Lívia Maria Dutra

# Exact Bayesian inference for Markov switching Cox processes

Inferência Bayesiana exata para processos de Cox com mudanças Markovianas

Belo Horizonte

2015

UNIVERSIDADE FEDERAL DE MINAS GERAIS
DEPARTAMENTO DE ESTATÍSTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ESTATÍSTICA

Lívia Maria Dutra

# Exact Bayesian inference for Markov switching Cox processes

Inferência Bayesiana exata para processos de
Cox com mudanças Markovianas

Orientador: Flávio Bambirra Gonçalves
Co-orientador: Roger William Câmara Silva

Belo Horizonte
2015

# Agradecimentos

Primeiramente agradeço à Deus por ter colocado a estatística em minha vida. Sou muito grata aos meus pais, Malu e Getúlio, pelo incentivo, carinho, paciência, suporte e muito mais. Aos meus irmãos, Getúlio e Duílio, pela amizade. Ao Leandro pelo amor e, principalmente, paciência. Inclusive, Leandro e minha mãe que já sabem tudo sobre processos de Cox, inferência Bayesiana, cadeias de Markov e etc.

Agradeço a dedicação e exigência dos meus orientadores Flávio e Roger, essenciais para a conclusão deste trabalho. À banca da defesa: Gregório, que foi um dos maiores incentivadores para a escolha desse caminho, e ao Helio Migon, pelas críticas e sugestões.

Agradeço também aos meus queridos amigos da pg-est e, claro, aos amigos não estatísticos também. À minha turma de mestrado, Rachel, Marcela, Maurício, Renata, Luíza e Fabrícia. Agradeço muito a "família" LESTE! Aos meninos, Luís e Bráulio, por sempre resolver aqueles problemas mais básicos referentes ao R, latex e Fedora. Em especial agradeço à Ju, juntas "na alegria e na tristeza", à Larissa, pela preocupação e por sempre acreditar que vai dar certo, à Gabi, pelo incentivo desde a graduação, e à Raquel, mesmo de longe nessa reta final, sempre presente.

# Resumo

A modelagem estatística de dados pontuais é um problema importante e comum em diversas aplicações. Um importante processo pontual, e uma generalização do processo de Poisson, é o processo de Cox, em que a sua função intensidade é também estocástica. O presente trabalho se concentra nos processos de Cox em que sua função intensidade é uma cadeia de Markov em tempo contínuo com espaço de estados finito. Estes processos são referidos como processos de Cox com mudanças Markovianas (PCMM). Algumas propriedades probabilísticas desses processos são investigadas, três novos teoremas enunciados e é desenvolvida uma metodologia Bayesiana para realizar inferência exata, baseada em algoritmos MCMC. O desenvolvimento de uma metodologia exata é facilitado, uma vez que a função de verossimilhança é tratável. São apresentados estudos simulados a fim de investigar a eficiência da metodologia para estimação da função intensidade dos PCMM's e dos parâmetros relacionados a ela. Ao fim, realiza-se uma análise com dados reais.

# Abstract

Statistical modelling of point patterns is an important and common problem in several applications. An important point process, and a generalisation of the Poisson process, is the Cox process, where the intensity function is itself stochastic. We focus on Cox processes in which the intensity function is driven by a finite state space continuous-time Markov chain. We refer to these as Markov switching Cox processes (MSCP). We investigate some probabilistic properties of these processes, three new theorems for these processes are derived and we develop a Bayesian methodology to perform exact inference based on MCMC algorithms. Since the likelihood function is tractable, it facilitates the development of an exact methodology. Simulated studies are presented in order to investigate the efficiency of the methodology on the estimation of MSCP's intensity function and the parameters indexing its law. Finally, an analysis with real data is performed.

# Contents

# Chapter 1

# Introduction

Statistical modelling of point patterns is an important and common problem in several applications. In general, one is interested in modelling the occurrences of a given event of interest in a given region. If this is a region in $\mathbb{R}$, it is commonly interpreted as time, meaning that one is modelling the occurrences of that event over a period of (continuous) time. If $\mathbb{R}^d$ with $d \geq 2$ is considered, it is interpreted as a region itself. Examples of the first case can be found in Daley and Vere-Jones (2003) and of the second case in Møller and Waagepetersen (2003). More general approaches also consider that the pattern is observed in a region of $R^d$ with $d \geq 2$ at multiple instants of time (see Gonçalves and Gamerman). Applications can be found in queueing theory (Brémaud, 1981), finances (Rolski et al., 1999), medical and biology fields (Tong Zhou et al., 1998) and many others.

The most widely used point process is the (homogeneous) Poisson process (PP), in which the number of events in a given region has a Poisson distribution and the point patterns of non-overlapping regions is independent - this property is referred to as stochastic independence. This process has constant rate, that is, the expected number of events in a sub-region is equal for any sub-region with the same size.

A possible generalisation of PP is the non-homogeneous Poisson process (NHPP), which allows the rate to vary across the considered region. Thus, the expected number of events can be different for distinct sub-regions with the same size, that is, one sub-region may be more leaning to the occurrence of events than the others. Further ahead in the text, the PP and NHPP will be properly defined and we only consider point processes on $\mathbb{R}$, therefore, the regions are interpreted as time.

Another generalisation, introduced by Cox (1955), is the doubly stochastic Poisson process, also known as the Cox process (CP), which is a NHPP where its intensity function is itself a stochastic process. While the PP has constant rate and the intensity of the NHPP is a deterministic function, the CP is more flexible to model real data where the intensity function is unknown. The CP has been applied in different contexts, e.g., bursts of rainfall (Smith and Karr, 1983), neuroscience (Amarasingham et al., 2006; Cunningham et al., 2008), finances (Lando, 1998; Dassios and Jang, 2003) and others.

There are many possibilities to describe the dynamics of the intensity function and a widely used class is the log Gaussian Cox process (Møller et al., 1998; Basu and Dassios, 2002). Some other classes can be found in the literature. We focus on Cox processes in which the intensity function is driven by a finite state space continuous-time Markov chain (CTMC) (Taylor and Karlin, 1998; Serfozo, 1972). We shall refer to these as Markov switching Cox processes.

The main objectives of this work are to investigate some probabilistic properties of the proposed process and develop a Bayesian methodology to perform exact inference. The exact inference does not use numerical approximations based on time discretisation schemes, which means that only Monte Carlo error is involved. Thus, the exact approach eliminates the bias introduced by discretisation schemes. Moreover, Monte Carlo error is easy to be controlled. The proposed inference methodology consists of a Monte Carlo Markov Chain (MCMC) algorithm that converges to the exact posterior distribuion of the unknown quantities, which may include: the intensity function, the states of the CTMC and the parameters indexing the law of the CTMC.

The dissertation is organised as follows: Chapter 2 contains a brief review of Markov chain theory and Poisson processes. The definitions, properties and theorems presented provide the required background on Markov chains to understand the work in the dissertation. Some simulations are also presented. If the reader is comfortable with this theory, this chapter may be skipped. Chapter 3 presents the Markov switching Cox processes. Three important results are derived for these processes and some properties when the intensity is a two-state Markov chain are shown together with some simulation examples. In Chapter 4 we propose a Bayesian methodology for exact inference. Chapter 5 contains some simulation studies for different scenarios and an analysis with real data is performed. The data are the number of traffic accidents along a highway. Some conclusions and future work is discussed in Chapter 6.

# Chapter 2

# Markov Chains

In this chapter a brief review of Markov chains theory and Poisson process is presented. Section 2.1 shows discrete-time and continuous-time Markov chains. Some simulation studies for discrete-time Markov chains are also presented with an algorithm to simulate a continuous-time Markov chain. A formal definition and some properties of the Poisson process are presented in Section 2.2, along with an example and two different algorithms to simulate a PP. Also, an important result for PP's simulation is reported. Section 2.3 presents the non-homogeneous Poisson process and the Poisson thinning algorithm.

## 2.1 Markov chain theory

Formally, given a probability space $(\Omega, \mathcal{F}, P)$ a univariate stochastic process is a collection $\{X_t : t \in T\}$, where $X_t$, $t \in T$, is a random variable on $\Omega$ taking values in a set $E \subset \mathbb{R}$. Here T is an index set and denotes the time, which may be discrete or continuous. Also, $E$ is called the state space and may also be discrete or continuous, but we will only consider the discrete case.

There are many sorts of stochastic processes and we shall discuss exclusively those where the future behaviour of the process is independent of its past given its present state. This memoryless property of a stochastic process is known as the Markov property. The formal definition is the following.

**Definition 2.1.** *Let* $0 \leq n_1 < n_2 < \cdots < n_m < n$, $x_{n_1}, x_{n_2}, \ldots, x_{n_m} \in E$ *and* $A \subset E$.

*We say that the process satisfies the Markov property if:*

$$P(X_n \in A | X_{n_1} = x_{n_1}, X_{n_2} = x_{n_2}, \ldots, X_{n_m} = x_{n_m}) = P(X_n \in A | X_{n_m} = x_{n_m}).$$

Let $m < n$ and $X = \{X_n : n \geq 0\}$ be a stochastic process, then the process increment is defined as $X_n - X_m$. The process $X$ is said to have independent increments if its increments for any disjoint time intervals are independent. When the distribution of $X_n - X_m$ depends only on the time interval length the process is said to have stationary increments.

In the following sections we shall present the main definitions and results for Markov chains. A detailed presentation of Markov chains theory may be found in Norris (1998) and Ross (1996).

### 2.1.1 Discrete-time Markov chains

A discrete-time Markov chain (DTMC) is a Markov process whose index set is finite or countable, i.e., $T \subset \{0\} \cup \mathbb{N}$. Each $i \in E$ is called a state and is a possible value for $X_n$. It is common to say that $X_n$ is at state $i$ if $X_n = i$.

Let $\phi = (\phi_i : i \in E)$ be a vector of probabilities such that $\phi_i > 0$ and $\sum_{i \in \mathrm{I}} \phi_i = 1$, then $\phi$ is called the distribution of $X_0$ if $\phi_i = P(X_0 = i)$. Also, $P = (p_{i,j} : i, j \in E)$ is a stochastic matrix if every row $(p_{i,j} : j \in I)$ is a distribution. Later we will call P a transition matrix. A DTMC is completely defined once their transition probabilities and probability distribution of $X_0$ are specified.

**Definition 2.2.** *$\{X_n : n \geq 0\}$ is DTMC with initial distribution $\phi$ and transition matrix $P$ if for $n \geq 0$ and $i, i_1, \ldots, i_{n+1} \in E$:*

*(i) $X_0$ has distribution $\phi$, $P(X_0 = i) = \phi_i$;*

*(ii) $P(X_{n+1} = i_{n+1} | X_0 = i, \cdots, X_n = i_n) = p_{i_n i_{n+1}}$.*

The matrix P gives the one-step transition probabilities of the chain. It can be proved that the n-step transition probabilities are given by the n-th power of P. Thus,

$$P(X_n = j) = (\phi P^n)_j, n \geq 0,$$

where $(\phi P^n)_j$ is the j-th row of the new matrix formed by the product of vector $\phi$ and $P^n$. Also

$$P_i(X_n = j) := P(X_{m+n} = j | X_m = i) = p_{ij}^{(n)}, \quad \forall n \; and \; m \geq 0, \tag{2.1}$$

where $p_{ij}^{(n)}$ is the $(i, j)$ entry of $P^n$. If Equation (2.1) holds for all $m \geq 0$, the DTMC is called homogeneous.

State $j$ is said to be accessible from state $i$ $(i \to j)$ if $p_{ij}^{(n)} > 0$ for some $n \geq 0$. If both $i \to j$ and $j \to i$ occur we say that $i$ communicates with $j$ and write $i \leftrightarrow j$. A chain or transition matrix P is called irreducible when all the states communicate.

A state $i$ is said to be recurrent if it is always possible to return to it, i.e.,

$$P_i(X_n = i \; for \; infinitely \; many \; n) = 1.$$

If in addition the expected return time to state $i$ is finite, then $i$ is said positive recurrent. A recurrent state which fails to have this stronger property is called null recurrent. A state $i$ is transient if

$$P_i(X_n = i \; for \; infinitely \; many \; n) = 0.$$

Every state is either recurrent or transient. If the chain or transition matrix P is irreducible then all states have the same recurrence or transience property. In this case, the chain is said either recurrent or transient.

Understanding the long-time behaviour of a Markov chain comes down to understanding the behaviour of $P^n$ for large n. The invariant distribution for P is $\pi$ if $\pi P = \pi$ and, in this case, it is also called the stationary distribution. This invariant distribution does not always exist and we will show the conditions for the process to have it. State $i$ is aperiodic if and only if the set $\{n \geq 0 : p_{ii}^{(n)} > 0\}$ has no common divisor other than 1.

If a chain is irreducible, aperiodic and positive recurrent it is said to be an ergodic chain. This definition will be used for the following theorems.

**Theorem 2.3** (**Convergence to equilibrium**). *If* $\{X_n : n \geq 0\}$ *is an ergodic chain,* *then:*

*(i) Exist an invariant distribution $\pi$ and it is unique;*

*(ii) $P(X_n = j) \to \pi_j$ as $n \to \infty$ $\forall j$.*

*In particular,*

$p_{ij}^{(n)} \to \pi_j$ *as* $n \to \infty$ $\forall i, j$.

The following ergodic theorem is a version of the strong law of large numbers for DTMC.

**Theorem 2.4 (Ergodic theorem).** *Let $\{X_n : n \geq 0\}$ be an ergodic chain. Then, for any bounded function $f : E \to \mathbb{R}$ we have*

$$P\left(\frac{1}{n}\sum_{k=0}^{n-1} f(X_k) \to \overline{f} \quad as \quad n \to \infty\right) = 1,$$

*where*

$$\overline{f} = E_\pi(f) = \sum_{i \in \mathrm{I}} \pi_i f_i.$$

Furthermore it is possible to estimate an unknown transition matrix P on the basis of observations of the corresponding DTMC. The maximum likelihood estimator (MLE) for $p_{ij}$ is given by

$$\hat{p}_{ij} = \sum_{n=0}^{N-1} \mathbb{I}\{X_n = i, X_{n+1} = j\} / \sum_{n=0}^{N-1} \mathbb{I}\{X_n = i\},$$

where $N$ is the total number of observed transitions.

Estimating this probability is analogous to the estimation of a discrete probability distribution. In the case that the chain is recurrent $\hat{p}_{ij}$ is consistent, i.e., $P(\hat{p}_{ij} \to p_{ij}\ when\ N \to \infty) = 1$.

### 2.1.2 Continuous-time Markov chains

A continuous-time Markov chain (CTMC) is a Markov process which index set is uncountable, i.e., $T \subset \{0\} \cup \mathbb{R}^+$. We shall restrict our attention to processes $\{X_t : t \geq 0\}$ which are right-continuous. The theory follows a very similar pattern as the

DTMC case. When $T$ is uncountable is not possible to define a transition matrix P as in Section 2.1.1. Therefore, we will define a matrix to make it possible to evaluate the behaviour of the chain.

**Definition 2.5.** *Let $E$ be a countable set. A Q-matrix on $E$ is a matrix $Q = (q_{ij} : i, j \in E)$ satisfying the following conditions:*

*(i)* $\sum_{j \in \mathrm{I}} q_{ij} = 0, \ \forall i;$

*(ii)* $q_{ij} > 0, \ \forall i \neq j;$

*(iii)* $0 \leq -q_{ii} < \infty, \ \ \forall i.$

For notation, set $-q_{ii} = q_i$. Note that, $q_i = \sum_{j \in \mathrm{I} \text{ and } j \neq i} q_{ij}$. We shall later interpret $q_i$ as the rate of leaving state $i$ and $q_{ij}$ for $i \neq j$ as the rate of going from $i$ to $j$ when living $i$.

For a fixed t and finite state space $E$, the transition probabilities can be obtained by

$$P(t) = e^{tQ} = \sum_{k=0}^{\infty} \frac{(tQ)^k}{k!}, \tag{2.2}$$

then, $P(t)$ is a stochastic matrix and $P_i(X_t = j) := P(X_t = j | X_0 = i) = p_{ij}(t)$, where $p_{ij}(t)$ is the entry of the matrix $P(t)$ corresponding to the $i$-th row and $j$-th column.

Every path of a CTMC is a right-continuous step-function. The initial distribution $\phi$ (distribution of $X_0$) and Q-matrix Q define a CTMC.

The jump matrix $\Pi = (\pi_{ij} : i, j \in I)$ of Q-matrix Q is defined by

$$\pi_{ij} = \begin{cases} q_{ij}/q_i, & \text{if } \ j \neq i \ \text{ and } \ q_i \neq 0, \\ 0, \ \text{ if } \ j \neq i \ \text{ and } \ q_i = 0. \end{cases}$$

$$\pi_{ii} = \begin{cases} 0, & \text{if } \ q_i \neq 0, \\ 1, & \text{if } \ q_i = 0. \end{cases}$$

Note that $\Pi$ is a stochastic matrix. The process $\{Y_n : n \geq 0\}$ with transition matrix $\Pi$ is said to be a jump chain.

The process dynamics is given by exponential waiting times with parameter $q_i$ and transitions probability according to jump matrix $\Pi$.

The notion of accessibility, communicating states and irreducibility in a CTMC are inherited from the jump chain, thus is the same as in DTMC.

For continuous time, a state $i$ is said to be recurrent if

$$P_i(\{t \geq 0 : X_t = i\} \text{ is unbounded}) = 1.$$

Moreover, when $q_i = 0$ or the expected return time to $i$ is finite, then $i$ is said to be positive recurrent. Otherwise, if the expected return time is infinite, the state is null recurrent. A state $i$ is called transient if

$$P_i(\{t \geq 0 : X_t = i\} \text{ is unbounded}) = 0.$$

Also, if state $i$ is recurrent (transient) for the jump chain $\{Y_n : n \geq 0\}$, then $i$ is recurrent (transient) for $\{X_t : t \geq 0\}$. If the chain is irreducible, then every state is either recurrent or transient.

In CTMC, $\gamma$ is called an invariant distribution if $\gamma Q = 0$, or equivalently, if $\gamma P(s) = \gamma$ for a given $s > 0$. The next theorem establishes when the invariant distribution exists.

**Theorem 2.6** (**Convergence to equilibrium**). *Let $Q$ be an irreducible and positive recurrent $Q$-matrix and $P(t) = e^{tQ}$. Then:*

*(i) Exist an invariant distribution $\gamma$ and it is unique;*

*(ii) $p_{ij}(t) \to \gamma_j$ as $t \to \infty$ $\forall i, j \in E$.*

The ergodic theorem for a CTMC is given below. It does not requires the chain to be aperiodic as in Theorem 2.4.

**Theorem 2.7** (**Ergodic theorem**). *Let $Q$ be an irreducible and positive recurrent $Q$-matrix. Then, for any bounded function $f : E \to \mathbb{R}$ we have*

$$P\left(\frac{1}{t}\int_0^t f(X_s)ds \to \overline{f} \quad as \quad t \to \infty\right) = 1,$$

*where*

$$\overline{f} = E_\gamma(f) = \sum_{i\in \mathrm{I}} \gamma_i f_i.$$

### 2.1.3 Simulations

A numerical example is done to illustrate some estimates of DTMC's. We simulate a DTMC and estimate the invariant distribution $\pi$.

Let P be an irreducible and aperiodic matrix, with finite state space $E$. Let $n_0$ be a number such that $P^{n_0} = P^{n_0+1}$, that is, the chain is stationary from the $n_0$-step onwards. We proceed in two different ways: firstly, we simulate N DTMC's $\{X_n : 0 \leq n \leq n_0\}$ and use the sample of $X_{n_0}$ to estimate $\pi$; secondly, we simulate a DTMC $\{X_n : 0 \leq n \leq n_0 + N\}$ and use the sample $X_{n_0+1}, \cdots, X_{n_0+N}$ to estimate $\pi$.

