and different models applied to the various categories to better capture the variety of behaviour observed. Input from analysts could guide how to categorise edges. Better modelling of count data observed on edges would result in more stable p-values so that recalibrating of p-values (Section 5.5) would not be needed.

Figure 6.1 shows the empirical count distributions for six edges in the network, demonstrating some different types of behaviours that are frequently observed. For edges where extreme counts are commonly observed, such as in the bottom right of Figure 6.1, the integer part of the logarithmic counts could be modelled rather than the raw count. Other edges, especially those connecting to more central parts of the network, often have a constant baseline single count observed and then occasional higher counts, such as the middle and bottom left of Figure

6.1. For these types of edges a hurdle model could be considered to capture the initial spikes in the counts. More refined edge models will increase detection sensitivity and decrease false alarms especially in capturing anomalous behaviour along central parts of the network that are extremely noisy.

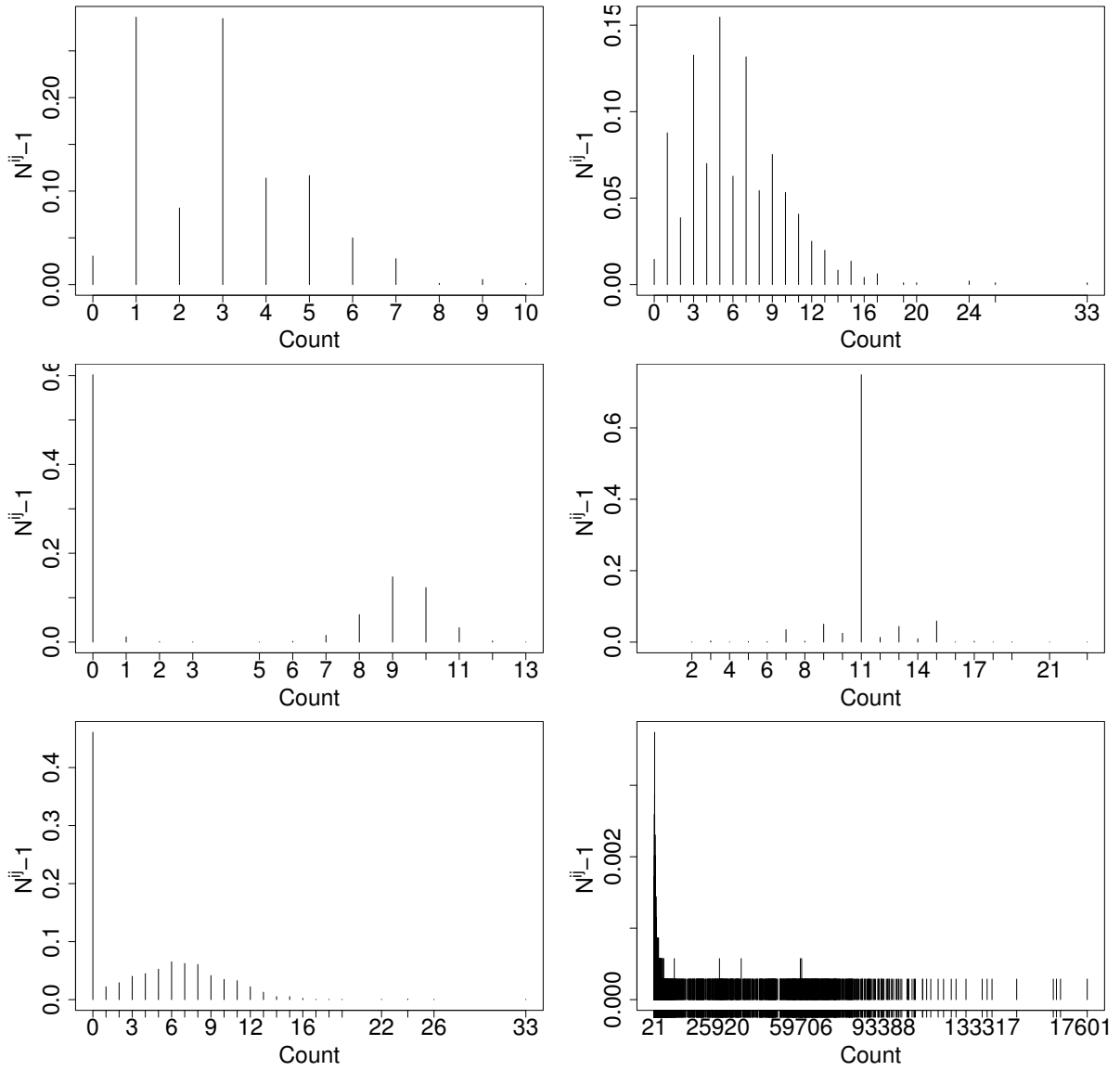


Figure 6.1: The empirical count distributions of edges in the LANL network during the training period. The bottom two graphs are the count distributions from edges that either connect to or from a central server in the network.

It was assumed here that each node had its own seasonality; by categorising different types of behaviour, the model could be simplified so that similar types of nodes share seasonality.

Central nodes that are always active and will not have seasonal behaviour and learning seasonality on nodes of this type could be avoided.

When an intruder infects a node on a computer network they do not typically have information on which nodes that node usually connects to. As a result, when an intruder traverses the network they often make new edges. As mentioned in Section 5.3.1, a separate model could be applied for the formation of new edges in the network. This could be based on the source node's historical propensity to make new edges and the receiving node's propensity to receive new edges. This would better capture the probability of observing new edges in the network, increasing the statistical power of capturing an intruder traversing the network.

As in the continuous time frame, further work would also be required to incorporate the diverse levels of information that exist across multiple data sets capturing network behaviour. Each edge and node would then have an associated multivariate time-series describing different behavioural aspects. For NetFlow records, like the data analysed in Chapter 5, there are many more fields available describing each connection, such as bytes and packets sent and received and the ports the connection was both initiated and received on. For a specific further example, LANL have developed a host-agent capable of collecting various attributes of the node over time, such as open files and services and processes in use (Neil et al., 2013b). Richer models incorporating these attributes will enhance detection capability as attackers tend to make changes to these attributes that would not be reflected in the NetFlow count data modelled here.