The first estimator is given by

$$\hat{\pi}_{j(1)} = \sum_{n=1}^{N} \mathbb{I}\{X_{n_0}^{(n)} = j\}/N,$$

where $X_{n_0}^{(n)}$ is the state of the n-th chain at time $n_0$. We have that,

$$E(\hat{\pi}_{j(1)}) = E\left[\frac{\sum_{n=1}^{N} \mathbb{I}\{X_{n_0}^{(n)} = j\}}{N}\right] = \frac{NE\left[\mathbb{I}\{X_{n_0}^{(1)} = j\}\right]}{N}$$

$$= P\left(X_{n_0}^{(1)} = j\right) = \pi_j$$

and

$$Var(\hat{\pi}_{j(1)}) = Var\left[\frac{\sum_{n=1}^{N} \mathbb{I}\{X_{n_0}^{(n)} = j\}}{N}\right] = \frac{NVar\left(\mathbb{I}\{X_{n_0}^{(1)} = j\}\right)}{N^2}$$

$$= \frac{P\left(X_{n_0}^{(1)} = j\right)\left[1 - P\left(X_{n_0}^{(1)} = j\right)\right]}{N} = \frac{\pi_j(1-\pi_j)}{N}.$$

Let $X_n$ be the state at time n. Then, the second estimator is given by

$$\hat{\pi}_{j(2)} = \sum_{n=n_0+1}^{n_0+N} \mathbb{I}\{X_n = j\}/N.$$

Again, we have,

$$E(\hat{\pi}_{j(2)}) = E\left[\frac{\sum_{n=n_0+1}^{n_0+N} \mathbb{I}\{X_n = j\}}{N}\right] = \frac{\sum_{n=n_0+1}^{n_0+N} P(X_n = j)}{N}$$

$$= \frac{NP\left(X_{n_0+1} = j\right)}{N} = \pi_j$$

and

$$Var(\hat{\pi}_{j(2)}) = Var\left[\frac{\sum_{n=n_0+1}^{n_0+N} \mathbb{I}\{X_n = j\}}{N}\right]$$

$$= \frac{\pi_j(1-\pi_j)}{N} + 2\pi_j \sum_{n_0+1 \le n_1 < n_2 \le n_0+N} (P_{jj}^{n_2-n_1} - \pi_j)/N^2.$$

Note that both estimators are unbiased and the variance of the first estimator is smaller.

For the numerical example, we use a matrix P such that $\pi = (0.083, 0.706, 0.211)$ and $n_0 = 10$. Table 2.1 shows the results. Note that $\hat{\pi}_{(1)}$ converges faster than $\hat{\pi}_{(2)}$ due to its smaller variance.

| N | Estimation 1 (by $\hat{\pi}_{(1)}$) | Estimation 2 (by $\hat{\pi}_{(2)}$) |
|---|---|---|
| 100 | (0.050,0.730,0.220) | (0.040,0.780,0.180) |
| 1000 | (0.076,0.679,0.245) | (0.088,0.731,0.181) |
| 5000 | (0.080,0.702,0.218) | (0.078,0.726,0.196) |
| 15000 | (0.083,0.706,0.211) | (0.079,0.704,0.217) |
| 20000 | (0.083,0.706,0.211) | (0.083,0.706,0.211) |

Table 2.1: Invariant distribution estimates for DTMC.

In order to simulate a CTMC, the following algorithm can be used. It is used to simulate a sample path of the CTMC $\{X_t : 0 \le t \le T\}$ that begins at state $a$, i.e., $X_0 = a$.

**Algorithm 1** (Forward sampling)**.**

1. Generate $\tau \sim$ exponential($q_a$). If $\tau \ge T$, stop and $X_t = a$ for all $t \in [0, T]$;

2. if $\tau < T$, choose a new state $c \neq a$ from a discrete probability distribution with probability masses $q_{ac}/q_a$. Go back to step 1 with new beginning state $c$.

## 2.2 Poisson process

A Poisson process $X = \{X(t) : t \geq 0\}$ is a counting process, that is, $X(t)$ represents the total number of events that have occurred in a given region. We consider PP's on the real line and, therefore, the regions are time intervals and $X(t)$ is the total number of events in a time interval $[0, t]$. Note that a PP is a CTMC. The theory described throughout this section is also presented in Norris (1998).

**Definition 2.8.** *A homogeneous Poisson process of rate $\lambda$ (PP($\lambda$)) on $\mathbb{R}^+$ is a CTMC with $X(0) = 0$ and Q-matrix*

$$Q = \begin{bmatrix} -\lambda & \lambda & & \\ & -\lambda & \lambda & \\ & & \ddots & \ddots \\ & & & \end{bmatrix},$$

*with $E = \{0\} \cup \mathbb{N}$.*

We shall denote each event as a jump of the Markov chain or as a point. Such jumps are deterministic and equal to 1. Also, the waiting time until a jump is an exponential random variable with parameter $\lambda$.

A PP may also be defined as follows.

**Definition 2.9.** *The process $X = \{X(t) : t \geq 0\}$ is a PP($\lambda$) if:*

*(i) $X(0) = 0$;*

*(ii) the process has stationary and independent increments;*

*(iii) a) $P(X(t + h) - X(t) = 0) = 1 - \lambda h + o(h)$*

*b) $P(X(t + h) - X(t) = 1) = \lambda h + o(h)$*

*c)* $P(X(t+h) - X(t) \geq 2) = o(h)$

*where* $\lim\limits_{h \to 0} \frac{o(h)}{h} = 0$.

By Definition 2.9 it is possible to find the distribution of $X(t)$. For each t, $X(t)$ has Poisson distribution with parameter $\lambda t$. More generally, the distribution of jumps is Poisson with parameter given by the product of $\lambda$ and the length of the time interval, i.e., $X(t_1) - X(t_0)$ follows Poisson distribution with parameter $\lambda(t_1 - t_0)$.

Some important properties of the PP are following:

1. if $\{X(t) : 0 \leq t \leq T\}$ is a $PP(\lambda)$ in an interval $[0, T]$, then $\{X(t) : t_0 \leq t \leq t_1\}$ is also a $PP(\lambda)$ in $[t_0, t_1]$ for $0 \leq t_0 \leq t_1 \leq T$;

2. if $\{X(t) : t \geq 0\}$ and $\{Y(t) : t \geq 0\}$ are independents PP of rates $\lambda$ and $\mu$ respectively, then the sum of processes, $\{X(t) + Y(t) : t \geq 0\}$, is a $PP(\lambda + \mu)$.

### 2.2.1 Simulations

There are two main ways to simulate a PP and the algorithms are following.

Let $J_0, J_1, \cdots$ be the jump times of a PP $X = \{X(t) : t \geq 0\}$, $S_1, S_2, \cdots$ be the waiting times between jumps and $n$ be the number of jumps in an interval $[0, T]$.

**Algorithm 2.**

```
1. Take $J_0 = S_0 = 0$ and make $n = 1$;
2. generate $S_n \sim$ Exponential($\lambda$);
3. make $J_n = \sum_{i=0}^{n} S_i$;
4. if $J_n \geq T$, stop and take $X(t) = n - 1$. Otherwise, $n = n + 1$ and go back to
   step 2.
```

**Algorithm 3.**

```
1. Generate $n \sim$ Poisson($\lambda T$);
2. simulate the $n$ jump times, $J_1, \cdots, J_n$, independently from a Uniform$(0, T)$.
```

Both algorithms, 2 and 3, are equivalent. Figure 2.1 presents a simulation of a PP(5) in the time interval [0,10].

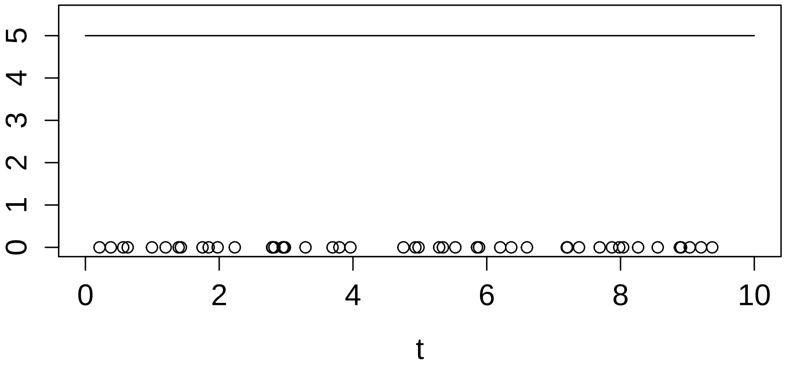

Figure 2.1: Simulation of a PP(5) in the time interval $[0, 10]$. The solid line represents the rate and the points are the jump times.

For a PP($\lambda$) in a time interval $[0, T]$, the likelihood function is

$$L(\lambda; n, J_1, \cdots, J_n) = \frac{(\lambda T)^n e^{-\lambda T}}{n! T^n}$$

and the MLE for $\lambda$ is given by

$$\hat{\lambda} = \frac{n}{T}.$$

We have,

$$E(\hat{\lambda}) = E\left(\frac{N}{T}\right) = \frac{\lambda T}{T} = \lambda \tag{2.3}$$

and

$$Var(\hat{\lambda}) = Var\left(\frac{N}{T}\right) = \frac{\lambda T}{T^2} = \frac{\lambda}{T}. \tag{2.4}$$

To have a numerical example, we simulated one sample of PP(5) for different intervals and estimate the rate $\lambda$. Results are presented in Table 2.2 and shows that the estimates are improved as the time interval increases, this is due to the variance of the estimator, given by equation (2.4).

| Interval | Estimation |
|---|---|
| $[0, 10]$ | 4.50000 |
| $[0, 50]$ | 4.82000 |
| $[0, 100]$ | 4.79000 |
| $[0, 1000]$ | 5.06000 |
| $[0, 10000]$ | 4.97260 |
| $[0, 50000]$ | 5.00318 |
| $[0, 100000]$ | 5.00048 |

Table 2.2: PP rate estimates.

### 2.2.2 Important results

Consider a PP($\lambda$) where in addition to observing a jump, the jump can be classified as belonging to $n$ different categories according to a probability distribution $\pi$. It is possible to construct the processes separately by category. This is referred to as splitting a Poisson process. The algorithm is given below.

**Algorithm 4.**

Let $\pi = (\pi_1, \cdots, \pi_n)$ be a probability distribution and X be a $PP(\lambda)$ in the interval $[0, T]$. Make $i = 1$.

1. Generate a PP($\lambda$): $\tau_1, \tau_2, \cdots, \tau_k$, are the points of the process in the time interval $[0, T]$;
2. generate $Y \sim$ multinomial$(1, \pi)$. If the $j$-th entry of the vector $Y$ is equal to 1, then $\tau_i$ is a realisation of $X^{(j)}$. Make $i = i + 1$ and repeat this step until $i = k$.

**Proposition 2.10.** *Algorithm 4 returns the n PP's separately*

$$X^{(1)} = (\tau_{1_1}, \cdots, \tau_{1_{m_1}}) \sim PP(\pi_1 \lambda),$$

$$\vdots$$

$$X^{(n)} = (\tau_{n_1}, \cdots, \tau_{n_{m_n}}) \sim PP(\pi_n \lambda)$$

*and $X^{(j)}$'s are all independent.*

*Proof.* Let $X(T)$ be the number of points in $[0, T]$. Given $X(T) = k$, events $\tau_1, \tau_2, \cdots, \tau_k$ are independent and uniformly distributed, thus

$$f(\tau_1, \cdots, \tau_k) = f(\tau_1) \cdots f(\tau_k) = \frac{1}{T^k}.$$

Let $Y_1, \cdots, Y_k$ independent random variables that will indicate if $\tau_j$ belongs $X^{(i)}$.

$$Y_j \sim \text{Bernoulli}(\pi_i).$$

First, we will show that $X^{(i)}(T) \sim$ Poisson$(\pi_i \lambda T)$.

$$\begin{aligned}
P(X^{(i)}(T) = n) &= \sum_{k=n}^{\infty} P(X^{(i)}(T) = n | X(T) = k) P(X(T) = k) \\
&= \sum_{k=n}^{\infty} P\left(\sum_{j=1}^{k} Y_j = n\right) P(X(T) = k) \\
&= \sum_{k=n}^{\infty} \binom{k}{n} {\pi_i}^n (1 - \pi_i)^{k-n} \frac{(\lambda T)^k e^{-\lambda T}}{k!} \\
&= \sum_{k=n}^{\infty} \frac{k!}{n!(k-n)!} {\pi_i}^n (1 - \pi_i)^{k-n} \frac{(\lambda T)^k e^{-\lambda T}}{k!} \\
&= \frac{e^{-\lambda T} \pi_i^n}{n!} \sum_{j=0}^{\infty} \frac{(\lambda T)^{j+n} (1 - \pi_i)^j}{j!} \\
&= \frac{e^{-\lambda T} (\pi_i \lambda T)^n}{n!} \sum_{j=0}^{\infty} \frac{(\lambda T - \lambda T \pi_i)^j}{j!} \\
&= \frac{e^{-\lambda T} (\pi_i \lambda T)^n}{n!} e^{\lambda T - \lambda T \pi_i} \\
&= \frac{(\pi_i \lambda T)^n e^{-\pi_i \lambda T}}{n!} \qquad \sim \text{ Poisson}(\pi_i \lambda T).
\end{aligned}$$

Finally, we have to prove that the processes are independent.

$$f_{X(1),\cdots,X(n)}((\tau_{1_1},\cdots,\tau_{1_{m_1}}),\cdots,(\tau_{n_1},\cdots,\tau_{n_{m_n}})) =$$

$$= f_{X(1),\cdots,X(n)|Y_T}\left((\tau_{1_1},\cdots,\tau_{1_{m_1}}),\cdots,(\tau_{n_1},\cdots,\tau_{n_{m_n}})\middle|\sum_{i=1}^{n} m_i\right) P\left(Y_T = \sum_{i=1}^{n} m_i\right)$$

$$= P\left(X(1) = (\tau_{1_1},\cdots,\tau_{1_{m_1}}),\cdots,X(n) = (\tau_{n_1},\cdots,\tau_{n_{m_n}}), X(T_1) = m_1,\cdots,X(T_n) = m_n \middle| Y_T = \sum_{i=1}^{n} m_i\right)$$
$$P\left(Y_T = \sum_{i=1}^{n} m_i\right)$$

$$= f_{X(1),\cdots,X(n)|X(T_1),\cdots,X(T_n),Y_T}\left((\tau_{1_1},\cdots,\tau_{1_{m_1}}),\cdots,(\tau_{n_1},\cdots,\tau_{n_{m_n}})\middle| m_1,\cdots,m_n,\sum_{i=1}^{n} m_i\right)$$
$$P\left(X(T_1) = m_1,\cdots,X(T_n) = m_n \middle| Y_T = \sum_{i=1}^{n} m_i\right) P\left(Y_T = \sum_{i=1}^{n} m_i\right)$$

$$= \frac{1}{T^{\sum_{i=1}^{n} m_i}} \frac{(\sum_{i=1}^{n} m_i)! \prod_{i=1}^{n} \pi_i^{m_i}}{\prod_{i=1}^{n} m_i!} \frac{e^{-\lambda T}(\lambda T)^{\sum_{i=1}^{n} m_i}}{(\sum_{i=1}^{n} m_i)!}$$

$$= \frac{1}{T^{\sum_{i=1}^{n} m_i}} \frac{e^{-\lambda T}(\lambda T)^{\sum_{i=1}^{n} m_i} \prod_{i=1}^{n} \pi_i^{m_i}}{\prod_{i=1}^{n} m_i!}$$

$$= f_{X(1)}(\tau_{1_1},\cdots,\tau_{1_{m_1}})\cdots f_{X(n)}(\tau_{n_1},\cdots,\tau_{n_{m_n}}).$$

□

## 2.3 Non-homogeneous Poisson process

A non-homogeneous Poisson process is a PP where the rate is a deterministic function of time. In NHPP, the rate is called intensity function. This process is more flexible than the PP due to the behaviour of its rate, that changes over time.

**Definition 2.11.** *The process* $\{X(t) : t \geq 0\}$ *is a NHPP with intensity function* $\lambda(t), t \geq 0$ *if:*

*(i)* $X(0) = 0$*;*

*(ii) the process has independent increments;*

*(iii) a)* $P(X(t+h) - X(t) = 0) = 1 - \lambda(t)h + o(h)$

*b)* $P(X(t+h) - X(t) = 1) = \lambda(t)h + o(h)$

*c)* $P(X(t+h) - X(t) \geq 2) = o(h)$

*where* $\lim\limits_{h \to 0} \frac{o(h)}{h} = 0.$

The increment $X(t) - X(s)$ gives the number of points on the interval $(s, t]$ and has Poisson distribution with parameter $\int_s^t \lambda(y)dy$ (Ross, 1996). The distribution of the waiting times is conditioned in the last time that an event has occurred. Let $T_n$ be the waiting time between occurrences of the $(n-1)$-th and $n$-th jump. Then,

$$P(T_n > t | T_{n-1} = s) = P(N(t) - N(s) = 0) = e^{-\int_s^t \lambda(y)dy},$$

thus,

$$F_{T_n|T_{n-1}=s}(t) = P(T_n \leq t | T_{n-1} = s) = 1 - e^{-\int_s^t \lambda(y)dy}.$$

Note that, for a given interval, as the intensity function increases the expected number of points increases and the expected waiting time decreases.

### 2.3.1 The Poisson thinning

To generate a NHPP we use the method called the Poisson thinning introduced by Lewis and Shedler (1979). It is an acceptance-rejection method and the algorithm is given below.

**Algorithm 5.**

Let X a NHPP with rate function $\lambda(t)$, in a fixed interval $[0, T]$.
Take $\lambda_0 = sup_{t \in [0,T]}(\lambda(t))$.

1. Generate a homogeneous PP with rate $\lambda_0$: $\tau_1, \tau_2, \cdots, \tau_k$, are the points of this process in the interval $[0, T]$;

2. Keep each $\tau_i$ with probability $\lambda(\tau_i)/\lambda_0$.

**Proposition 2.12.** *Algorithm 5 returns an exact simulation of a PP($\lambda(t)$).*

*Proof.* We have to show that $(X(t_2)-X(t_1)) \sim \text{Poisson}\left(\int_{t_1}^{t_2}\lambda(s)ds\right)$ for $0 \leq t_1 < t_2 \leq T$ and $(X(t_2)-X(t_1))$ is independent of $(X(t_4)-X(t_3))$ for $0 \leq t_1 < t_2 \leq t_3 < t_4 \leq T$.

Let Y be the number of realisations of the PP($\lambda_0$) in the time interval $[t_1, t_2]$. Given $Y = k$, events $\tau_1, \cdots, \tau_k$ are the jump times in $[t_1, t_2]$ which are independent and uniformly distributed in $[t_1, t_2]$. Thus,

$$f(\tau_1, \cdots, \tau_k) = f(\tau_1)\cdots f(\tau_k) = \frac{1}{(t_2-t_1)^k}.$$

Let $Z_1, \cdots, Z_k$ independent random variables that indicate if each $\tau_i$ is kept or not as a realisation of the PP($\lambda(t)$) in $[t_1, t_2]$. Then

$$Z_i \sim \text{Bernoulli}\left(\int_{t_1}^{t_2}\frac{\lambda(y)}{\lambda_0(t_2-t_1)}dy\right) \quad \text{and} \quad \sum_{i=1}^{k} Z_i \sim \text{binomial}\left(k, \int_{t_1}^{t_2}\frac{\lambda(y)}{\lambda_0(t_2-t_1)}dy\right),$$

where $\int_{t_1}^{t_2}\lambda(y)dy$ is the acceptance rate under all possible values of $\lambda$ in the time interval $[t_1, t_2]$.

Furthermore,

$$\begin{aligned}
P(X(t_2)-X(t_1)=n) &= \sum_{k=n}^{\infty} P(X(t_2)-X(t_1)=n|Y=k)P(Y=k) \\
&= \sum_{k=n}^{\infty} P\left(\sum_{i=1}^{k} Z_i = n\right) P(Y=k) \\
&= \sum_{k=n}^{\infty}\binom{k}{n}\left(\int_{t_1}^{t_2}\frac{\lambda(y)}{\lambda_0(t_2-t_1)}dy\right)^n\left(1-\int_{t_1}^{t_2}\frac{\lambda(y)}{\lambda_0(t_2-t_1)}dy\right)^{k-n} \\
&\qquad \frac{[\lambda_0(t_2-t_1)]^k e^{-\lambda_0(t_2-t_1)}}{k!} \\
&= \frac{e^{-\lambda_0(t_2-t_1)}\left(\int_{t_1}^{t_2}\lambda(y)dy\right)^n}{n!}\sum_{k=n}^{\infty}\frac{\left[\lambda_0(t_2-t_1)-\int_{t_1}^{t_2}\lambda(y)dy)\right]^{k-n}}{(k-n)!} \\
&= \frac{e^{-\lambda_0(t_2-t_1)}\left(\int_{t_1}^{t_2}\lambda(y)dy\right)^n}{n!}\sum_{j=0}^{\infty}\frac{\left[\lambda_0(t_2-t_1)-\int_{t_1}^{t_2}\lambda(y)dy\right]^{j}}{j!}
\end{aligned}$$

$$= \frac{e^{-\lambda_0(t_2-t_1)} \left(\int_{t_1}^{t_2} \lambda(y)dy\right)^n}{n!} exp\left\{\lambda_0(t_2-t_1) - \left(\int_{t_1}^{t_2} \lambda(y)dy\right)\right\}$$

$$= \frac{\left(\int_{t_1}^{t_2} \lambda(y)dy\right)^n e^{-\int_{t_1}^{t_2} \lambda(y)dy}}{n!} \qquad \sim \text{Poisson}\left(\int_{t_1}^{t_2} \lambda(y)dy\right).$$

To simplify the notation, define $(X(t_2) - X(t_1)) = X_1$, $(X(t_4) - X(t_3)) = X_2$, and let $Y_1$ be a $PP(\lambda_0)$ in $[t_1, t_2]$ and $Y_2$ be a $PP(\lambda_0)$ in $[t_3, t_4]$. Then,

$$\begin{aligned}
P(X_1 = k, X_2 = n) &= \sum_{l_1}\sum_{l_2} P(X_1 = k, X_2 = n, Y_1 = l_1, Y_2 = l_2) \\
&= \sum_{l_1}\sum_{l_2} P(Y_1 = l_1, Y_2 = l_2) P(X_1 = k, X_2 = n | Y_1 = l_1, Y_2 = l_2) \\
&= \sum_{l_1}\sum_{l_2} P(Y_1 = l_1)P(Y_2 = l_2)P(X_1 = k | Y_1 = l_1)P(X_2 = n | Y_2 = l_2) \\
&= \sum_{l_1}\sum_{l_2} P(X_1 = k, Y_1 = l_1)P(X_2 = n, Y_2 = l_2) \\
&= \sum_{l_1} P(X_1 = k, Y_1 = l_1) \sum_{l_2} P(X_2 = n, Y_2 = l_2) \\
&= P(X_1 = k)P(X_2 = n).
\end{aligned}$$

□

To illustrate the thinning method we simulate two examples which are shown in Figure 2.2. The results are consistent, as the rate increases (decreases) the number of events increases (decreases).

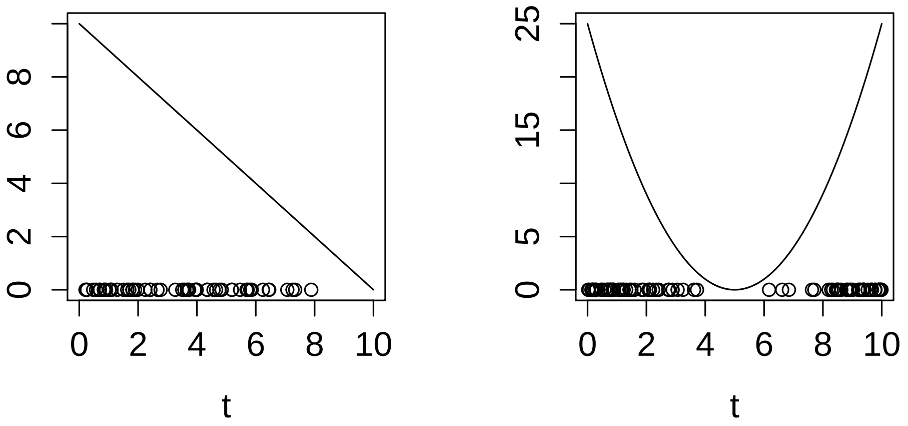

Figure 2.2: NHPP is simulated via thinning. The solid line represents the rate and the points are the jump times. Left: $\lambda(t) = 10 - t$, right: $\lambda(t) = (t - 5)^2$.

# Chapter 3

# Markov switching Cox processes

A Cox process is a generalisation of NHPP, where its intensity function is itself stochastic. There are many possibilities to describe the dynamics of the intensity function and we focus in a class which we refer as Markov switching Cox process (MSCP). This class of processes is defined following.

**Definition 3.1.** *Let $X = \{X(t) : t \geq 0\}$ be a Non-homogeneous Poisson process such that the intensity function $\lambda = \{\lambda(t) : t \geq 0\}$ is a continuous-time Markov chain. Then, $X$ is a Markov switching Cox process.*

In Section 3.1 we report some important results for MSCP's where the intensity is a two-state CTMC. Section 3.2 presents three important results derived for MSCP's. Section 3.3 contains two different algorithms to simulate a MSCP and some simulation examples.

## 3.1 Two-state Markov switching Cox processes

Taylor and Karlin (1998) present some properties for MSCP's in which the intensity function is a two-state CTMC, some results are reported following.

Let $X = \{X(t) : t \geq 0\}$ be a MSCP with intensity function $\lambda = \{\lambda(t) : t \geq 0\}$ a two-state CTMC, with state space $E = \{0, \gamma\}$. For notation, we say that $X$ is a MSCP$(0, \gamma)$ and $X\big((a, b]\big) := \{X(t) : t \in (a, b]\}$.

It can be shown that

$$f(t;\gamma) := P\Big(X\big((0,t]\big) = 0\Big) = E\left[e^{-\Lambda(t)}\right] = f_0(t) + f_1(t),$$

where

$$\Lambda(t) = \int_0^t \lambda(s)ds,$$

$$f_0(t) = P\Big(X\big((0,t]\big) = 0 \text{ and } \lambda(t) = 0\Big) \quad \text{and} \quad f_1(t) = P\Big(X\big((0,t]\big) = 0 \text{ and } \lambda(t) = \gamma\Big).$$

This result is valid only when the CTMC has one state equal to zero and the other equal to a positive constant $\gamma$. It is possible to generalise these results for any two-state CTMC. Let $E = \{\gamma_0, \gamma_1\}$ and assume $0 < \gamma_0 < \gamma_1$. Now, $X$ is a MSCP$(\gamma_1, \gamma_2)$. In order to evaluate $P\big(X\big((0,t]\big) = 0\big)$, we write $X$ as the sum $X = X_1 + X_2$ of two independent processes, where $X_1$ is a PP$(\gamma_0)$ and $X_2$ is a MSCP$(0, \gamma_1 - \gamma_0)$. Thus,

$$\begin{aligned} P\Big(X\big((0,t]\big) = 0\Big) &= P\Big(X_1\big((0,t]\big) = 0\Big)P\Big(X_2\big((0,t]\big) = 0\Big) \\ &= e^{-\gamma_0 t} f(t; \gamma_1 - \gamma_0). \end{aligned}$$

To find the probability distribution of $X\big((0,t]\big)$, let $X$ be a MSCP$(0,\gamma)$, suppose that we have evaluated

$$f\big(t; (1-\theta)\gamma\big) = E\left[e^{-(1-\theta)\Lambda(t)}\right], \qquad 0 < \theta < 1. \tag{3.1}$$

Expanding the equation (3.1) as a power series in $\theta$ we get

$$f\big(t; (1-\theta)\gamma\big) = \sum_{k=0}^{\infty} P\Big(X\big((0,t]\big) = k\Big)\theta^k,$$

hence, the coefficient of $\theta^k$ in the power series is the $P\Big(X\big((0,t]\big) = k\Big)$.

## 3.2 Some probabilistic properties

Other important results obtained for general MSCP's led to three new theorems. Theorem 3.2 concerns the MSCP's increments, Theorem 3.3 and Theorem 3.4 establishes the Markov property and asymptotic behaviour, respectively.

**Theorem 3.2.** *Let $X = \{X(t) : t \geq 0\}$ be a Markov switching Cox process with intensity function $\lambda = \{\lambda(t) : t \geq 0\}$. Then, the process has independent increments.*

*Proof.* Let $t_1 < t_2 \leq t_3 < t_4$ and $Y_1 := \{X(t) : t \in [t_1, t_2]\}$, $Y_2 := \{X(t) : t \in [t_3, t_4]\}$, $\lambda_1 := \{\lambda(t) : t \in [t_1, t_2]\}$ and $\lambda_2 := \{\lambda(t) : t \in [t_3, t_4]\}$.

$$\begin{aligned}
P(Y_1 = k_1, Y_2 = k_2) &= \mathbb{E}_{\lambda_1,\lambda_2}\Bigg[\mathbb{E}_{Y_1,Y_2}\left[\mathbb{I}_{\{Y_1=k_1\}}\mathbb{I}_{\{Y_2=k_2\}}|\lambda_1,\lambda_2\right]\Bigg] \\
&= \mathbb{E}_{\lambda_1,\lambda_2}\Bigg[\mathbb{E}_{Y_1}\left[\mathbb{I}_{\{Y_1=k_1\}}|\lambda_1\right]\mathbb{E}_{Y_2}\left[\mathbb{I}_{\{Y_2=k_2\}}|\lambda_2\right]\Bigg] \\
&= \mathbb{E}_{\lambda_1}\Bigg[\mathbb{E}_{\lambda_2}\Bigg[\mathbb{E}_{Y_1}\left[\mathbb{I}_{\{Y_1=k_1\}}|\lambda_1\right]\mathbb{E}_{Y_2}\left[\mathbb{I}_{\{Y_2=k_2\}}|\lambda_2\right]\Bigg|\lambda_1\Bigg]\Bigg] \\
&= \mathbb{E}_{\lambda_1}\Bigg[\mathbb{E}_{Y_1}\left[\mathbb{I}_{\{Y_1=k_1\}}|\lambda_1\right]\mathbb{E}_{\lambda_2}\Bigg[\mathbb{E}_{Y_2}\left[\mathbb{I}_{\{Y_2=k_2\}}|\lambda_2\right]\Bigg|\lambda(t_2)\Bigg]\Bigg] \\
&= \mathbb{E}_{\lambda_1}\Bigg[\mathbb{E}_{Y_1}\left[\mathbb{I}_{\{Y_1=k_1\}}|\lambda_1\right]\mathbb{E}_{Y_2}\left[\mathbb{I}_{\{Y_2=k_2\}}|\lambda(t_2)\right]\Bigg] \\
&= \mathbb{E}_{\lambda_1}\Bigg[\mathbb{E}_{Y_1}\left[\mathbb{I}_{\{Y_1=k_1\}}|\lambda_1\right]\Bigg]\mathbb{E}_{\lambda_1}\Bigg[\mathbb{E}_{Y_2}\left[\mathbb{I}_{\{Y_2=k_2\}}|\lambda(t_2)\right]\Bigg] \\
&= \mathbb{E}_{Y_1}\left[\mathbb{I}_{\{Y_1=k_1\}}\right]\mathbb{E}_{Y_2}\left[\mathbb{I}_{\{Y_2=k_2\}}\right] \\
&= P(Y_1 = k_1)P(Y_2 = k_2).
\end{aligned}$$

□

**Theorem 3.3.** *Let $X = \{X(t) : t \geq 0\}$ be a Markov switching Cox process with intensity $\lambda = \{\lambda(t) : t \geq 0\}$. Let $0 \leq t_1 < t_2 < \cdots < t_m < t$ and $n_1 \leq n_2 \leq \cdots \leq n_m \leq n$. Then,*

$$P\big(X(t) = n\big|X(t_1) = n_1, X(t_2) = n_2, \cdots, X(t_m) = n_m\big) = P\big(X(t) = n\big|X(t_m) = n_m\big).$$

*Proof.* $P\big(X(t) = n\big|X(t_1) = n_1, \cdots, X(t_m) = n_m\big) =$

$$= \int_\Lambda P\Big(X(t) = n\Big|X(t_1) = n_1, \cdots, X(t_m) = n_m, \lambda\big([0,t]\big)\Big)dP(\lambda([0,t]))$$

$$= \int_\Lambda P\Big(X(t) = n\Big|X(t_m) = n_m, \lambda([0,t])\Big) dP(\lambda([0,t]))$$

$$= P\big(X(t) = n\big|X(t_m) = n_m\big).$$

□

**Theorem 3.4.** *Let $X = \{X(t) : t \geq 0\}$ be a Markov switching Cox process with intensity $\lambda = \{\lambda(t) : t \geq 0\}$. Suppose that $\exists$ a random variable $\lambda^*$ such that $\lambda(t) \xrightarrow[t\to\infty]{d} \lambda^*$. Then, exist a discrete random variable $Y_h$, $\forall h > 0$, such that*

$$P\Big(X\big((t, t+h]\big) = k\Big) \to P(Y_h = k).$$

*Proof.* $\lambda(t) \xrightarrow[t\to\infty]{d} \lambda^*$ implies that $P\big(\lambda(t) = \epsilon\big) \xrightarrow[t\to\infty]{} p^*_\lambda(\epsilon), \forall \epsilon \in E$, for some p.m.f. $p^*_\lambda(\cdot)$ on $E$.

Define $I_{t,h} = \int_t^{t+h} \lambda(s)ds$ and consider the joint density of $\big(\lambda(t), I_{t,h}\big)$, which can be factorised as

$$\pi\big(\lambda(t), I_{t,h}\big) = \pi\big(\lambda(t)\big)\pi\big(I_{t,h}\big|\lambda(t)\big).$$

Now note that the first term converges to $p^*_\lambda(\cdot)$ and the second term is the same for all $t$, given the same value of $\lambda(t)$, by the time-homogeneity property of $\lambda$. Therefore, $\exists\ I_h$ such that $I_{t,h} \xrightarrow{d} I_h$ (Scheffé's Theorem).

Finally, note that

$$\begin{aligned} P\Big(X\big((t, t+h]\big) = k\Big) &= E\left[\mathbb{I}_{\{X((t,t+h])=k\}}\right] \\ &= E_{I_{t,h}}\left[E[\mathbb{I}_{\{X((t,t+h])=k\}}|I_{t,h}]\right] \\ &= E_{I_{t,h}}\left[\frac{e^{-I_{t,h}}(I_{t,h})^k}{k!}\right] \xrightarrow[t\to\infty]{} E_{I_h}\left[\frac{e^{-I_h}(I_h)^k}{k!}\right] = p_h(k), \end{aligned}$$

the convergence of the expectation is due to the fact that $E[\mathbb{I}_{\{X((t,t+h])=k\}}|I_{t,h}]$ is a continuous and bounded function of $I_{t,h}$. The result is established by defining $Y_h$ as the discrete random variable with probability mass function given by $p_h(\cdot)$.

□

## 3.3 Simulations

In order to simulate a MSCP in an interval $[0, T]$ the following algorithm can be used:

**Algorithm 6.**

```
1. simulate $\lambda(t)$ in $[0, T]$;
2. perform the Poisson thinning.
```

Algorithm 6 can be improved in terms of computation by reversing the two steps. The new algorithm is given below.

**Algorithm 7.**

```
Take $\lambda^* = sup_{t \in [0,T]}(\lambda(t))$.
1. Simulate $\lambda(0)$;
2. make $i = 1$;
3. generate $\Delta t \sim$ exponetial($\lambda^*$), and make $\tau_i = \tau_{i-1} + \Delta t$;
4. simulate $\lambda(\tau_i)$;
5. keep $\tau_i$ with probability $\frac{\lambda(\tau_i)}{\lambda^*}$;
6. if $\sum_{i=1}^{n} \tau_i > T$, stop and output the kept $\tau_i$'s. Otherwise, $i = i + 1$ and go
   back to step 3.
```

To illustrate a MSCP, we simulate two examples using Algorithm 6. Note that the algorithms 6 and 7 are equivalent, but only Algorithm 6 allow us to know the real intensity function in the whole interval. The results are displayed in Figures 3.1 and 3.2.

The first simulation, in Figure 3.1, the CTMC of the MSCP has state space $E = \{1, 4, 8\}$ and Q-matrix

$$Q = \begin{bmatrix} -1 & 0.5 & 0.5 \\ 0.5 & -2 & 1.5 \\ 1 & 2 & -3 \end{bmatrix}.$$

The second simulation, in Figure 3.2, the CTMC of the MSCP has state space $E = \{0, 3, 5, 10\}$ and Q-matrix

$$Q = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix}.$$

In both simulated processes, for higher rate were observed more events, that is, the waiting time until the next occurrence is lower. With respect to the process presented in Figure 3.2, no events occurred for time intervals with zero rate.

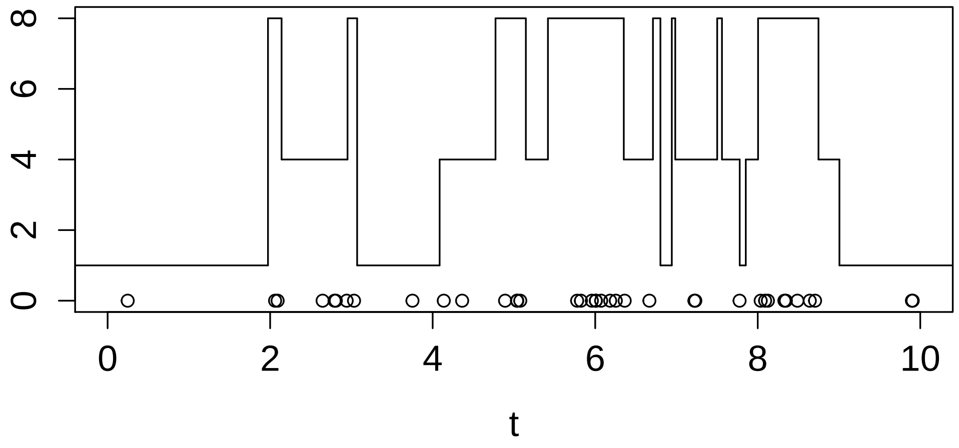

Figure 3.1: Realisation of a three-state MSCP

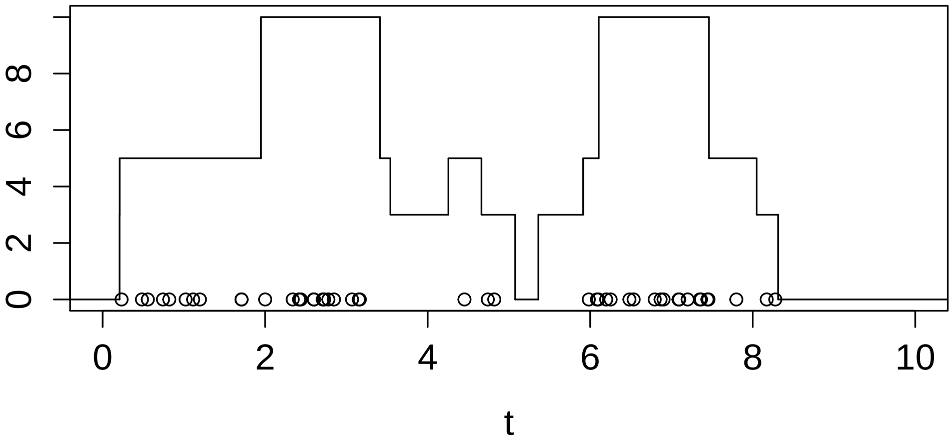

Figure 3.2: Realisation of a four-state MSCP.

# Chapter 4

# Bayesian Inference for Markov switching Cox processes

In this chapter we discuss the inference problem for a Markov switching Cox processes with finite state space. Section 4.1 is an introduction to the inference problem and presents all the notation used in this chapter. The likelihood function is presented in Section 4.2 and in Section 4.3 we define the prior distributions. The full conditional distributions and the respective algorithm to sample from them are presented in Sections 4.6 to 4.5. Section 4.7 discusses some identifiability issues.