Finally, with different models capturing various network behaviours there will be a need for reliable methods for combining different levels of surprise from multiple sources to give an overall anomaly score and provide a complete picture of the type of anomaly detected that can then be passed on to an analyst.

References

- S. Axelsson. Intrusion detection systems: A survey and taxonomy. Technical report, Technical report, 2000.
- D. Barry and J. Hartigan. Product Partition Models for Change Point Problems. *The Annals of Statistics*, 20:260–279, 1992.
- D. Barry and J. Hartigan. A Bayesian Analysis for Change Point Problems. *Journal of the American Statistical Association*, 88:309–319, 1993.
- L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Chapman and Hall/CRC, 1984.
- S. P. Brooks. Markov Chain Monte Carlo Method and Its Application. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 47:69–100, 1998.
- M. H. Cahill, D. Lambert, J. C. Pinheiro, and D. X. Sun. Detecting fraud in the real world. *Computing Reviews*, 45(7):447, 2004.
- B. P. Carlin, A. E. Gelfand, and A. F. M. Smith. Heirarchical Bayesian Analysis of Changepoint Problems. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 41:389–405, 1992.
- V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3):15:1–15:58, 2009.

- N. Chopin. Dynamic detection of change points in long time series. *Annals of the Institute of Statistical Mathematics*, 59:349–366, 2007.
- C. Cortes, D. Pregibon, and C. Volinsky. Computational Methods for Dynamic Graphs. *Journal of Computational and Graphical Statistics*, 12:950–970, 2003.
- P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 68:411–436, 2006.
- D. G. T. Denison, C. C. Holmes, B. K. Mallick, and A. F. M. Smith. *Bayesian Methods for Nonlinear Classification and Regression*. Wiley, West Sussex, 2002.
- D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81:425–455, 1994.
- A. Doucet and A. M. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. In *Handbook of Nonlinear Filtering*. Oxford University Press, 2009.
- A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York, 2001.
- P. Fearnhead. Exact and efficient Bayesian inference for multiple changepoint problems. *Statistics and Computing*, 16:203–213, 2006.
- P. Fearnhead and P. Clifford. On-line Inference for hidden Markov models via particle filters. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 65:887–899, 2003.
- P. Fearnhead and Z. Liu. On-line inference for multiple changepoint problems. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 69:589–605, 2007.
- V. S. Frost and B. Melamed. Traffic modeling for telecommunications networks. *Communications Magazine, IEEE*, 32(3):70–81, 1994.