## 4.1 The inference problem

Let $Y = \{Y(t) : 0 \leq t \leq T\}$ be a non-homogeneous Poisson process with intensity function $\lambda$ and $\lambda = \{\lambda(t) : 0 \leq t \leq T\}$ a CTMC with Q-matrix $Q_\theta$, initial distribution $\pi_{\theta_0}$ and state space E, where $\theta$ and $\theta_0$ are parameter vectors associated to the entries of the Q-matrix and initial distribution, respectively.

Based on a realisation of Y in $[0, T]$, we want to estimate:

1. $(\lambda, \theta, E)$;
2. $\mathbb{E}(f(Y, \lambda, \theta, E))$, for suitable $f's$.

The size of E, denoted by $|E|$, is fixed and we estimate only the values of the state space.

Inference is performed under the Bayesian paradigm which means that inference should be based on the posterior distribution $\pi(\lambda, \theta, E|y)$ where $y$ is the realisation of Y.

By Bayes Theorem (see Shao, 2003)

$$\pi(\lambda, \theta, E|y) \propto \pi(y|\lambda, \theta, E)\pi(\lambda|\theta, E)\pi(E)\pi(\theta), \tag{4.1}$$

where the $\pi$'s are densities with respect to suitable dominating measures.

The posterior distribution in (4.1) is quite complex due to the infinite-dimensional nature of $\lambda$. For that reason, the use of Monte Carlo methods is the best way to explore this distribution. In particular, we devise a Markov chain Monte Carlo (MCMC) algorithm to sample from the joint posterior distribution in (4.1).

The MCMC consists of generating samples from a Markov chain which has the joint posterior distribution in (4.1) as its invariant distribution. Thus, after a sufficient number of iterations of the chain we have an approximate sample from the joint posterior distribution which is good enough to provide reasonable estimates.

The algorithm is a Gibbs sampler, that breaks the vector $(\lambda, \theta, E)$ in blocks which are sampled from their respective full conditional distributions at each iteration of the chain. We adopt the following blocking scheme:

$$\{\lambda\}\ ,\ \{\theta\}\ ,\ \{E\},$$

with respective full conditional distributions

$$(\lambda|\theta, E, y), (\theta|\lambda, E, y), (E|\lambda, \theta, y).$$

Before presenting the full conditional distributions, will establish some notation to be used throughout this chapter. The process Y is characterized by the times when the Poisson events occur, then $Y = (\tau_1, \cdots, \tau_N)$, where $N$ is the total events in the observed time interval. Conditioned on $\lambda$, N has a Poisson distribution with parameter $\int_0^T \lambda(s)ds$. The state space is $E = \{\epsilon_1, \cdots, \epsilon_{|E|}\}$, $M_E$ and $m_E$ are the $max\{E\}$ and the $min\{E\}$, respectively. We denote $n$ as the number of observed events.

The process $\lambda$ is characterized by jump times $(S_0, S_1, \cdots, S_m)$ and the trajectory of the states $(\lambda(S_0), \lambda(S_1), \cdots, \lambda(S_m))$, where $\lambda(S_i)$ is the state of the Markov chain in the time interval $[S_i, S_{i+1})$. Note that, if the process is in the time interval $[0, T]$, $S_{m+1} = T$. For inference purposes, we construct a second representation of a trajectory of $\lambda$. For this, it will not be considered the real value of state but an enumeration of $E$. We use $t_{1\boldsymbol{\cdot}} = (t_{1i_1} : i_1 = 0, \cdots, m_1)$ to denote the waiting times in state 1, that is, $t_{1i_1}$ is the time that the chain remains in state 1 in its $i$-th visit. Note that, if the chain does not visit state 1, $t_{1\boldsymbol{\cdot}} = 0$. In general, we have $t_{j\boldsymbol{\cdot}} = (t_{ji_j} : i_j = 0, \cdots, m_j)$ for $j = 1, \cdots, |E|$ and $t_{\boldsymbol{\cdot\cdot}}$ is the vector obtained from the concatenation of all the $t_{j\boldsymbol{\cdot}}$ vectors. Furthermore, $m_j$ is the number of times that the Markov chain visits state $j$ so that $\sum_{j=1}^{|E|} m_j - 1 = m$. Also, $m_{i\boldsymbol{\cdot}} = (m_{ij} : j = 1, \cdots, |E|)$ is the vector of the number of jumps from state $i$ to state $j$, for example, $m_{12}$ is the number of transitions from state 1 to state 2. Note that when $i = j$ the entry of the vector is equal to 0. Finally, $m_{\boldsymbol{\cdot\cdot}} = [m_{ij}]$, for $i = 1, \cdots, |E|$ and $j = 1, \cdots, |E|$, is the matrix with the number of jumps which has null diagonal. We define the statistic $\lambda_{\boldsymbol{\cdot}} = (m_{\boldsymbol{\cdot\cdot}}, t_{\boldsymbol{\cdot\cdot}})$ of $\lambda$ which, as we will demonstrate, is a sufficient statistic for $\theta$.

The parameter vector $\theta$ is associated to all entries of the Q-matrix. We define $\theta_j$ as the rates of the waiting times and $\theta_{ji}$, for $j \neq i$, as the transition probabilities. Thus,

$$\theta = (\theta_1, \cdots, \theta_{|E|}, \theta_{12}, \cdots, \theta_{1|E|}, \theta_{21}, \cdots, \theta_{2|E|}, \cdots, \theta_{|E|1}, \cdots, \theta_{|E|(|E|-1)})$$

and the Q-matrix is given by

$$Q = \begin{bmatrix} -\theta_1 & \theta_1\theta_{12} & \cdots & \theta_1\theta_{1|E|} \\ & & \ddots & \\ & & & -\theta_{|E|} \end{bmatrix}.$$

Note that each row of the Q-matrix, without the diagonal term, upon normalisation, is a probability distribution. Let $\theta_{j\boldsymbol{\cdot}}$ be the transition probability vector starting from state $j$, that is, the probability distribution given in the $j$-th row.

For inference reasons which will be clarified further ahead in the text, we need to define a partition of the intensity function $\lambda$. We define this partition by splitting the interval $[0, T]$ into B sub-intervals. Let $[t_{k-1}, t_k]$ be the $k$-th sub-interval and $\lambda_{(k)}$ the process $\lambda$ in this interval. Let also $\lambda_{(-k)}$ be the process $\lambda$ in $[0, T] \setminus (t_{k-1}, t_k)$ and $\lambda^*_{(k)}$

the end points of $\lambda_{(k)}$. Furthermore, $n_{(k)}$ and $m_{(k)}$ are the number of observed events and the number of jumps of the Markov chain, respectively, in the $k$-th interval.

## 4.2 Likelihood function

The likelihood function $L(\lambda, \theta, E)$ is the density $\pi(Y|\lambda, \theta, E)$ evaluated at the observed $y$. This density is the Radon-Nikodym derivate $\frac{dP}{dP_0}(Y)$, where $P$ is the probability measure of Y and $P_0$ is the probability measure of a homogeneous PP with known rate, e.g., PP(1). These two measures are defined on the same measurable space and are equivalent ($P \ll P_0$ and $P_0 \ll P$, $\ll$ means absolute continuous with respect to). Then, we have (Cox and Lewis, 1966)

$$\begin{aligned}\pi(Y|\lambda, \theta, E) := L(\lambda, \theta, E) = \frac{dP}{dP_0}(Y) &\propto \left[ e^{-\int_0^T \lambda(s)ds} \prod_{i=1}^{n} \lambda(\tau_i) \right] \\ &\propto \left[ e^{-\sum_{j=0}^{m} (S_{j+1}-S_j)\lambda(S_j)} \prod_{i=1}^{n} \lambda(\tau_i) \right],\end{aligned}$$

where the proportionality sign refers to $\lambda$.

## 4.3 Prior distributions

The Bayesian approach requires the assignment of prior distributions to the parameter vector. Firstly, we take $\theta = \{\theta_{\boldsymbol{\cdot}}, \theta_{\boldsymbol{\cdot\cdot}}\}$, where $\theta_{\boldsymbol{\cdot}}$ is the vector of all $\theta_j$'s and $\theta_{\boldsymbol{\cdot\cdot}}$ is the vector of all $\theta_{j\boldsymbol{\cdot}}$'s. We assume independence among $\theta_{\boldsymbol{\cdot}}, \theta_{\boldsymbol{\cdot\cdot}}$ and $E$. We also, assume prior independence of all $\theta_j$'s and $\theta_{j\boldsymbol{\cdot}}$'s. The prior distribution is given by

$$\begin{aligned}\pi(\theta, E) &= \left[ \prod_{j=1}^{|E|} \pi(\theta_j)\pi(\theta_{j\boldsymbol{\cdot}}) \right] \pi(E) \\ &= \left[ \prod_{j=1}^{|E|} f_G(\theta_j; \alpha_j, \beta_j) f_D(\theta_{j\boldsymbol{\cdot}}; \gamma_{j1}, \cdots, \gamma_{j(|E|-1)}) f_G(\epsilon_j; \alpha_{\epsilon_j}, \beta_{\epsilon_j}) \right] \mathbb{I}(\epsilon_1 < \cdots < \epsilon_{|E|}),\end{aligned}$$

where $f_G$ is the probability density function of a Gamma distribution and $f_D$ is the probability density function of a Dirichlet distribution. Therefore, $\theta_j \sim \text{Gamma}(\alpha_j, \beta_j)$,

$\theta_{j\bullet} \sim$ Dirichlet$(\gamma_{j1}, \cdots, \gamma_{j(|E|-1)})$, $\epsilon_j \sim$ Gamma$(\alpha_{\epsilon_j}, \beta_{\epsilon_j})$ and the indicator function gives the dependence structure of the $\epsilon_j$'s.

## 4.4 The full conditional distribution of $\theta$

In order to sample $\theta$, we break this vector into two blocks: $\theta_{\bullet}$ and $\theta_{\bullet\bullet}$. Now, note that

$$\pi(\lambda|\theta_{\bullet}, \theta_{\bullet\bullet}, E) = \pi(t_{\bullet\bullet}, m_{\bullet\bullet}|\theta_{\bullet}, \theta_{\bullet\bullet}) = \pi(t_{\bullet\bullet}|\theta_{\bullet})\pi(m_{\bullet\bullet}|\theta_{\bullet\bullet}).$$

Thus, the full conditional distribution for $\theta_{\bullet}$ is given by

$$\begin{aligned}\pi(\theta_{\bullet}|\lambda, E, Y) &\propto \pi(t_{\bullet\bullet}|\theta_{\bullet})\pi(\theta_{\bullet})\\ &= \prod_{j=1}^{|E|}\prod_{i_j=1}^{m_j} \pi(t_{ji_j}|\theta_j) \prod_{j=1}^{|E|} f_G(\theta_j; \alpha_j, \beta_j)\\ &\propto \left(\prod_{j\in(E\backslash e)} \theta_j^{m_j} e^{-\theta_j \Delta(\epsilon_j)} \theta_j^{\alpha_j - 1} e^{-\theta_j\beta_j}\right) \theta_e^{m_e} e^{-\theta_e\Delta(\epsilon_e)} \theta_e^{\alpha_e-1} e^{-\theta_e\beta_e}\\ &\propto \prod_{j\in(E\backslash e)} f_G(\theta_j; \alpha_j + m_j, \beta_j + \Delta(\epsilon_j)) + f_G(\theta_e; \alpha_e + m_e - 1, \beta_e + \Delta(\epsilon_e)),\end{aligned}$$

where $\Delta(\epsilon_j) = \sum_{i_j=0}^{m_j} t_{ji_j}$ is the total time that the Markov chain $\lambda$ remained in state $\epsilon_j$ and $e$ is the last visited state in the CTMC. Note that each $t_{ji_j}|\theta_j \sim$ Exponential$(\theta_j)$. Set $t_e$ as the last time spent in $e$. This event is equivalent to an Exponential$(\theta_e)$ random variable being greater or equal to $t_e$ and, therefore, has probability $e^{-\theta_e t_e}$.

Therefore, the full conditional distribution for each $\theta_j$, for $j \neq e$, is Gamma$(\alpha_j + m_j, \beta_j + \Delta(\epsilon_j))$, for $\theta_e$ is Gamma$(\alpha_e + m_e - 1, \beta_e + \Delta(\epsilon_e))$ and they are all independent.

The full conditional distribution for $\theta_{\bullet\bullet}$ is given by

$$\begin{aligned}\pi(\theta_{\bullet\bullet}|\lambda, E, Y) &\propto \pi(m_{\bullet\bullet}|\theta_{\bullet\bullet})\pi(\theta_{\bullet\bullet})\\ &\propto \prod_{j=1}^{|E|} \pi(m_{j\bullet}|\theta_{j\bullet}) \prod_{j=1}^{|E|} f_D(\theta_{j\bullet}; \gamma_{j1}, \cdots, \gamma_{j(|E|-1)})\end{aligned}$$

$$\propto \prod_{j=1}^{|E|} \prod_{i \neq j, i=1}^{|E|-1} \theta_{ji}^{m_{ji}} \theta_{ji}^{\gamma_{ji}-1} \propto \prod_{j=1}^{|E|} f_D(\theta_{j\boldsymbol{\cdot}}; m_{j1}+\gamma_{j1}, \cdots, m_{j(|E|-1)}+\gamma_{j(|E|-1)}).$$

Therefore, the full conditional distribution for each $\theta_{j\boldsymbol{\cdot}}$ is
Dirichlet$\big(m_{j1}+\gamma_{j1}, \cdots, m_{j(|E|-1)}+\gamma_{j(|E|-1)}\big)$ and they are all independent.

## 4.5 The full conditional distribution of E

The full conditional of E is given by

$$\pi(E|\lambda, \theta, Y) \propto L(\lambda, \theta, E)\pi(\lambda|\theta, E)\pi(E).$$

However, the following proposition shows that $\pi(\lambda|\theta, E)$ is constant with respect to E.

**Proposition 4.1.** *Consider two state spaces $E = (\epsilon_1, \cdots, \epsilon_{|E|})$ and $E' = (\epsilon'_1, \cdots, \epsilon'_{|E|})$, both with size $|E|$. Then, $\pi(\lambda|\theta, E) = \pi(\lambda|\theta, E')$.*

*Proof.*

$$\begin{aligned}
\pi(\lambda|\theta, E) &= \pi(S_0, S_1, \cdots, S_m, \lambda(0), \lambda(S_0), \cdots, \lambda(S_m)|\theta, E) \\
&= \pi(\lambda(0)|\theta, E)\pi(S_0|\lambda(0), \theta, E)\pi(\lambda(S_0)|\lambda(0), \theta, E)\pi(S_1|S_0, \lambda(S_0), \theta, E) \\
&\qquad \pi(\lambda(S_1)|\lambda(S_0), \theta, E)\cdots\pi(S_m|S_{m-1}, \lambda(S_{m-1}), \theta, E) \\
&\qquad\qquad \pi(\lambda(S_m)|\lambda(S_{m-1}), \theta, E) \\
&= \pi_{\theta_0}(\lambda(0))\pi_{exp}(S_0|q_{\lambda(0)})p_{\lambda(0)\lambda(S_0)}(S_0)\pi_{exp}(S_1-S_0|q_{\lambda(S_0)})p_{\lambda(S_0)\lambda(S_1)}(S_1-S_0) \\
&\qquad \cdots\pi_{exp}(S_m-S_{m-1}|q_{\lambda(S_{m-1})})p_{\lambda(S_{m-1})\lambda(S_m)}(S_m-S_{m-1}), \qquad (4.2)
\end{aligned}$$

where $q_{\lambda(S_i)}$ and $p_{\lambda(S_{i-1})\lambda(S_i)}(S_i - S_{i-1})$ are defined in Section 2.1.2. Equation (4.2) shows that $\pi(\lambda|\theta, E)$ does not depend on the actual value of E, but its enumeration. □

Then, by Proposition 4.1,

$$\pi(E|\lambda, \theta, Y) \propto L(\lambda, \theta, E)\pi(E)$$

$$\propto exp\left\{-\sum_{j=0}^{m}(S_{j+1}-S_j)\lambda(S_j)\right\}\prod_{i=1}^{n}\lambda(\tau_i)\prod_{j=1}^{|E|}f_G(\epsilon_j;\alpha_{\epsilon_j},\beta_{\epsilon_j})\mathbb{I}(\epsilon_1<\cdots<\epsilon_{|E|}).$$

Let $n_{(\epsilon_j)}$ be the number of Poisson events occurring during the time period where the intensity function is $\epsilon_j$. Thus

$$\pi(E|\lambda,\theta,Y)\propto exp\left\{-\sum_{j=1}^{|E|}\Delta(\epsilon_j)\epsilon_j\right\}\prod_{j=1}^{|E|}\epsilon_j^{n_{(\epsilon_j)}}\prod_{j=1}^{|E|}\epsilon_j^{\alpha_{\epsilon_j}-1}e^{-\beta_{\epsilon_j}\epsilon_j}\mathbb{I}(\epsilon_1<\cdots<\epsilon_{|E|})$$

$$\propto\prod_{j=1}^{|E|}\epsilon_j^{\alpha_{\epsilon_j}+n_{(\epsilon_j)}-1}e^{-(\beta_{\epsilon_j}+\Delta(\epsilon_j))\epsilon_j}\mathbb{I}(\epsilon_1<\cdots<\epsilon_{|E|})$$

$$\propto\prod_{j=1}^{|E|}f_G(\epsilon_j;\alpha_{\epsilon_j}+n_{(\epsilon_j)},\beta_{\epsilon_j}+\Delta(\epsilon_j))\mathbb{I}(\epsilon_1<\cdots<\epsilon_{|E|}).$$

We sample from this distribution via RS by proposing from the independent Gamma distribution for each state and accepting if $\mathbb{I}(\epsilon_1<\cdots<\epsilon_{|E|})=1$.

## 4.6 The full conditional distribution of $\lambda$

The most challenging step of the MCMC algorithm is the one where $\lambda$ is sampled from its full conditional distribution, we have that

$$\pi(\lambda|\theta,E,Y)\propto L(\lambda,\theta,E)\pi(\lambda|\theta,E)\propto\left[e^{-\int_0^T\lambda(s)ds}\prod_{i=1}^{n}\lambda(\tau_i)\right]\pi(\lambda|\theta,E).$$

It is very hard to construct an efficient algorithm to sample from the full conditional of $\lambda$ in the whole observed interval $[0,T]$. Two (inefficient) possibilities are a Rejection Sampling (RS) and a Metropolis-Hastings (MH) algorithm, both proposing from $\pi(\lambda|\theta,E)$. The acceptance probability of the RS is given by

$$exp\left\{-\int_0^T(\lambda(s)-m_E)ds\right\}\prod_{i=1}^{n}\left(\frac{\lambda(\tau_i)}{M_E}\right),$$

which decreases exponentially in T. The acceptance probability of the MH is given by

$$min\left\{1,\frac{L(\lambda_{\text{``}new\text{''}},\theta,E)}{L(\lambda_{\text{``}old\text{''}},\theta,E)}\right\}.$$

The RS would have a high computational cost and the MH a small acceptance rate leading to slow convergence of the chain.

A possible solution for this problem is to adopt a partition $\lambda_{(1)},\cdots,\lambda_{(B)}$ of $\lambda$, as defined previously, and update $\lambda$ interval-wise. The problem with this approach is that $\lambda$ would never be updated at the end points of each interval. To overcome this issue, we introduce an auxiliary variable $U$ in the Markov chain such that $U\sim$ uniform$(0,L)$, for a suitable choice of L and define the partition $[0,U],(U,U+L],(U+L,U+2L],\cdots,(U+(B-2)L,T]$. The fact that $U$ changes in every iteration of the chain guarantees its irreducibility. We call this the overlapping Gibbs sampling.

Naturally, the efficiency of the algorithm relies heavily on the choice of $L$. Small values of $L$ lead to high acceptance probability/rate but increases the number of blocks and, consequently, the autocorrelation of the chain which, in turn, affects its convergence. For a large $L$ we have the opposite problem.

After breaking $\lambda$ in blocks, the new full conditional distribution of each block is given by

$$\begin{aligned}\pi(\lambda_{(k)}|\theta,E,\lambda_{(-k)},Y)&\propto L(\lambda,\theta,E)\pi(\lambda_{(k)}|\theta,E,\lambda_{(-k)})\\&\propto\left[e^{-\sum_{j=0}^{m_{(k)}}(S_{k,j+1}-S_{k,j})\lambda(S_{k,j})}\prod_{i=1}^{n_{(k)}}\lambda(\tau_{k,i})\right]\pi(\lambda_{(k)}|\theta,E,\lambda^{*}_{(k)}),\quad(4.3)\end{aligned}$$

where $S_{k,j}$ is the $j$-th jump time of the Markov chain and $\tau_{k,i}$ is the $i$-th realisation of Y, both in the $k$-th interval.

Also, we partition $\lambda_{(k)}$ as $\{\lambda(\tau_{k,i}),i=1,\cdots,n_{(k)}\}$ and $\{\lambda_{(k,i)},i=1,\cdots,n_{(k)}+1\}$, where $\lambda(\tau_{k,i})$ is the intensity function at the time instance $\tau_{k,i}$ and $\lambda_{(k,i)}$ is the intensity function between the $(i-1)$-th and the $i$-th event of the PP. Set $\tau_{k,0}=t_{i-1}$ and $\tau_{k,n_{(k)}+1}=t_i$. Note that $\lambda_{(k,i)}$ is infinite-dimensional. We have the following factorisation for the equation (4.3)

$$\pi(\lambda_{(k)}|\theta, E, \lambda_{(-k)}, Y) \propto \left[e^{-\sum_{j=0}^{m(k)}(S_{k,j+1}-S_{k,j})\lambda(S_{k,j})}\prod_{i=1}^{n_{(k)}}\lambda(\tau_{k,i})\right] \underbrace{\prod_{i=1}^{n_{(k)}+1}\pi\left(\lambda_{(k,i)}|\cdot\right)\prod_{i=1}^{n_{(k)}}\pi\left(\lambda(\tau_{k,i})|\cdot\right)}, \tag{4.4}$$

where,

$$\pi\left(\lambda_{(k,i)}|\cdot\right) = \pi\left(\lambda_{(k,i)}|\theta, E, \lambda(\tau_{k,i-1}), \lambda(\tau_{k,i})\right)$$

and

$$\pi\left(\lambda(\tau_{k,i})|\cdot\right) = \pi\left(\lambda(\tau_{k,i})|\theta, E, \lambda(\tau_{k,i-1}), \lambda(\tau_{k,n_{(k)}+1})\right).$$

Considering both RS and MH alternatives, the first possibility we consider is to propose from the prior, given by the underbraced term in (4.4). In this case, the acceptance probability of the RS is proportional to the likelihood and the acceptance probability of a move in the MH is the minimum of 1 and the ration of likelihoods at new and old values.

This proposal may be improved by using information from the data. We shall devise a decomposition family of the full conditional distribution such that there is a part where a new $\lambda_{(k)}$ is proposed from and one that defines the acceptance probability of both algorithms. Consider the following decomposition

$$\pi(\lambda_{(k)}|\theta, E, \lambda_{(-k)}, Y) \propto \underbrace{\left[e^{-\sum_{j=0}^{m(k)}(S_{k,j+1}-S_{k,j})\lambda(S_{k,j})}\prod_{i=1}^{n_{(k)}}\lambda(\tau_{k,i})^{1-r}\right]} \underbrace{\prod_{i=1}^{n_{(k)}+1}\pi\left(\lambda_{(k,i)}|\cdot\right)}\underbrace{\prod_{i=1}^{n_{(k)}}\lambda(\tau_{k,i})^{r}\pi\left(\lambda(\tau_{k,i})|\cdot\right)}, \tag{4.5}$$

where $r \geq 0$. The idea of using this $r$ function is due to the fact that for each case the algorithm's convergence may be faster for a suitable choice of $r$.

To obtain a sample from the full conditional distribution given in (4.5), first we sample $\lambda(\tau_{k,i})$'s from the density given by the third underbraced term. Then, the simulated $\lambda(\tau_{k,i})$'s are used to sample the bridges from the density in the second underbraced term. Finally, the first underbraced term is used to define the acceptance probability of the algorithms.

The distribution of $\lambda(\tau_{k,i})$ defined by the third underbraced term in (4.5) is a discrete distribution in $E$, for which the probability vector is obtained as follows. Firstly, it is necessary to obtain the probability vector $(p_1, \cdots, p_{|E|})$ where,

$$\begin{aligned} p_l &= P\left(\lambda(\tau_{k,i}) = l|\lambda(\tau_{k,i-1}) = e_1, \lambda(\tau_{k,n_{(k)}+1}) = e_2\right) \\ &\propto P\left(\lambda(\tau_{k,i-1}) = e_1, \lambda(\tau_{k,i}) = l, \lambda(\tau_{k,n_{(k)}+1}) = e_2\right) \\ &\propto P\left(\lambda(\tau_{k,i}) = l|\lambda(\tau_{k,i-1}) = e_1\right) P\left(\lambda(\tau_{k,n_{(k)}+1}) = e_2|\lambda(\tau_{k,i}) = l\right). \end{aligned} \tag{4.6}$$

These transition probabilities are obtained from the matrix given by equation (2.2). Therefore, the distribution of $\lambda(\tau_{k,i})$ is proportional to $(\epsilon_1^r p_1, \cdots, \epsilon_{|E|}^r p_{|E|})$. The actual probability vector is obtained upon normalisation of this.

In order to sample the trajectories $\lambda_{(k,i)}$, we need to sample a Markov chain conditioned on initial and ending states, that is, a bridge of a CTMC. We present three possible algorithms to do this in Section 4.6.3.

For the proposal in equation (4.5) we have that the acceptance probability of the RS is given by

$$\begin{cases} exp\left\{-\sum_{j=0}^{m_{(k)}}(S_{k,j+1}-S_{k,j})\lambda(S_{k,j}) + (t_k - t_{k-1})m_E\right\}\prod_{i=1}^{n_{(k)}}\left(\frac{\lambda(\tau_{k,i})}{M_E}\right)^{1-r}, & \text{if } r \leq 1, \\ exp\left\{-\sum_{j=0}^{m_{(k)}}(S_{k,j+1}-S_{k,j})\lambda(S_{k,j}) + (t_k - t_{k-1})m_E\right\}\prod_{i=1}^{n_{(k)}}\left(\frac{\lambda(\tau_{k,i})}{m_E}\right)^{1-r}, & \text{if } r > 1. \end{cases} \tag{4.7}$$

The acceptance probability $\alpha(\lambda, \lambda^*)$ of the MH is given by

$$\alpha(\lambda, \lambda^*) = min\left\{1, \frac{exp\left\{-\sum_{j=0}^{m^*_{(k)}}(S^*_{k,j+1}-S^*_{k,j})\lambda^*(S^*_{k,j})\right\}\prod_{i=1}^{n_{(k)}}\left(\lambda^*(\tau_{k,i})\right)^{1-r}}{exp\left\{-\sum_{j=0}^{m_{(k)}}(S_{k,j+1}-S_{k,j})\lambda(S_{k,j})\right\}\prod_{i=1}^{n_{(k)}}\left(\lambda(\tau_{k,i})\right)^{1-r}}\right\}, \tag{4.8}$$

where all terms marked by an asterisk refer to the proposal trajectory and the other ones refer to the current state of the chain.

### 4.6.1 Comparing the algorithms

We now compare the RS and MH alternatives described above. The former has the advantage of outputing and exact draw from the full conditional distribution. However, its use requires smaller intervals to provide reasonable computational cost. That is because the acceptance probability of the RS decreases faster then the acceptance probability of the MH as the time interval increases. Small values of $L$, however, lead to two main problems. Firstly, it increases the correlation among the blocks of the Gibbs sampler, as smaller $L$'s lead to more blocks. Secondly, it deteriorates the mixing of the chain if the waiting times are large w.r.t. $L$. Basically, the algorithm will take too long to sample a jump.

The MH allows us to work with larger values of $L$. Although larger values of $L$ lead to lower acceptance rates, we can mitigate the problem by perform several MH updates at each iteration of the Gibbs sampler.

Therefore, the general strategy for the RS is to choose the largest possible $L$ that still provides a reasonable computational cost. In the case of MH, we should consider the largest $L$ that leads to reasonable acceptance rate, considering multiple MH updates per iteration.

Note that our MH algorithm is an independent Metropolis as the proposal does not depend on the previous iteration of the chain. This implies that the MH chains are either uniformly ergodic or not geometrically ergodic. Fortunately, the former is true in our case as it is established in the following Lemma.

**Lemma 4.2.** *The Metropolis-Hastings chain defined in (4.5) and (4.8) is uniformly ergodic.*

*Proof.* To establish uniform ergodicity it is enough to show that the spectral gap $\frac{q}{\pi}$ is uniformly bounded below by a positive constant $\beta$ (see Mengersen and Tweedie, 1996), where $q(\lambda_{(k)})$ is the proposal distribution. We have that

$$\frac{q}{\pi}(\lambda_{(k)}) = \frac{exp\left\{\sum_{j=0}^{m_{(k)}}(S_{k,j+1} - S_{k,j})\lambda(S_{k,j})\right\}}{\prod_{i=1}^{n_{(k)}}\left(\lambda(\tau_{k,i})\right)^{1-r}}$$

$$\geq \begin{cases} \frac{exp\{m_E(t_k - t_{k-1})\}}{\prod_{i=1}^{n(k)} M_E^{1-r}}, & \text{if } r \leq 1 \\ \frac{exp\{m_E(t_k - t_{k-1})\}}{\prod_{i=1}^{n(k)} m_E^{1-r}}, & \text{if } r > 1 \end{cases} > 0.$$

□

### 4.6.2 Sampling $\lambda$ in the first and last intervals

Sampling $\lambda$ in the first and last sub-interval is different since the initial and final states need to be updated at each MCMC iteration.

For the first interval it is necessary to do a backward sampling for the $\lambda(\tau_{1,i})'s$, conditional on $\lambda(t_1)$ and then to sample the bridge(s). The probability vector of $\lambda(\tau_{1,i})$ is obtained as follow

$$\begin{aligned} p_l &= P\left(\lambda(\tau_{1,i}) = l | \lambda(\tau_{1,n_{(1)}+1}) = e\right) \\ &\propto P\left(\lambda(\tau_{1,n_{(1)}+1}) = e | \lambda(\tau_{1,i}) = l\right) P\left(\lambda(\tau_{1,i}) = l\right) \\ &= P\left(\lambda(\tau_{1,n_{(1)}+1}) = e | \lambda(\tau_{1,i}) = l\right) \sum_{j=1}^{|E|} P\left(\lambda(\tau_{1,i}) = l | \lambda(t_0) = \epsilon_j\right) P\left(\lambda(t_0) = \epsilon_j\right), \end{aligned} \tag{4.9}$$

where $P(\lambda(t_0) = \epsilon_j)$ is obtained by the initial distribution of the CTMC. The actual distribution is obtained upon normalisation of these values.

Finally, the last interval is only conditioned on the initial state and, therefore, consists of a forward sampling of the $\lambda(\tau_{B,i})'s$. We remain with the bridges and the last trajectory in $(\tau_{B,n_{(B)}}, T]$ is only conditioned on $\lambda(\tau_{B,n_{(B)}})$.

### 4.6.3 Sampling bridges from CTMC's

Hobolth and Stone (2009) present three algorithms to simulate a realisation of a finite-state CTMC $X = \{X(t) : 0 \leq t \leq T\}$ endpoint-conditioned, that is, conditional on its initial state $X(0) = a$ and ending state $X(T) = b$.

Before presenting these algorithms, note that the simplest (but extremely expensive) way to sample a conditioned CTMC is by simulating it unconditionally on the end point and accepting as a realisation if it hits the correct end point. This is a RS algorithm and has a very high cost due to its small acceptance probability. Moreover, this cost increases with the length of the time interval.

The first method proposed in Hobolth and Stone (2009) is the modified rejection sampling. Conditioned that at least one state change occurs before $T$ and $X(0) = a$, the time $\tau$ to the first state change has density

$$f(\tau) = \frac{q_a e^{-\tau q_a}}{1 - e^{-T q_a}}, \qquad 0 \leq \tau \leq T, \tag{4.10}$$

where $q_a$ is given in Section 2.1.2. Based on this result, we have the following algorithm.

**Algorithm 8** (Modified rejection sampling)**.**

If $a = b$:

1. Simulate from $\{X(t) : 0 \leq t \leq T\}$ using the Algorithm 1;
2. accept the simulated path if $X(T) = a$; otherwise, return to step 1.

If $a \neq b$:

1. Sample $\tau$ from the density (4.10) using the inverse transformation method, and choose a new state $c \neq a$ from a discrete probability distribution with probability masses $q_{ac}/q_a$;
2. simulate the remainder $\{X(t) : \tau \leq t \leq T\}$ using the Algorithm 1 from the beginning state $X(\tau) = c$;
3. accept the simulated path if $X(T) = b$; otherwise, return to step 1.

The second algorithm is the direct sampling procedure. It requires that the Q-matrix admits an eigenvalue decomposition, i.e., $Q = U D_\zeta U^{-1}$ where $\zeta$ are the corresponding eigenvalues. Then, the transition probability matrix of the CTMC can be obtained by

$$P(t) = e^{tQ} = U e^{t D_\zeta} U^{-1} \quad \text{and} \quad (P(t))_{ab} = \sum_j U_{aj} U^{-1}_{jb} e^{t\zeta_j}.$$

Conditioned on $X(0) = a$ and $X(T) = a$ the probability that there are no state changes in the time interval $[0, T]$ is given by

$$p_a = \frac{e^{-q_a T}}{(P(T))_{aa}}. \tag{4.11}$$

On the other hand, if $X(T) = b \neq a$, the probability that the first state change is to $i$ is

$$p_i = \int_0^T f_i(t)dt, \qquad i \neq a.$$

The integrand can be written as

$$f_i(t) = \frac{q_{ai}}{(P(T))_{ab}} \sum_j U_{ij} U_{jb}^{-1} e^{T\zeta_j} e^{-t(\zeta_j + q_a)}, \tag{4.12}$$

thus,

$$p_i = \frac{q_{ai}}{(P(T))_{ab}} \sum_j U_{ij} U_{jb}^{-1} J_{aj}, \tag{4.13}$$

where

$$J_{aj} = \begin{cases} Te^{T\zeta_j}, & \text{if } \zeta_j + q_a = 0, \\ \frac{e^{T\zeta_j - e^{-q_a T}}}{\zeta_j + q_a}, & \text{if } \zeta_j + q_a \neq 0. \end{cases}$$

**Algorithm 9** (Direct sampling)**.**

1. If $a = b$, sample $Z \sim \text{Bernoulli}(p_a)$, where $p_a$ is given by equation (4.11). If $Z = 1$, we are done: $X(t) = a$, $0 \leq t \leq T$;

2. if $a \neq b$ or $Z = 0$, then at least one state change occurs. Calculate $p_i$ for all $i \neq a$ from equation (4.13). Sample $i \neq a$ from the discrete probability distribution with probability masses $p_i/p_{-a}$, $i \neq a$, where $p_{-a} = \sum_{j \neq a} p_j$;

3. sample the waiting time $\tau$ in state $a$ according to the continuous density $f_i(t)/p_i$, $0 \leq t \leq T$, where $f_i(t)$ is given by equation (4.12) and simulate from this density using the inverse transformation method. Set $X(t) = a$, $0 \leq t < \tau$;

4. repeat procedure with new starting value $i$ and new time interval of length $T - \tau$.

The drawback from this algorithm is that the inversion on step 3 cannot be performed analytically.

The last algorithm allows to sample from $X$ through the construction of an auxiliary stochastic process $Y = \{Y(t) : 0 \leq t \leq T\}$. Let $\mu = max\{q_i, i = 1, \cdots, |E|\}$ and suppose that the state changes of the process $Y$ are determined by a DTMC with transition matrix

$$R = I + \frac{1}{\mu}Q.$$

By construction, it is allowed virtual state changes in which a jump occurs but the state does not change. The stochastic process $Y$ is equivalent to the original CTMC $X$.

Conditioned on $X(0) = a$ and $X(T) = b$ the number of state changes M, including the virtual changes, is given by

$$P(M = m|X(0) = a, X(T) = b) = e^{-T\mu}\frac{(T\mu)^m}{m!}R^m_{ab}/(P(T))_{ab}. \tag{4.14}$$

**Algorithm 10** (Uniformization)**.**

1. Simulate the number of state changes $m$ from the distribution (4.14);
2. if the number of state changes is 0, we are done: $X(t) = a$, $0 \leq t \leq T$;
3. if the number of state changes is 1 and $a = b$, we are done: $X(t) = a$, $0 \leq t \leq T$;
4. if the number of state changes is 1 and $a \neq b$ simulate $t_1$ uniformly in $[0, T]$, we are done: $X(t) = a$, $t < t_1$, and $X(t) = b$, $t_1 \leq t \leq T$;
5. when the number of state changes $m$ is at least 2, simulate $m$ independent uniform random numbers in $[0, T]$ and sort the numbers in increasing order

to obtain the times of state changes $0 < t_1 < \cdots < t_m < T$. Simulate $X(t_1), \cdots, X(t_{m-1})$ from a DTMC with transition matrix R and conditional on starting state $X(0) = a$ and ending state $X(t_m) = b$. Determine which state changes are virtual and return the remaining changes and corresponding times of change.

Hobolth and Stone (2009) show that no algorithm is globally better then the other. The endpoint-conditioned sample paths can be simulated using one of the three algorithms described above. Because of the numerical approximation required by Algorithm 9, we consider only the other two algorithms.

Algorithm 8 is more efficient when: *i*) the ending state b is very likely, in a given interval; *ii*) the endpoints are the same and the time interval is shorter then the expected jump time. Otherwise, Algorithm 10 performs better.

## 4.7 Some important identifiability issues

Identifiability is an intrinsic issue when performing inference for Poisson processes. Reliable estimates require a minimum amount of information which is positively related to the number of events observed from the PP. For example, to estimate a constant rate 1, observing the process for only one time unit would not provide information good enough, whereas observing it for five time units certainly would. Also, one or two time units are sufficient to estimate well a constant rate 10. Therefore, good estimates rely on a minimum amount of information scaled with the rate and observed times.

Consider the general case where all the possible components of the model are unknown and we just assume that the rate is driven by a CTMC. There are a myriad of settings of the CTMC which fit the data reasonably well. For example, consider a trajectory with short waiting time in a given state, but if this state is very recurrent, it can be as well fitted as a trajectory with long waiting time. If the data does not have enought information, we have to provide some prior information for the Q-matrix and the state space $E$.

If we believe that the data do not have enough information to estimate the Q-matrix reasonably well, we choose the prior densities for the diagonal parameters of the Q-matrix such that the waiting times in the states are not too small neither too large, according to the time scale. It is not interesting to have a CTMC with short waiting times with respect to the scale of the data set as the data will not contain enough information for the all the changes to be captured. On the other hand, the waiting time in a given state can not be too large such that the average time spent in the remaining states is not large enough to estimate their values. One reasonable strategy is to assume the same prior for each element in the diagonal of the Q-matrix.

Adopting an uniform distribution (on the simplex) for the transition probabilities (i.e. a Dirichlet$(1, 1, \cdots, 1)$) is a reasonable strategy. To estimate these transition probabilities, all sorts of transitions in the CTMC and many of them are necessary. Otherwise, it is advisable to fix these parameters.

The state space values ought to be significantly different with respect to the Poisson distribution variance. For example, if the data have ten events in a two time units interval, it is similarly likely that the rate is 4, 5 or 6.

# Chapter 5

# Simulations and Application

In this chapter we present an analysis with simulated and real data to investigate the Bayesian methodology devised in the previous chapter. In section 5.1 we compare the sampling methods of the full conditional distribution of $\lambda$. The most efficient is chosen for the further analysis. Section 5.2 contains three different scenarios to perform inference using our methodology. In section 5.3 an analysis with real data is performed. The studies are performed using the Ox language. The reported computational time is using a CPU Intel Core i7-3770, 3.40GHz x 8.