- N. Gans, G. Koole, and A. Mandelbaum. Telephone call centers: Tutorial, review, and research prospects. *Manufacturing & Service Operations Management*, 5(2):79–141, 2003.
- W. R. Gilks and C. Berzuini. Following a moving target - Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 63:127–146, 2001.
- P. Grassberger. Entropy Estimates from Insufficient Samplings. *ArXiv Physics e-prints*, 2003.
- P. J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82:711–732, 1995.
- D. M. Hawkins. Fitting multiple change-point models to data. *Computational Statistics and Data Analysis*, 37:323–341, 2001.
- N. A. Heard and M. Turcotte. Monte Carlo convergence of rival samplers. *Submitted to Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 2013a.
- N. A. Heard and M. J. M. Turcotte. Monitoring a device in a communication network. In *Statistical Aspects of Network Cyber Security*. Imperial College Press, 2013b.
- N. A. Heard, D. J. Weston, K. Platanioti, and D. J. Hand. Bayesian Anomaly Detection Methods for Social Networks. *Annals of Applied Statistics*, 4:645–662, 2010.
- D. V. Hinkley. Inference about the change-point in a sequence of random variables. *Biometrika*, 57:1–17, 1970.
- T. Idé and H. Kashima. Eigenspace-based anomaly detection in computer systems. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04, pages 440–449, 2004.
- R. E. Kass and A. E. Raftery. Bayes Factors. *Journal of the American Statistical Association*, 90:773–795, 1995.

- E. D. Kolaczyk. *Statistical Analysis of Network Data: Methods and Models*. Springer, New York, 2000.
- D. Lambert and C. Liu. Adaptive thresholds: Monitoring streams of network counts. *Journal of the American Statistical Association*, 101:78–88, 2006.
- A. Lazarevic, A. Ozgur, L. Ertöz, J. Srivastava, and V. Kumar. A comparative study of anomaly detection schemes in network intrusion detection. In *In Proceedings of the Third SIAM International Conference on Data Mining*, 2003.
- P. A. W. Lewis. Some results on tests for Poisson processes. *Biometrika*, 52(1-2):67–77, 1965.
- J. Lin. Divergence measures based on the Shannon entropy. *Information Theory, IEEE Transactions on*, 37(1):145–151, 1991. ISSN 0018-9448. doi: 10.1109/18.61115.
- J. S. Liu. Metropolized independent sampling with comparisons to rejection sampling and importance sampling. *Statistics and Computing*, 6:113–119, 1996.
- W. A. Massey, G. A. Parker, and W. Whitt. Estimating the parameters of a nonhomogeneous Poisson process with linear rate. *Telecommunication Systems*, 5(2):361–388, 1996.
- L. Mendo and J.M. Hernando. A simple sequential stopping rule for Monte Carlo simulation. *Communications, IEEE Transactions on*, 54(2):231–241, 2006.
- J. C. Meza, S. Campbell, and D. H. Bailey. Mathematical and statistical opportunities in cyber security. *CoRR*, abs/0904.1616, 2009.
- J. Neil, C. Storlie, C. Hash, A. Brugh, and M. Fisk. Scan statistics for the online detection of locally anomalous subgraphs. *Technometrics*, 55(4):403–414, 2013a.
- J. Neil, B. Uphoff, C. Hash, and C. Storlie. Towards improved detection of attackers in computer networks: New edges, fast updating, and host agents. In *Resilient Control Systems (ISRCs), 2013 6th International Symposium on*, pages 218–224. IEEE, 2013b.