## 5.1 Comparing RS and MH

To compare the sampling methods of the full conditional distribution of $\lambda$, we use a simulated example of MSCP in the time interval $[0, 200]$, where the CTMC has state space $E = \{1, 4, 7\}$ and Q-matrix

$$Q = \begin{bmatrix} -\frac{1}{60} & \frac{1}{120} & \frac{1}{120} \\ \frac{1}{60} & -\frac{1}{30} & \frac{1}{60} \\ \frac{1}{40} & \frac{1}{40} & -\frac{1}{20} \end{bmatrix}.$$

Therefore, the average waiting times are 60, 30 and 20, in the states 1, 4 and 7, respectively. As the aim is to compare the RS and MH alternatives, we fix the Q-matrix and the state space and estimate only the intensity function $\lambda$. Note that both algorithms are sampling from the same distribution. Therefore, the choice of a suitable algorithm is in the sense of the faster convergence and, hence, lower computational cost.

Each chain runs for 50k iterations, with a burn-in of 20k. Thus, the intensity function estimation is performed by a sample of size 30k. To obtain the estimated trajectory, we set a grid in the interval, with length 1 and the posterior mean and mode are computed for each time.

For the MH, defined in (4.5) and (4.8), we investigate the sampling method with several values for $L$ and $r$. A good estimation was obtained with $L = 10$ and $r = 0$ (which proposes from the prior CTMC), as shown in Figure 5.1. The estimated trajectory followed the true trajectory. Only in the initial interval $[0, 50]$ the estimated $\lambda$ differs a bit from the true rate, this is due to the data information in this region. A way to evaluate the convergence is by taking the empirical probability distribution of $\lambda(t)$ for each time of the grid. Two plots of these empirical distributions along the chain can be seen in Figures A.1 and A.2 in Appendix A. Some estimates via MH with different specifications for $L$ and $r$ are also in Appendix A. All the studies were done with 10 MH updates for each Gibbs iteration.

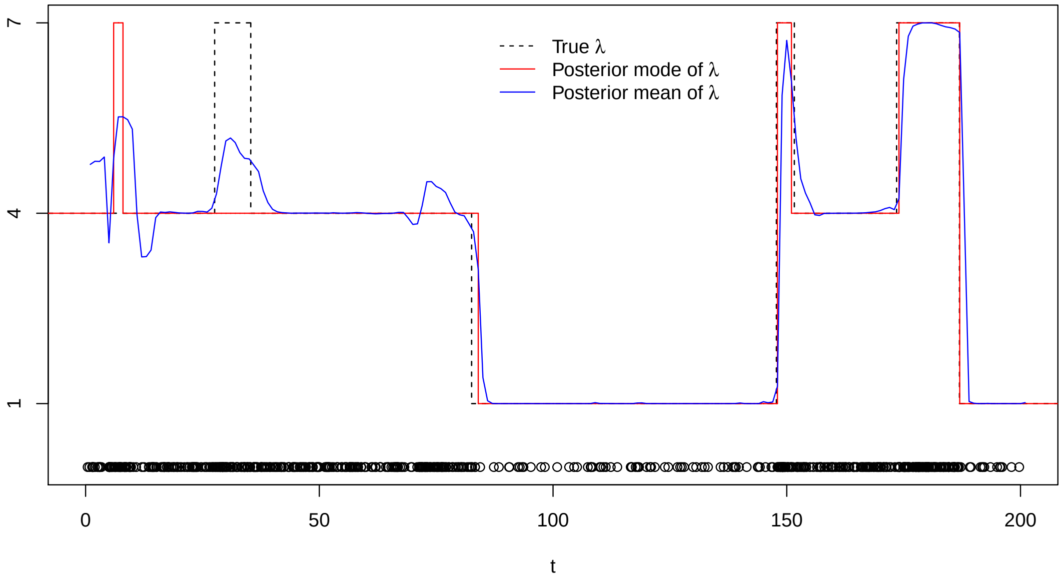

Figure 5.1: Estimation of $\lambda$ via MH: $L = 10$ and $r = 0$. The lines are the intensity functions, true and estimated, and the points are the observed data.

For the RS, defined in (4.5) and (4.7), the $L$ was fixed equal to 0.5. We tested other values but larger $L$ leads with a high computational cost. The choice of $L = 0.5$ is plausible for the data, which provides a minimum acceptance probability

$$\begin{cases} 0.0498 \prod_{i=1}^{n(k)} \left(\frac{m_E}{M_E}\right)^{1-r}, & \text{if } r \leq 1, \\ 0.0498 \prod_{i=1}^{n(k)} \left(\frac{M_E}{m_E}\right)^{1-r}, & \text{if } r > 1. \end{cases}$$

The 50k MCMC iterations were still not sufficient for a reasonable estimation of $\lambda$ via RS, independent of the $r$ value. Figure 5.2 presents the estimates with $r = 1.8$. This scenario was the one that presented the estimated trajectory closer to the true one, however, convergence was not yet achieved. This can be seen in Figure B.1 in Appendix B. The RS convergence has a high computational cost and the algorithm needs to run a long time to converge.

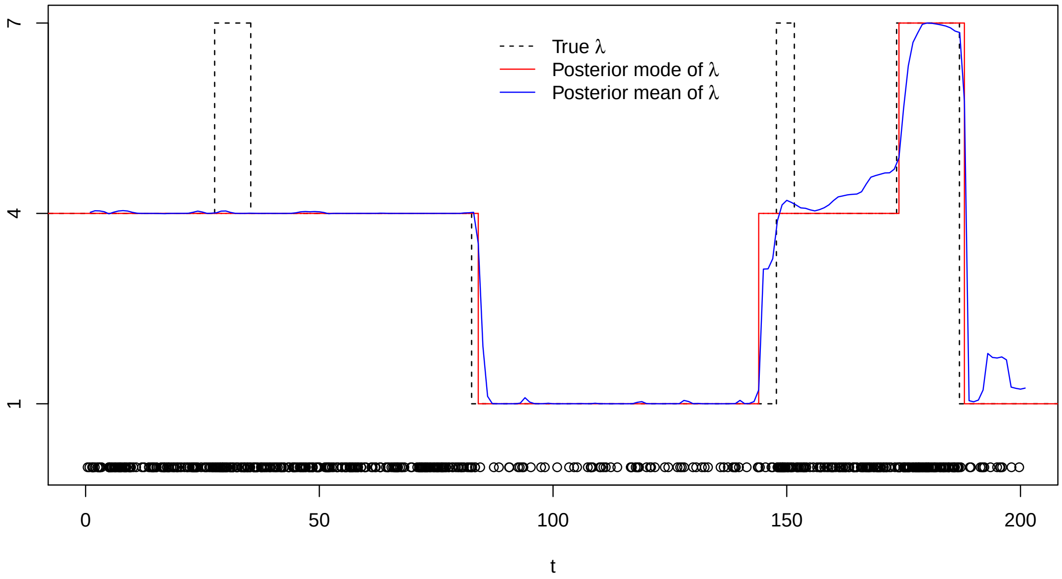

Figure 5.2: Estimation of $\lambda$ via RS: $L = 0.5$ and $r = 1.8$. The lines are the intensity functions, true and estimated, and the points are the observed data.

Inference with other $r$ values are in Appendix B. Note that as the value of $r$ increases, the greater is the chance to propose a trajectory of the CTMC with large state values. However, the acceptance probability is greatly penalised when the proposed trajectory is at the higher states. Choosing an appropriate value for $r$ in the RS algorithm is a challenging problem. Therefore, the Metropolis-Hastings algorithm will be chosen to sample from the full conditional distribution of $\lambda$ for the next analysis.

## 5.2 Simulated studies

We construct three scenarios to perform inference by sampling from the posterior distribution given in (4.1). Table 5.1 contains the specifications of the simulated data to be used and the algorithm's specifications to perform inference.

| | Time interval | E | Q-matrix | $L$ | $r$ | MH updates |
|---|---|---|---|---|---|---|
| Scenario 1 | $[0,100]$ | $\{0,1\}$ | $\begin{bmatrix} -\frac{1}{5} & \frac{1}{5} \\ \frac{1}{10} & -\frac{1}{10} \end{bmatrix}$ | 20 | 0 | 10 |
| Scenario 2 | $[0,100]$ | $\{1,5\}$ | $\begin{bmatrix} -\frac{1}{20} & \frac{1}{20} \\ \frac{1}{10} & -\frac{1}{10} \end{bmatrix}$ | 20 | 0 | 10 |
| Scenario 3 | $[0,200]$ | $\{1,4,7\}$ | $\begin{bmatrix} -\frac{1}{60} & \frac{1}{120} & \frac{1}{120} \\ \frac{1}{60} & -\frac{1}{30} & \frac{1}{60} \\ \frac{1}{40} & \frac{1}{40} & -\frac{1}{20} \end{bmatrix}$ | 10 | 0 | 10 |

Table 5.1: Scenario's specifications to perform inference.

In the cases where the process has a state zero, this state is fixed. Scenario 1 is an example of this, where the aim is to estimate the positive state. The results for scenario 1 are for a 20k MCMC iterations and a burn-in of 5k. A previous study to investigate prior sensitivity indicated that reasonable results are obtained without the need of very informative priors. Then, we set Gamma(1,1) as the prior distribution for the state $\epsilon_2$ and Gamma(1, 2) as the prior distribution for $\theta_1$ and $\theta_2$. Table 5.2 contains the estimates for $\theta$ and $E$.

| Parameter | True value | Estimated value | Interval |
|---|---|---|---|
| $-\theta_1$ | $-0.2$ | $-0.179$ | $(-0.350, -0.064)$ |
| $-\theta_2$ | $-0.1$ | $-0.124$ | $(-0.245, -0.046)$ |
| $\epsilon_2$ | 1 | 0.988 | $(0.720, 1.300)$ |

Table 5.2: Results for scenario 1: posterior mean and a 95% credibility interval.

Table 5.2 shows that all the parameters were well estimated. The point estimates are close to the true values and the credibility interval includes them. The trace plots are in Appendix C.

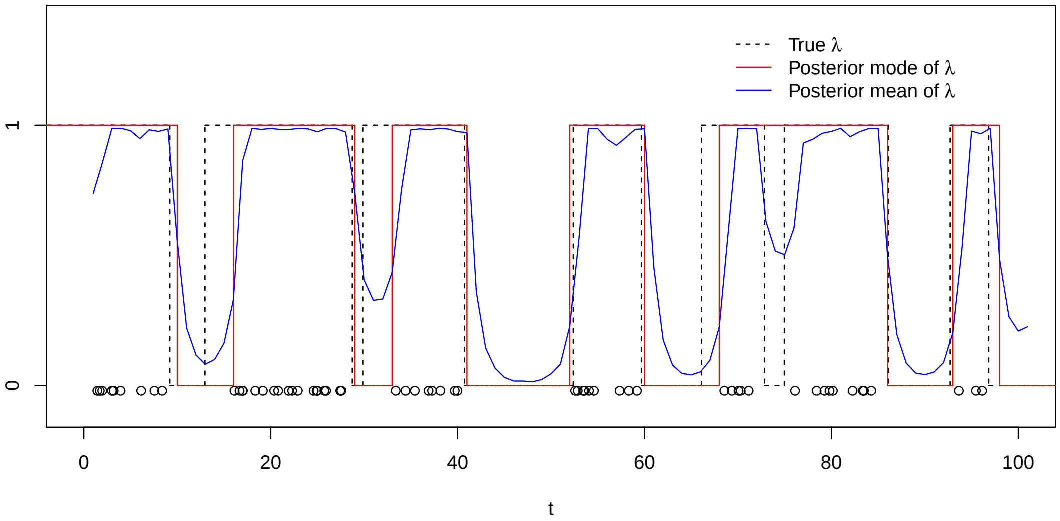

Figure 5.3: Estimation of $\lambda$ in scenario 1.

Note that in each MCMC iteration we have a different value for the states. To obtain an estimated trajectory of $\lambda$ by posterior mode, we compute, in each iteration, the position of the current $\lambda(t)$ in the vector $E$, i.e., if the $\lambda(t)$ is the first or second state in $E$. Then, we get the posterior mode, by positions. Figure 5.3 shows that the estimations of $\lambda$ are in agreement with the true $\lambda$ and, mainly, with the data.

For scenario 2, a 20k MCMC iterations were run and a burn-in of 5k. We chose Gamma$(1, 1)$ as the prior distribution for each state and Gamma$(1, 2)$ as the prior distribution for $\theta_1$ and $\theta_2$. Table 5.3 shows that these parameters were well estimated. The trace plots for each parameter are in Appendix C.

| Parameter | True value | Estimated value | Interval |
|---|---|---|---|
| $-\theta_1$ | $-0.05$ | $-0.066$ | $(-0.149, -0.017)$ |
| $-\theta_2$ | $-0.1$ | $-0.128$ | $(-0.278, -0.034)$ |
| $\epsilon_1$ | 1 | 0.929 | $(0.696, 1.191)$ |
| $\epsilon_2$ | 5 | 4.920 | $(4.164, 5.751)$ |

Table 5.3: Results for scenario 2: posterior mean and a 95% credibility interval.

Also, for this scenario, we obtain good estimations of $\lambda$. They are in agreement with the true $\lambda$ and with the data, as it can be seen in Figure 5.4.

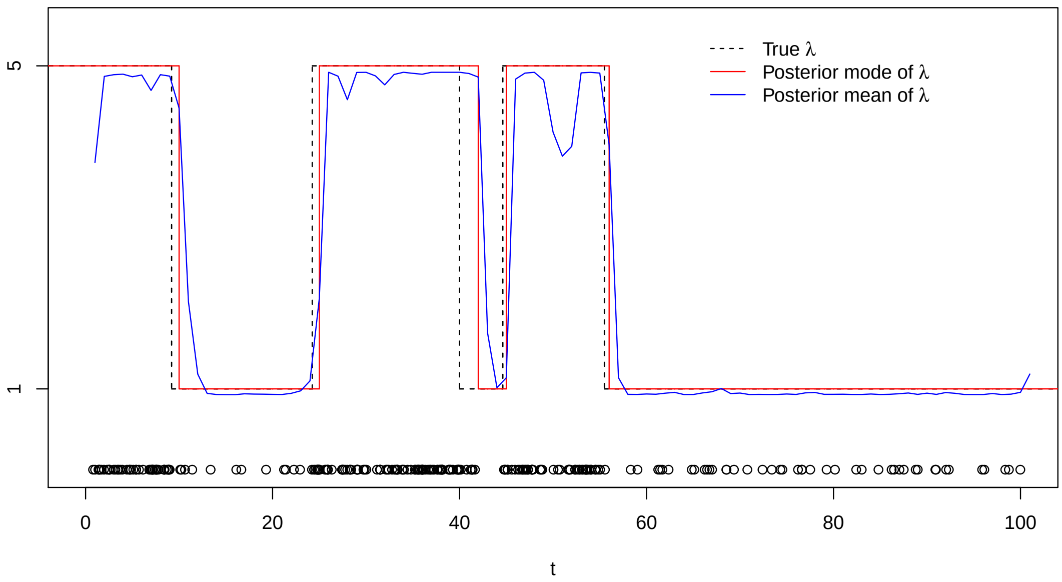

Figure 5.4: Estimation of $\lambda$ in scenario 2.

Scenario 3 is the same that is in section 5.1. However, to generate from the posterior distribution with all unknown parameters, 30k MCMC iterations were run and a burn-in of 10k. We used Gamma$(1, 1)$ as the prior distribution for each state, Gamma$(1, 2)$ as the prior distribution for each diagonal parameter of the Q-matrix and Beta$(1, 1)$ for each probability distribution of the Q-matrix. Note that for a 3-state CTMC, the probability vector of the transition distribution has size 2, thus, it is sufficient to estimate only one probability $p$ and the other is $(1 - p)$.

We observed that the point estimates of $\theta_{\bullet}$ were not as good as in the other scenarios, by Table 5.4. All $\theta_i$'s were overestimated, consequently, after the transformation, the diagonal parameters of the Q-matrix were underestimated. Although, all the credibility intervals contain their true values. By the credibility intervals in Table 5.4 and Figure C.6 in Appendix C, the $\theta_{j\bullet}$'s were estimated with too large variance.

All the $E$ parameters were well estimated. The pontual estimates are close to the true values and the credibility intervals include them. The trace plots are in Appendix C.

| Parameter | True value | Estimated value | Interval |
|---|---|---|---|
| $-\theta_1$ | $-0.017$ | $-0.030$ | $(-0.081, -0.005)$ |
| $\theta_{12}$ | 0.5 | 0.388 | $(0.017, 0.907)$ |
| $\theta_{13}$ | 0.5 | 0.612 | – |
| $\theta_{21}$ | 0.5 | 0.437 | $(0.060, 0.877)$ |
| $-\theta_2$ | $-0.033$ | $-0.047$ | $(-0.112, -0.010)$ |
| $\theta_{23}$ | 0.5 | 0.563 | – |
| $\theta_{31}$ | 0.5 | 0.434 | $(0.060, 0.881)$ |
| $\theta_{32}$ | 0.5 | 0.566 | – |
| $-\theta_3$ | $-0.050$ | $-0.164$ | $(-0.383, -0.040)$ |
| $\epsilon_1$ | 1 | 1.044 | $(0.823, 1.293)$ |
| $\epsilon_2$ | 4 | 3.946 | $(3.360, 4.473)$ |
| $\epsilon_3$ | 7 | 6.755 | $(5.367, 8.303)$ |

Table 5.4: Results for scenario 3: posterior mean and a 95% credibility interval.

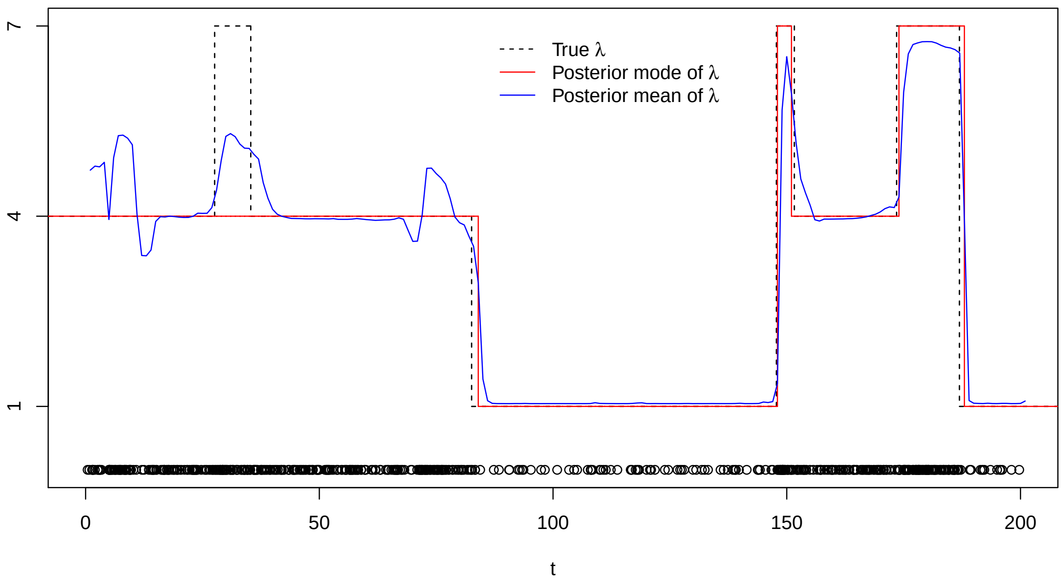

Figure 5.5: Estimation of $\lambda$ in scenario 3.

Although the Q-matrix parameters were not precisely estimated, it did not affect the $\lambda$ estimate. The estimations in Figure 5.5 are good as in Figure 5.1, where all parameters were fixed and only the intensity function was estimated.

We did many simulations for the three scenarios, where we fixed $\theta$ and $E$, or only $E$, or only $\theta$. But there was no significant difference from when we treated all parameters as unknown.