- C. C. Noble and D. J. Cook. Graph-based anomaly detection. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 631–636. ACM, 2003.
- Y. Park, C.E. Priebe, and A. Youssef. Anomaly detection in time series of graphs using fusion of graph invariants. *Selected Topics in Signal Processing, IEEE Journal of*, 7(1):67–75, 2013.
- A. Patcha and J. Park. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer Networks*, 51(12):3448–3470, 2007.
- C. E. Priebe, J. M. Conroy, and D. J. Marchette. Scan statistics on Enron graphs. *Computational and Mathematical Organization Theory*, 11:229–247, 2005.
- O. Punska, C. Andrieu, A. Doucet, and W. J. Fitzgerald. Bayesian Segmentation of Piecewise Constant Autoregressive Processes using MCMC methods. Technical report, University of Cambridge, 1999.
- A. E. Raftery and V. E. Akman. Bayesian analysis of a Poisson process with a change-point. *Biometrika*, 73:85–89, 1986.
- C. J. Rhodes and E. M. J. Keefe. Social network topology: a Bayesian approach. *Journal of the Operational Research Society*, 58:1605–1611, 2007.
- C. P. Robert and G. Casella. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.
- S. L. Scott. Detecting network intrusion using a Markov modulated nonhomogeneous Poisson process. *Submitted to the Journal of the American Statistical Association*, 2001.
- J. Sexton, C. Storlie, J. Neil, and A. Kent. Intruder detection based on graph structured hypothesis testing. In *Resilient Control Systems (ISRCS), 2013 6th International Symposium on*, pages 86–91. IEEE, 2013.
- D. B. Skillicorn. Detecting Anomalies in Graphs. In *ISI*, pages 209–216. IEEE, 2007.

- A. F. M. Smith and G. O. Roberts. Bayesian Computation via the Gibbs Sampler and Related Markov Chain Monte Carlo Methods. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 55:3–23, 1993.
- A. Sperotto, G. Schaffrath, R. Sadre, C. Morariu, A. Pras, and B. Stiller. An Overview of IP Flow-Based Intrusion Detection. *Communications Surveys Tutorials, IEEE*, 12(3):343–356, 2010.
- D. A. Stephens. Bayesian Retrospective Multiple-changepoint Identification. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 43:159–178, 1994.
- U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17:395–416, 2007.
- J. Weinberg, L. D. Brown, and J. R. Stroud. Bayesian forecasting of an inhomogeneous Poisson process with applications to call center data. *Journal of the American Statistical Association*, 102(480):1185–1198, 2007.
- N Whiteley, A. M. Johansen, and S. Godsill. Monte Carlo Filtering of Piecewise Deterministic Process. *Journal of Computational and Graphical Statistics*, to appear, 2011.
- Q. Ye, T. Zhu, D. Hu, B. Wu, N. Du, and B. Wang. Cell Phone Mini Challenge Award: Social Network Accuracy - Exploring temporal communication in mobile call graphs. In *Proceedings of IEEE VAST Symposium*, pages 207–208. IEEE, 2008.

Appendix A

Calculating conditional changepoint distributions

A.1 Individual changepoints

The likelihood function given in (2.5) can be rewritten so that

$$\mathcal{L}(y([0, T]) | \tau_{1:k}, \lambda_{0:k}, s_{1:\ell}, \mu_{0:\ell}) = \left(\prod_{j=0}^{\ell} \mu_j^{\sum_{i=0}^k r_{ij}} \right) \left(\prod_{i=0}^k \lambda_i^{\sum_{j=0}^{\ell} r_{ij}} e^{-\lambda_i \sum_{j=0}^{\ell} \mu_j m_{ij}} \right).$$

Given that $\lambda_i \sim \text{Gamma}(\alpha, \beta)$ independently for $0 \leq i \leq k$, then the marginal likelihood

$$\begin{aligned} \mathcal{L}(y([0, T]) | \tau_{1:k}, s_{1:\ell}, \mu_{0:\ell}) &= c_1 \prod_{i=0}^k \int_0^{\infty} \lambda_i^{\sum_{j=0}^{\ell} r_{ij} + \alpha - 1} e^{-\lambda_i (\sum_{j=0}^{\ell} \mu_j m_{ij} + \beta)} d\lambda_i \\ &= c_1 \prod_{i=0}^k \frac{\Gamma(\alpha + \sum_{j=0}^{\ell} r_{ij})}{\left(\beta + \sum_{j=0}^{\ell} \mu_j m_{ij} \right)^{\alpha + \sum_{j=0}^{\ell} r_{ij}}}, \end{aligned}$$

where

$$c_1 = \frac{\beta^{\alpha(k+1)}}{\Gamma(\alpha)^{k+1}} \left(\prod_{j=0}^{\ell} \mu_j^{\sum_{i=0}^k r_{ij}} \right).$$

Given the prior for the changepoints (2.1), (2.8) immediately follows.

A.2 Seasonal changepoints

The likelihood function given in (2.5) can be rewritten so that

$$\mathcal{L}(y([0, T]) | \tau_{1:k}, \lambda_{0:k}, s_{1:\ell}, \mu_{0:\ell}) = \left(\prod_{i=0}^k \lambda_i^{\sum_{j=0}^{\ell} r_{ij}} e^{-\lambda_i m_{i0}} \right) \left(\prod_{j=1}^{\ell} \mu_j^{\sum_{i=0}^k r_{ij}} e^{-\mu_j \sum_{i=0}^k \lambda_i m_{ij}} \right),$$

as $\mu_0 = 1$.

Given that $\mu_j \sim \text{Gamma}(\tilde{\alpha}, \tilde{\beta})$ independently for $1 \leq j \leq \ell$, then the marginal likelihood

$$\begin{aligned} \mathcal{L}(y([0, T]) | \tau_{1:k}, \lambda_{0:k}, s_{1:\ell}, \mu_0) &= c_2 \prod_{j=1}^{\ell} \int_0^{\infty} \mu_j^{\sum_{i=0}^k r_{ij} + \tilde{\alpha} - 1} e^{-\mu_j (\sum_{i=0}^k \lambda_i m_{ij} + \tilde{\beta})} d\mu_j \\ &= c_2 \prod_{j=0}^{\ell} \frac{\Gamma(\tilde{\alpha} + \sum_{i=0}^k r_{ij})}{\left(\tilde{\beta} + \sum_{i=0}^k \lambda_i m_{ij} \right)^{\tilde{\alpha} + \sum_{i=0}^k r_{ij}}}, \end{aligned}$$

where

$$c_2 = \frac{\tilde{\beta}^{\tilde{\alpha}(\ell+1)}}{\Gamma(\tilde{\alpha})^{\ell+1}} \left(\prod_{i=0}^k \lambda_i^{\sum_{j=0}^{\ell} r_{ij}} e^{-\lambda_i m_{i0}} \right).$$

Given the Poisson process prior for the seasonal changepoints,

$$p(s_{1:\ell}, \ell) = \tilde{\nu}^{\ell} e^{-\tilde{\nu} S}, \quad (\text{A.1})$$

(2.9) immediately follows.

Consider a network with L nodes then

$$\begin{aligned} \mathcal{L}(\underline{y}_L | \underline{\tau}_L, \underline{\lambda}_L, s_{1:\ell}, \mu_{0:\ell}) &= \prod_{z=1}^L \prod_{i=0}^{k_z} \prod_{j=0}^{\ell} (\lambda_i^z \mu_j^z)^{r_{ij}^z} e^{-\lambda_i^z \mu_j^z m_{ij}^z} \\ &= \left(\prod_{z=1}^L \prod_{i=0}^{k_z} (\lambda_i^z)^{\sum_{j=0}^{\ell} r_{ij}^z} e^{-\lambda_i^z m_{i0}^z} \right) \\ &\quad \times \left(\prod_{j=0}^{\ell} \mu_j^{\sum_{z=1}^L \sum_{i=0}^{k_z} r_{ij}^z} e^{-\mu_j \sum_{z=1}^L \sum_{i=0}^{k_z} \lambda_i^z m_{ij}^z} \right). \end{aligned}$$