In addition to estimating the parameters involved in the proposed process, we may do other estimates. It is possible, for example, to estimate the expected number of events in an interval. Let $Y((t_1, t_2])$ be the number of events in the interval $[t_1, t_2]$, then

$$\mathbb{E}\left(\mathbb{E}\left(Y((t_1, t_2])|\lambda, y\right)\right) = \mathbb{E}\left(\int_{t_1}^{t_2} \lambda(s)ds \middle| y\right).$$

The Monte Carlo estimator can be obtained by

$$\frac{1}{M}\sum_{i=1}^{M}\int_{t_1}^{t_2} \lambda^{(i)}(s)ds,$$

where M is the number of MCMC iterations (after burn-in) and $\lambda^{(i)}$ is the intensity function generated at iteration $i$, according to the current $E$ and $\theta$. Table 5.5 presents the Monte Carlo estimates for the expected number of events in a given interval for the three scenarios discussed above.

| | Time interval | Estimate |
|---|---|---|
| Scenario 1 | $[100, 125]$ | 22.672 |
| Scenario 2 | $[100, 125]$ | 140.662 |
| Scenario 3 | $[200, 250]$ | 380.661 |

Table 5.5: Monte Carlo estimates for the expected number of events in a given interval.

According to Table 5.5, for scenario 1, if we observe the process in the time interval $[100, 125]$ we expect 22.672 event occurrences. For this prediction, we observe the process at time interval $[0, T]$ and assume that $\lambda$ will have the same distribution after $T$.

The computational time for the simulated studies was about 10 minutes, 30 minutes and 2 hours for scenarios 1, 2 and 3, respectively.

## 5.3 Application

We are interested in modelling the occurrences of traffic accidents along the BR-381 highway. Figure 5.6 presents the highway map. This Brazilian highway goes through the states of Espírito Santo, Minas Gerais and São Paulo and is approximatly 1,180 kilometers long. We do not have information on the highway in Espírito Santo, therefore, the initial kilometer mark considered is 136.0 (in Mantena city, Minas Gerais).

The data refers to all the 11,324 accidents in 2011. This data is available in the Brazilian traffic department (DNIT) website (http://www.dnit.gov.br/).

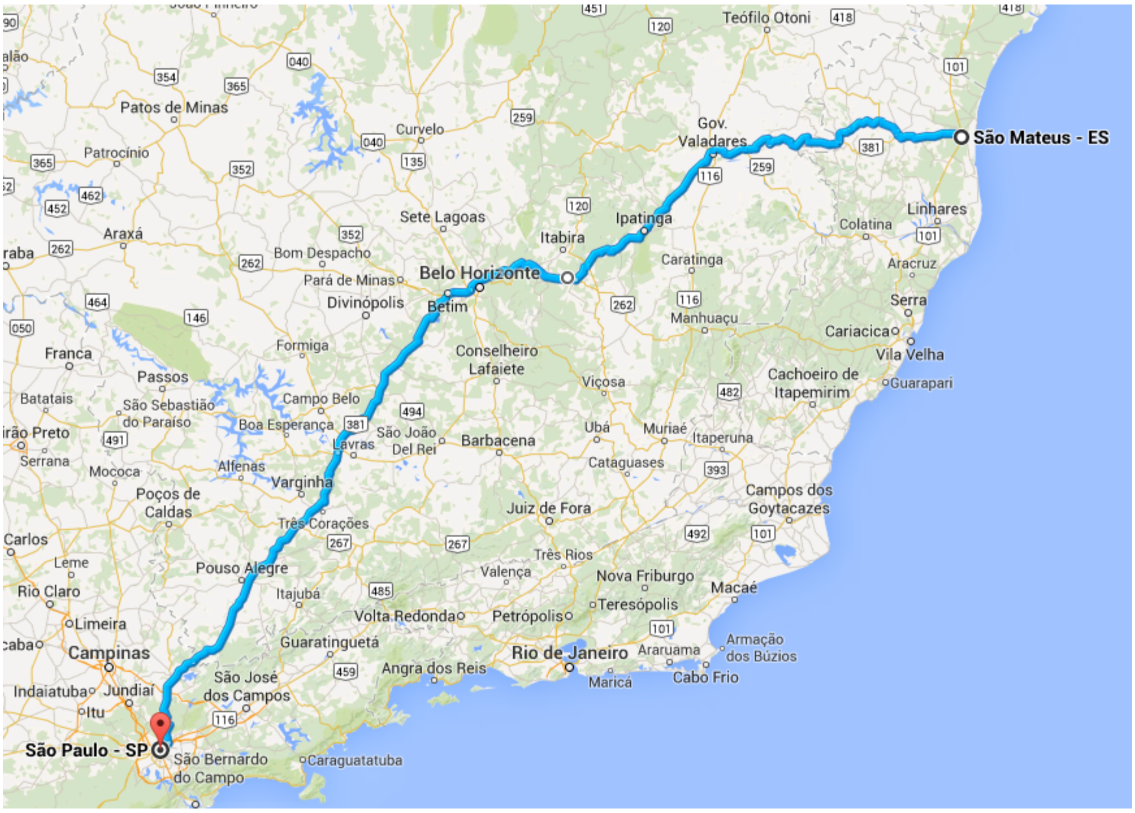

Figure 5.6: BR-381 highway map.

We adopt the scale of 100 meters per unit and consider a four state space for the intensity function. Table 5.6 shows the parameter estimates. The estimated values of the intensity function are: 0.012, 0.506, 1.678 and 14.320. The trace plots for each parameter are in Appendix D. The computational time for this analysis was about 50 hours.

Figure 5.7 shows the intensity function estimation, by posterior mode and mean. The colorful rectangles are to identify the stretches that present high estimated levels of accidents. The first rectangle is the Mantena stretch, the second is the stretch between João Monlevade and Sabará, the third is the stretch in Belo Horizonte (state capital), Contagem and Betim, and the other ones are Itatiaiuçu, Santo Antônio do Amparo, Camanducaia and São Paulo city, respectively.

| Parameter | Estimated value | Interval |
|---|---|---|
| $-\theta_1$ | $-0.031$ | $[-0.039; -0.024]$ |
| $\theta_{12}$ | $0.366$ | $[0.242; 0.506]$ |
| $\theta_{13}$ | $0.292$ | $[0.178; 0.421]$ |
| $\theta_{14}$ | $0.341$ | – |
| $\theta_{21}$ | $0.154$ | $[0.099; 0.216]$ |
| $-\theta_2$ | $-0.040$ | $[-0.046; -0.034]$ |
| $\theta_{23}$ | $0.323$ | $[0.252; 0.397]$ |
| $\theta_{24}$ | $0.524$ | – |
| $\theta_{31}$ | $0.110$ | $[0.056; 0.176]$ |
| $\theta_{32}$ | $0.446$ | $[0.356; 0.533]$ |
| $-\theta_3$ | $-0.093$ | $[-0.111; -0.077]$ |
| $\theta_{34}$ | $0.443$ | – |
| $\theta_{41}$ | $0.132$ | $[0.087; 0.183]$ |
| $\theta_{42}$ | $0.496$ | $[0.422; 0.571]$ |
| $\theta_{43}$ | $0.372$ | – |
| $-\theta_4$ | $-0.608$ | $[-0.696; -0.524]$ |
| $\epsilon_1$ | $0.012$ | $[0.006; 0.020]$ |
| $\epsilon_2$ | $0.506$ | $[0.479; 0.533]$ |
| $\epsilon_3$ | $1.678$ | $[1.586; 1.783]$ |
| $\epsilon_4$ | $14.320$ | $[13.857; 14.819]$ |

Table 5.6: Results for traffic accidents: posterior mean and a 95% credibility interval.

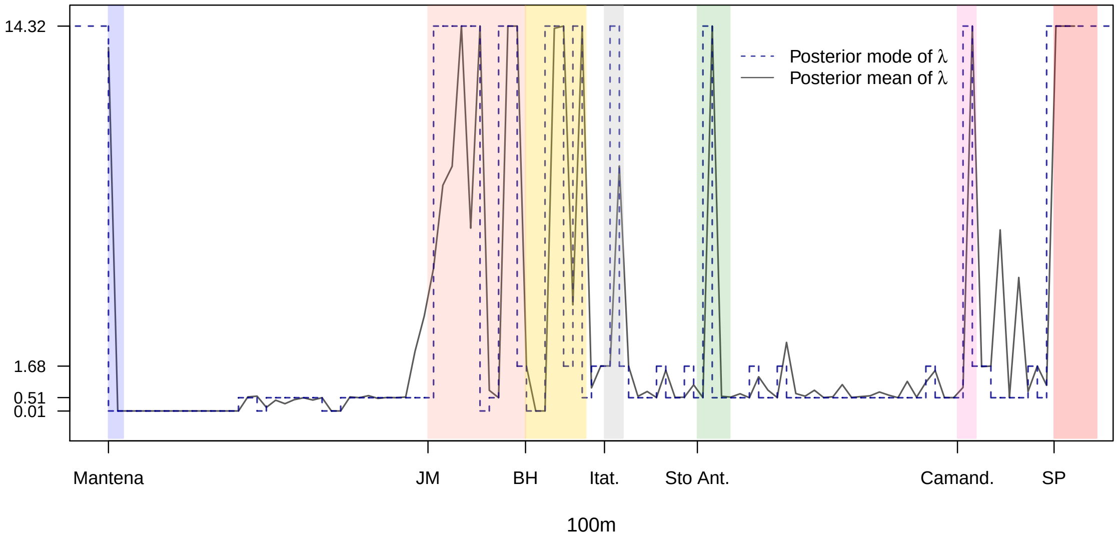

Figure 5.7: Estimation of $\lambda$ for traffic accidents.

# Chapter 6

# Conclusions and future work

Statistical modelling of point patterns is an important and common problem in several applications. An important point process, and a generalisation of the Poisson process, is the Cox process, where its intensity function is itself stochastic. There are many possibilities to describe the dynamics of the intensity function and we chose a class in which it is a continuous-time Markov chain, which we refer these processes as a Markov switching Cox process. Also, the use of this dynamics means that the process alternates between different homogeneous Poisson processes.

Some probabilistic properties for the MSCP's were investigated and three new theorems were derived.

We developed a Bayesian methodology, in which the inference is based on the posterior distribution. The great advantage to use a CTMC as the intensity function is that the likelihood function is tractable which facilitates the development of an exact methodology.

From simulated studies, we had good estimates using our methodology, therefore, our sampling method is effective to perform exact inference for the MSCP's intensity function and the parameters indexing its law. Also, the simulated studies allowed us to evaluate two different algorithms for the full conditional distribution of the intensity function, they are: rejection sampler and Metropolis-Hastings. According to the simulations, in both, the choice of their specifications are crucial for the timely algorithm's convergence and this is not easy to do. For this work, we opted for the MH as the better choice.

For future work, we intend to improve the sampling methods of the full conditional distribution of $\lambda$, the ideal block sizes and number of Metropolis-Hastings updates will be investigated. Also, we intend to find new alternatives to describe the dynamics of the MSCP's intensity function.

# Appendix A

In this appendix we show some $\lambda$ sampling results via MH defined in (4.5) and (4.8).

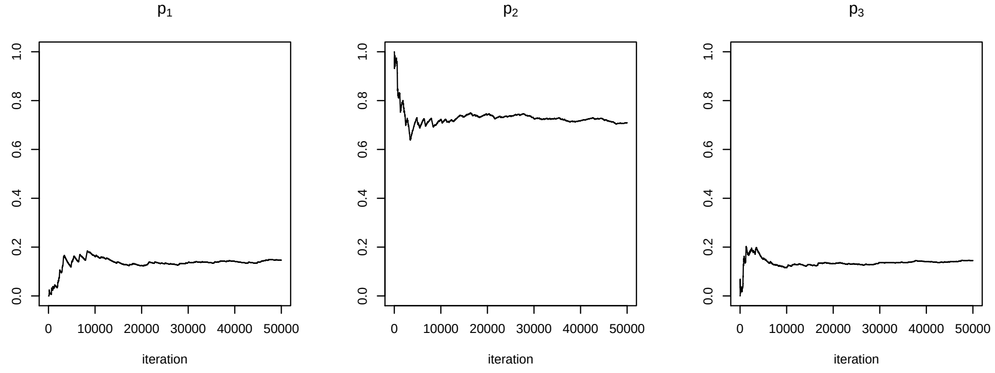

Figure A.1: Empirical probability distribution for $\lambda(10)$. Via MH: $L = 10$ and $r = 0$.

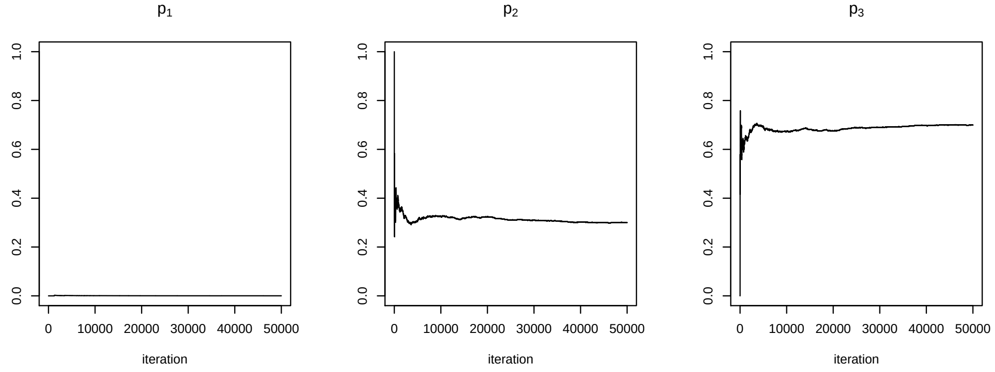

Figure A.2: Empirical probability distribution for $\lambda(150)$. Via MH: $L = 10$ and $r = 0$.

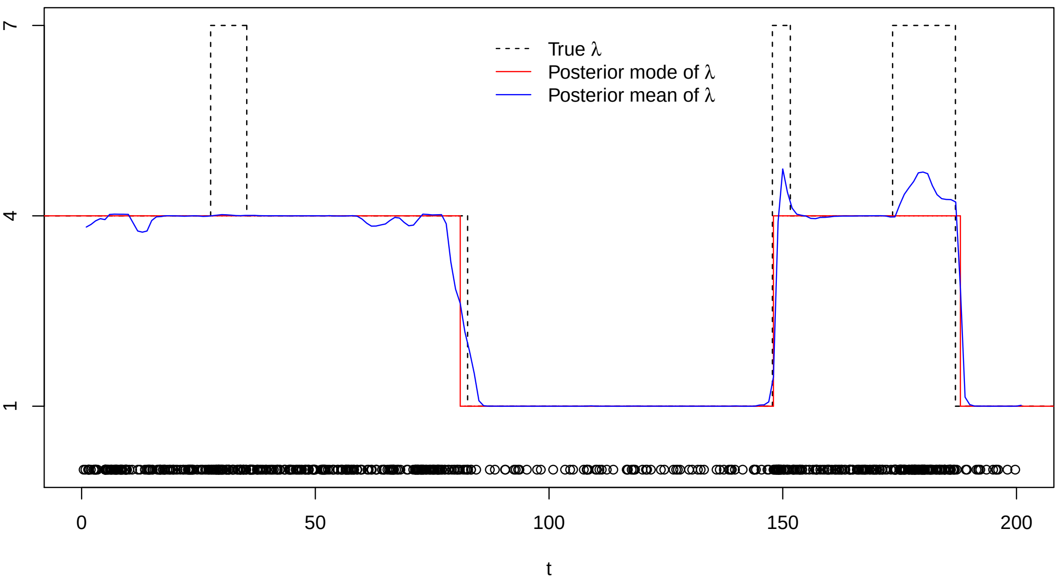

Figure A.3: Estimations of $\lambda$ via MH: $L = 10$ and $r = 0.3$.

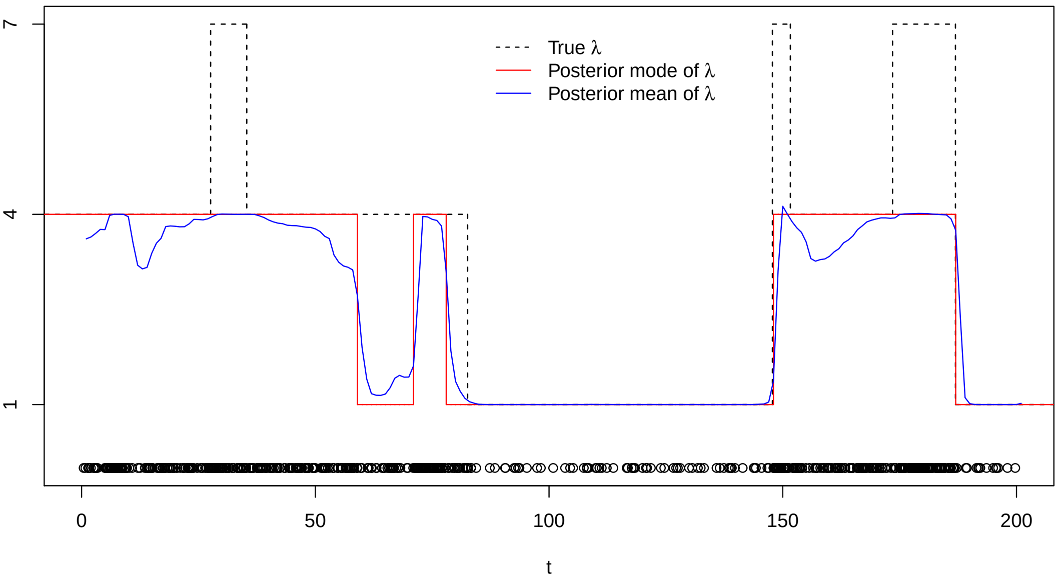

Figure A.4: Estimations of $\lambda$ via MH: $L = 10$ and $r = 0.5$.

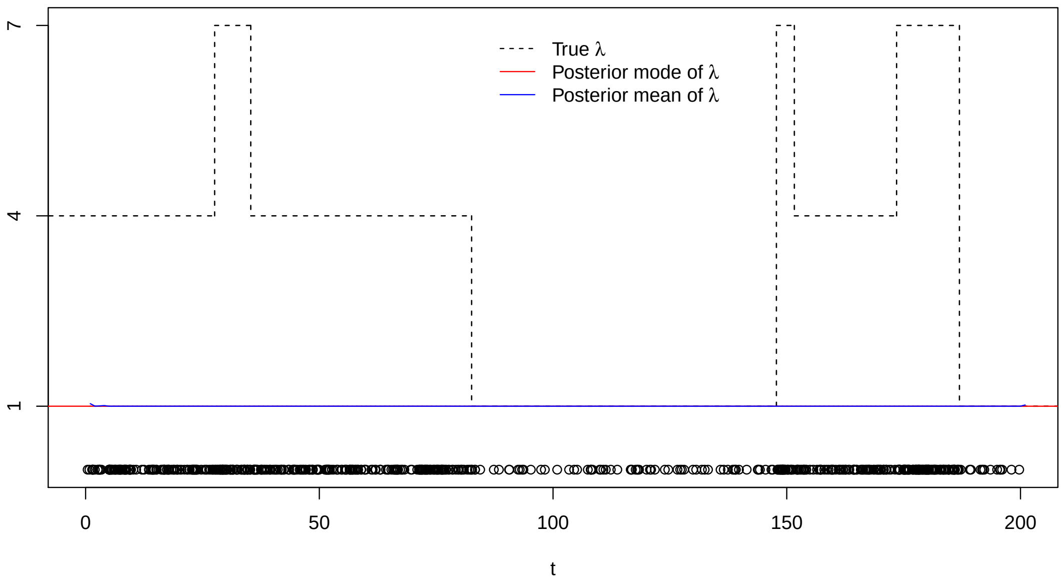

Figure A.5: Estimations of $\lambda$ via MH: $L = 10$ and $r = 1$.