Again, integrating out μ_j gives the marginal likelihood

$$\begin{aligned}\mathcal{L}(\underline{y}_L | \underline{\mathcal{I}}_L, \underline{\lambda}_L, s_{1:\ell}, \mu_0) &= c_2 \prod_{j=1}^{\ell} \int_0^{\infty} \mu_j^{\sum_{z=1}^L \sum_{i=0}^{k_z} r_{ij}^z + \tilde{\alpha} - 1} \mathbf{e}^{-\mu_j (\sum_{z=1}^L \sum_{i=0}^{k_z} \lambda_i^z m_{ij}^z + \tilde{\beta})} d\mu_j \\ &= c_2 \prod_{j=0}^{\ell} \frac{\Gamma(\tilde{\alpha} + \sum_{z=1}^L \sum_{i=0}^{k_z} r_{ij}^z)}{\left(\tilde{\beta} + \sum_{z=1}^L \sum_{i=0}^{k_z} \lambda_i^z m_{ij}^z\right)^{\tilde{\alpha} + \sum_{z=1}^L \sum_{i=0}^{k_z} r_{ij}^z}},\end{aligned}$$

where

$$c_2 = \frac{\tilde{\beta}^{\tilde{\alpha}(\ell+1)}}{\Gamma(\tilde{\alpha})^{\ell+1}} \left(\prod_{z=1}^L \prod_{i=0}^{k_z} \lambda_i^z \sum_{j=0}^{\ell} r_{ij}^z \mathbf{e}^{-\lambda_i^z m_{i0}^z} \right).$$

Given the prior in (A.1) for the seasonal changepoints, (2.13) immediately follows.

Appendix B

Sequential Monte Carlo for non-conjugate changepoint models

When the parameters for the model cannot be marginalised out, the target distribution of interest is given by (1.1). Similarly to Section 4.2, the proposal distributions for the sequence of target distributions $\pi_{[t_0, t_n]}(\tau_{1:k_n}, \theta_{0:k_n})$ could sample the changepoints and parameters for each update interval $(t_{n-1}, t_n]$ indendently from the posterior distributions for previous intervals.

Assume at t_n a sample of changepoints and parameters

$$\{\tau_{1:k_{n-1}}^{(i)}, \theta_{0:k_{n-1}}^{(i)}\}_{i=1}^N$$

has been obtained from $\pi_{[t_0, t_{n-1}]}(\tau_{1:k_{n-1}}, \theta_{0:k_{n-1}})$ and a sample

$$\{\tilde{\tau}_{1:\tilde{k}_n}^{(i)}, \tilde{\theta}_{0:\tilde{k}_n}^{(i)}\}_{i=1}^N$$

has been obtained from $\pi_{(t_{n-1}, t_n]}(\tilde{\tau}_{1:\tilde{k}_n}, \tilde{\theta}_{0:\tilde{k}_n}, \tilde{k}_n | y((t_{n-1}^*, t_n]))$.

A problem that arises is the sample drawn from the proposal distribution is over parameterised. When augmenting the vectors of changepoints as in (4.8) to give an approximate sample from the updated target distribution, there is an extra redundant parameter between $(\tau_{k_{n-1}}, \tilde{\tau}_0)$. The parameter pair $(\theta_{k_{n-1}}, \tilde{\theta}_0)$ needs to be replaced with a combined single parameter θ_n^* to

cover the interval $(\tau_{k_{n-1}}, \tilde{\tau}_1]$.

Let $\theta_n^* = s_1(\theta_{k_{n-1}}, \tilde{\theta}_0)$ be a suitably chosen function to combine the model parameters. As the marginal distribution of θ_n^* is unlikely to have an analytical solution, a joint change of variables is introduced

$$(\theta_n^*, u_{n-1}) = s(\theta_{k_{n-1}}, \tilde{\theta}_0) = (s_1(\theta_{k_{n-1}}, \tilde{\theta}_0), s_2(\theta_{k_{n-1}}, \tilde{\theta}_0)),$$

where s_2 is some other transformation of $(\theta_{k_{n-1}}, \tilde{\theta}_0)$, so that we have a one to one mapping $(\theta_{k_{n-1}}, \tilde{\theta}_0) \mapsto (\theta_n^*, u_{n-1})$. Let $|\mathbf{J}_n|$ be the determinant of the Jacobian for the transformation s .

The proposal distribution after the change of variables $\bar{q}_{[t_0, t_n]}(\tau_{1:k_n}, \theta_{0:k_n}, u_{1:n-1})$ then generates suitable parameters, θ_j^* , for each of the combined regions but also the nuisance parameters $u_{1:n-1}$:

$$\begin{aligned} \bar{q}_{[t_0, t_n]}(\tau_{1:k_n}, \theta_{0:k_n}, u_{1:n-1}) &= \bar{q}_{[t_0, t_{n-1}]}(\tau_{1:k_n}, \theta_{0:k_{n-1}}, u_{1:n-2}) \\ &\quad \times \gamma_{(t_{n-1}, t_n]}(\tilde{\tau}_{1:\tilde{k}}, \tilde{\theta}_{0:\tilde{k}}, \tilde{k}, y((t_{n-1}^*, t_n])) |\mathbf{J}_n|, \end{aligned}$$

where $\theta_{0:k_n} = (\theta_0, \dots, \theta_{k_{n-1}-1}, \theta_n^*, \tilde{\theta}_1, \dots, \tilde{\theta}_{\tilde{k}})$.

The solution to this is to extend the target distribution so that

$$\pi_{[t_0, t_n]}(\tau_{1:k_n}, \theta_{0:k_n}, u_{1:n-1}) = \pi_{[t_0, t_n]}(\tau_{1:k_n}, \theta_{0:k_n}) \prod_{j=1}^{n-1} f(u_j | \tau_{1:k_j}, \theta_{1:k_j})$$

where f can be any density with the correct support for u_j .

As the true target $\pi_{[t_0, t_n]}(\tau_{1:k_n}, \theta_{0:k_n})$ is a marginal of $\pi_{[t_0, t_n]}(\tau_{1:k_n}, \theta_{0:k_n}, u_{1:n-1})$, standard IS estimates can still be used to give an approximation for the distribution and its normalizing constant.

The importance weights given in (4.9) are now expressed as

$$\begin{aligned} w_n(\tau_{1:k_n}^{(i)}, \theta_{0:k_n}^{(i)}, u_{1:n-1}^{(i)}) &= \frac{\gamma_{[t_0, t_n]}(\tau_{1:k_n}^{(i)}, \theta_{0:k_n}^{(i)}, u_{1:n-1}^{(i)})}{\bar{q}_{[t_0, t_n]}(\tau_{1:k_n}^{(i)}, \theta_{0:k_n}^{(i)}, u_{1:n-1}^{(i)})} \\ &= w_{n-1}(\tau_{1:k_{n-1}}^{(i)}, \theta_{0:k_{n-1}}^{(i)}, u_{1:n-2}^{(i)}) \bar{w}_n(\tau_{1:k_n}^{(i)}, \theta_{0:k_n}^{(i)}, u_{n-1}^{(i)}), \end{aligned}$$

where

$$\bar{w}_n(\tau_{1:k_n}^{(i)}, \theta_{0:k_n}^{(i)}, u_{n-1}^{(i)}) = \frac{\gamma_{[t_0, t_n]}(\tau_{1:k_n}^{(i)}, \theta_{0:k_n}^{(i)}) f(u_{n-1}^{(i)} | \tau_{1:k_n}^{(i)}, \theta_{0:k_n}^{(i)})}{\gamma_{[t_0, t_{n-1}]}(\tau_{1:k_{n-1}}^{(i)}, \theta_{0:k_{n-1}}^{(i)}) \gamma_{(t_{n-1}, t_n]}(\tau_{k_{n-1}+1:k_n}^{(i)}, \tilde{\theta}_{0:k}^{(i)}) |\mathbf{J}_n^{(i)}|}.$$

The parameter transformation s_1 should be chosen so that if $\theta_{k_{n-1}}$ and $\tilde{\theta}_0$ are samples from their own conditional posterior distributions, then $\theta_n^* = s_1(\theta_{k_{n-1}}, \tilde{\theta}_0)$ should be close to a draw from the posterior for the single parameter on the joined segment. The parameter transformation s_2 should have a distribution which ideally is loosely identifiable so that it can guide how to extend the target. For example, for the Poisson process when $t_{n-1}^* = t_{n-1}$ if $u_{n-1} = \lambda_{k_{n-1}} - \tilde{\lambda}_0$ the domain of u_{n-1} is $\mathbb{R}$ and should be near zero if the merger is a good match; setting $f(u_{n-1} | \tau_{1:k_n}, \lambda_{0:k_n}) = \phi(u_{n-1})$, the density of a standard normal, could then be a good choice.