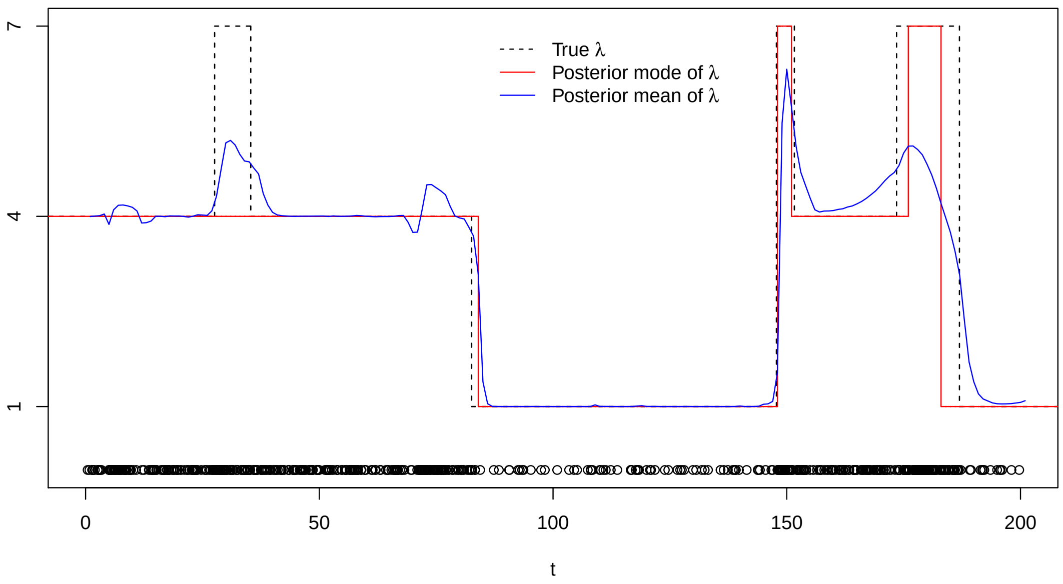

Figure A.6: Estimations of $\lambda$ via MH: $L = 30$ and $r = 0$.

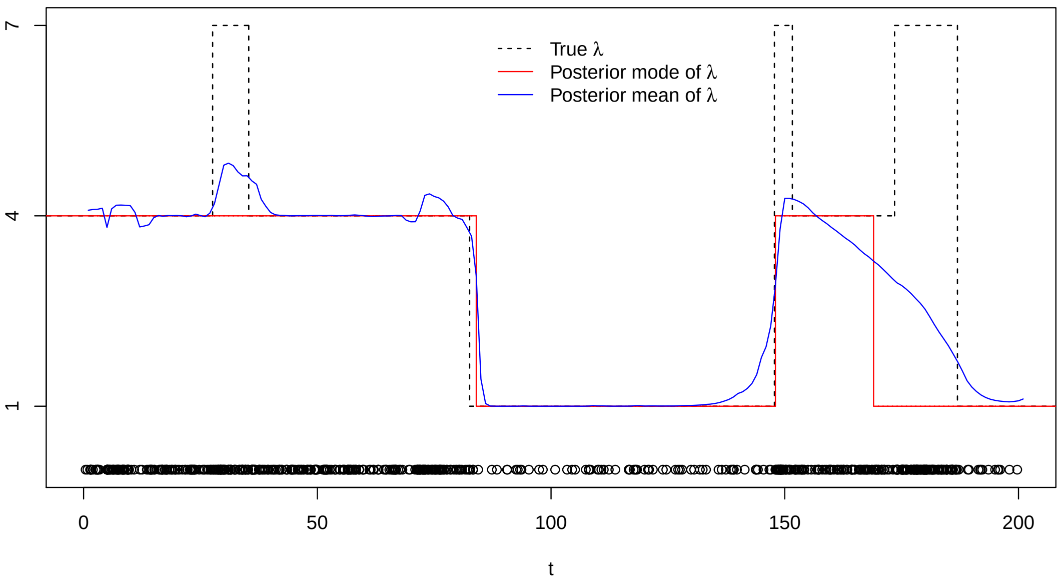

Figure A.7: Estimations of $\lambda$ via MH: $L = 50$ and $r = 0$.

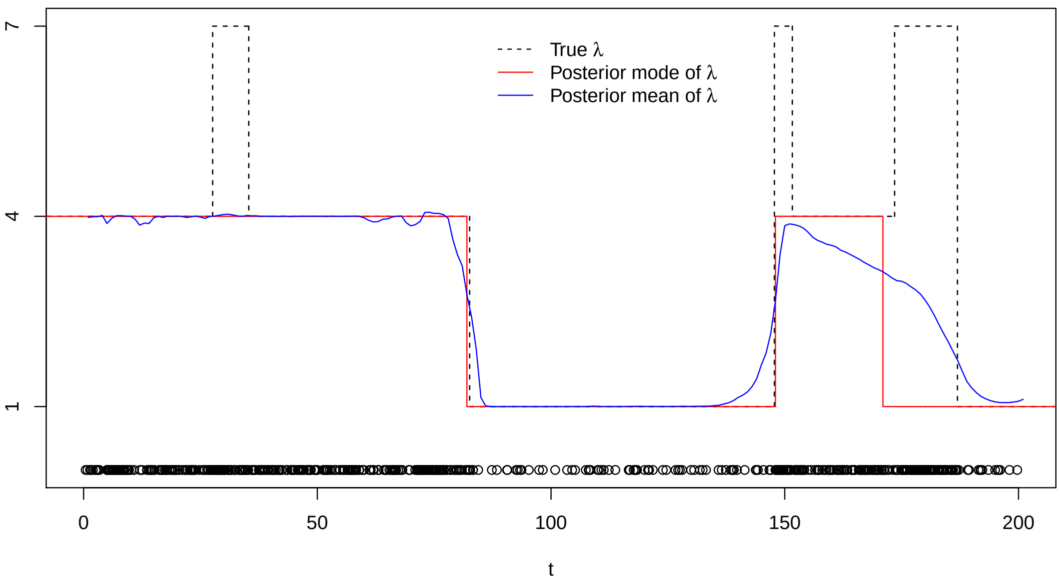

Figure A.8: Estimations of $\lambda$ via MH: $L = 50$ and $r = 0.2$.

# Appendix B

In this appendix we show some $\lambda$ sampling results via RS defined in (4.5) and (4.7).

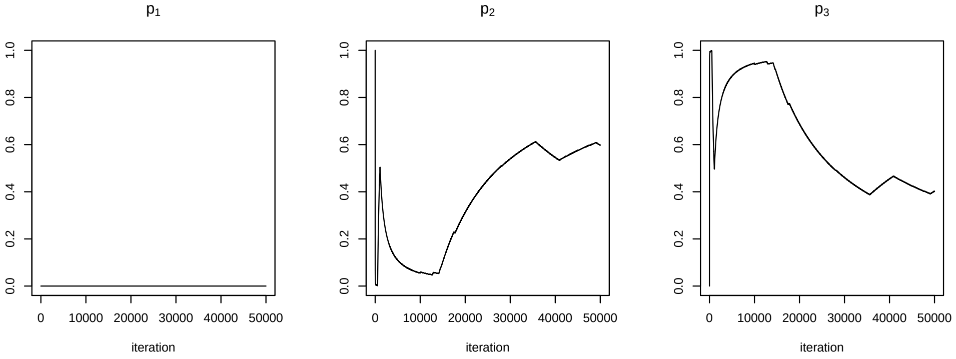

Figure B.1: Empirical probability distribution for $\lambda(170)$. Via RS: $L = 0.5$ and $r = 1.8$.

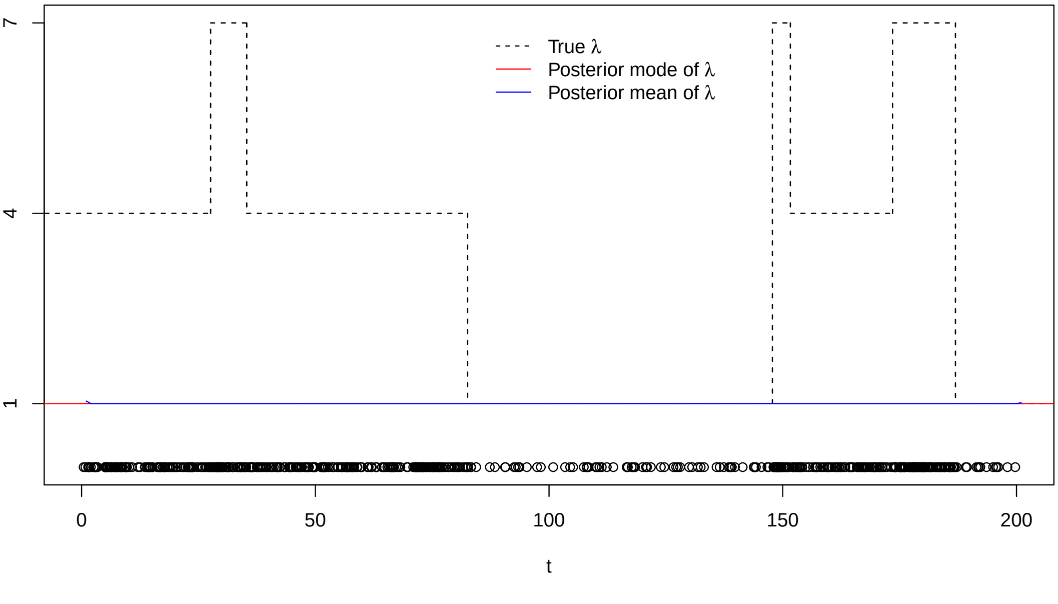

Figure B.2: Estimations of $\lambda$ via RS: $L = 0.5$ and $r = 0$.

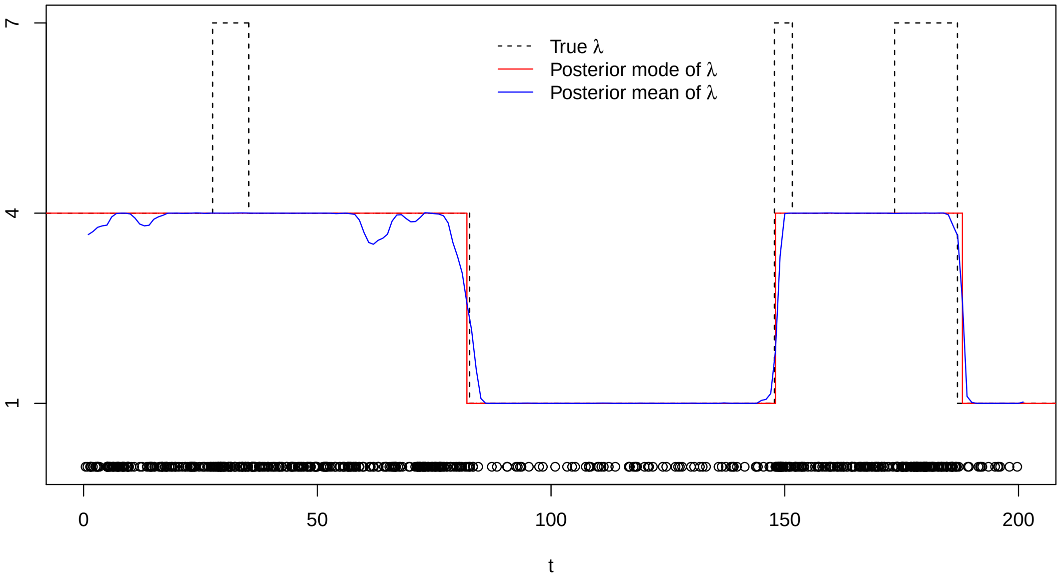

Figure B.3: Estimations of $\lambda$ via RS: $L = 0.5$ and $r = 1$.

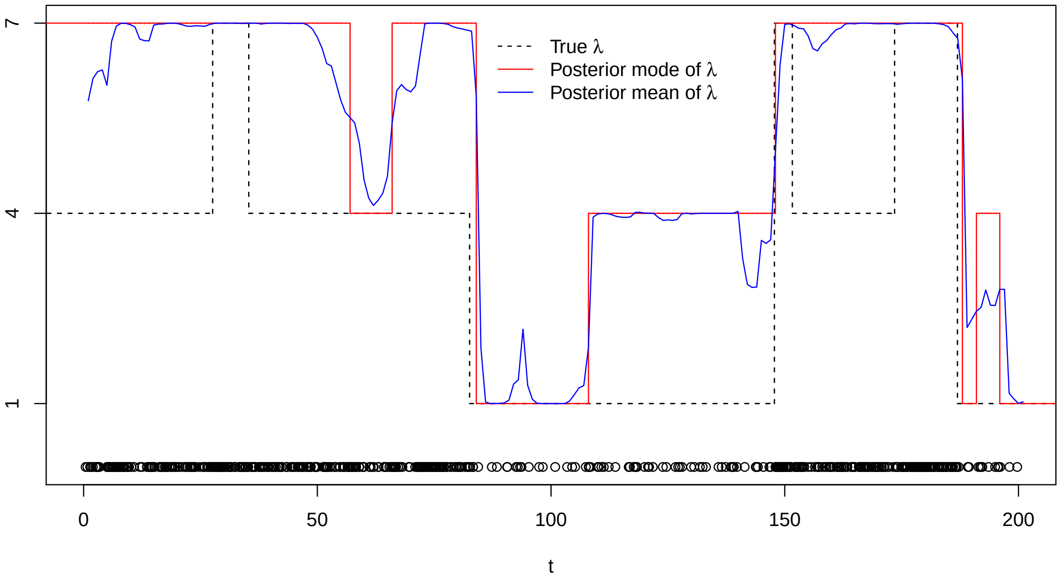

Figure B.4: Estimations of $\lambda$ via RS: $L = 0.5$ and $r = 2.3$.

# Appendix C

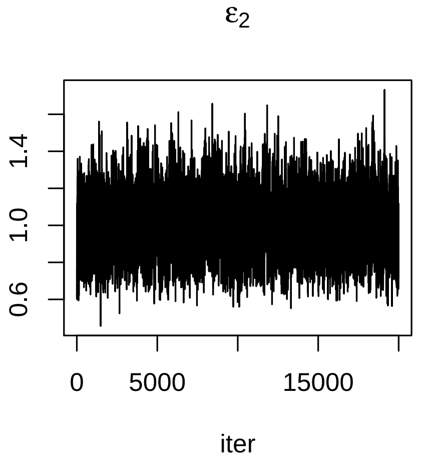

Figure C.1: Trace plot for $\epsilon_2$ in scenario 1.

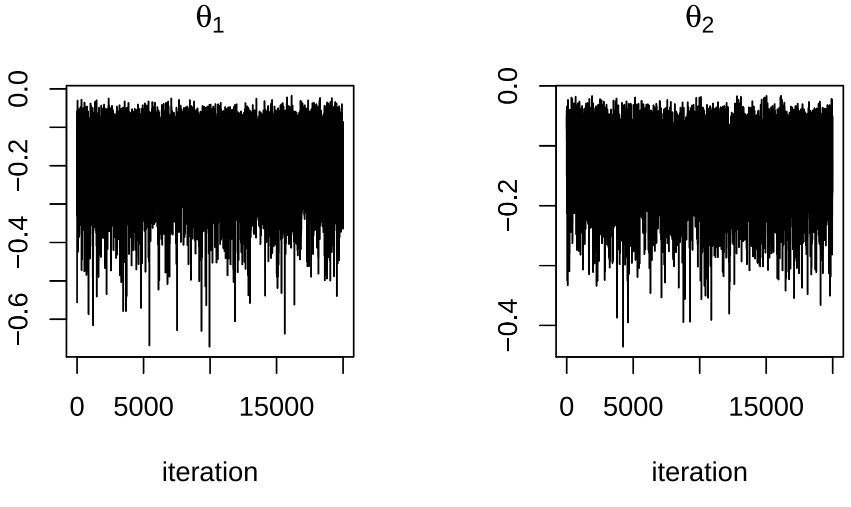

Figure C.2: Trace plots for $\theta$ in scenario 1.

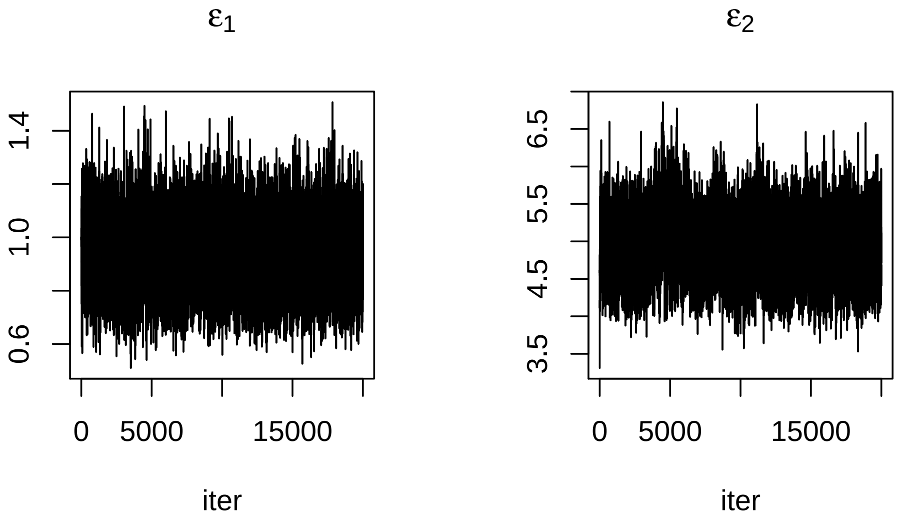

Figure C.3: Trace plots for $E$ in scenario 2.

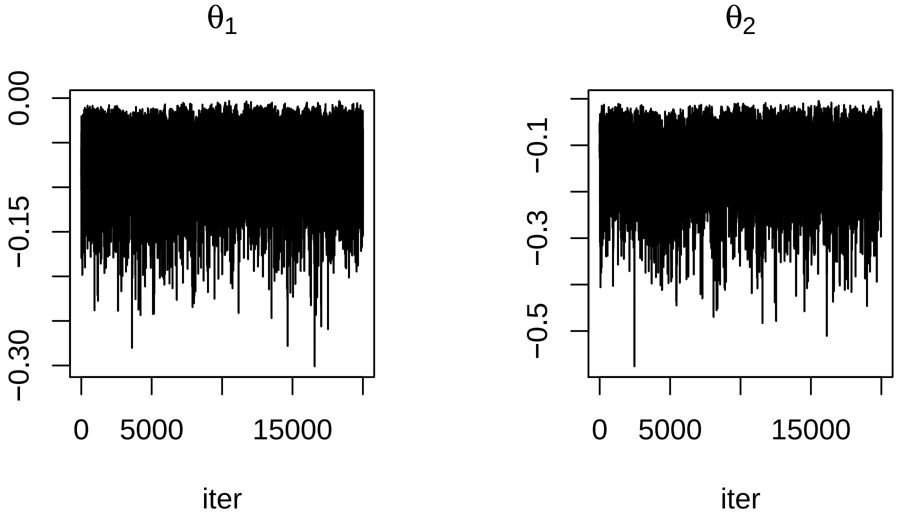

Figure C.4: Trace plots for $\theta$ in scenario 2.

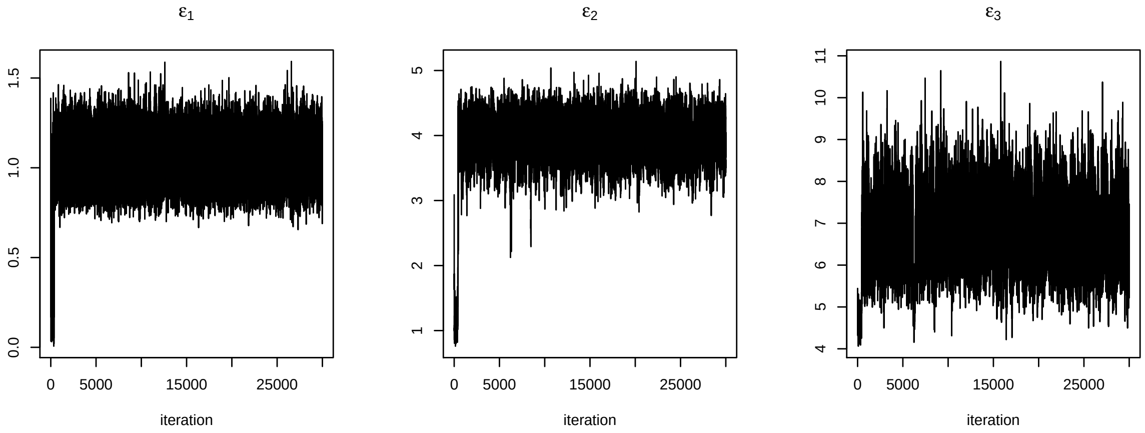

Figure C.5: Trace plots for $E$ in scenario 3.

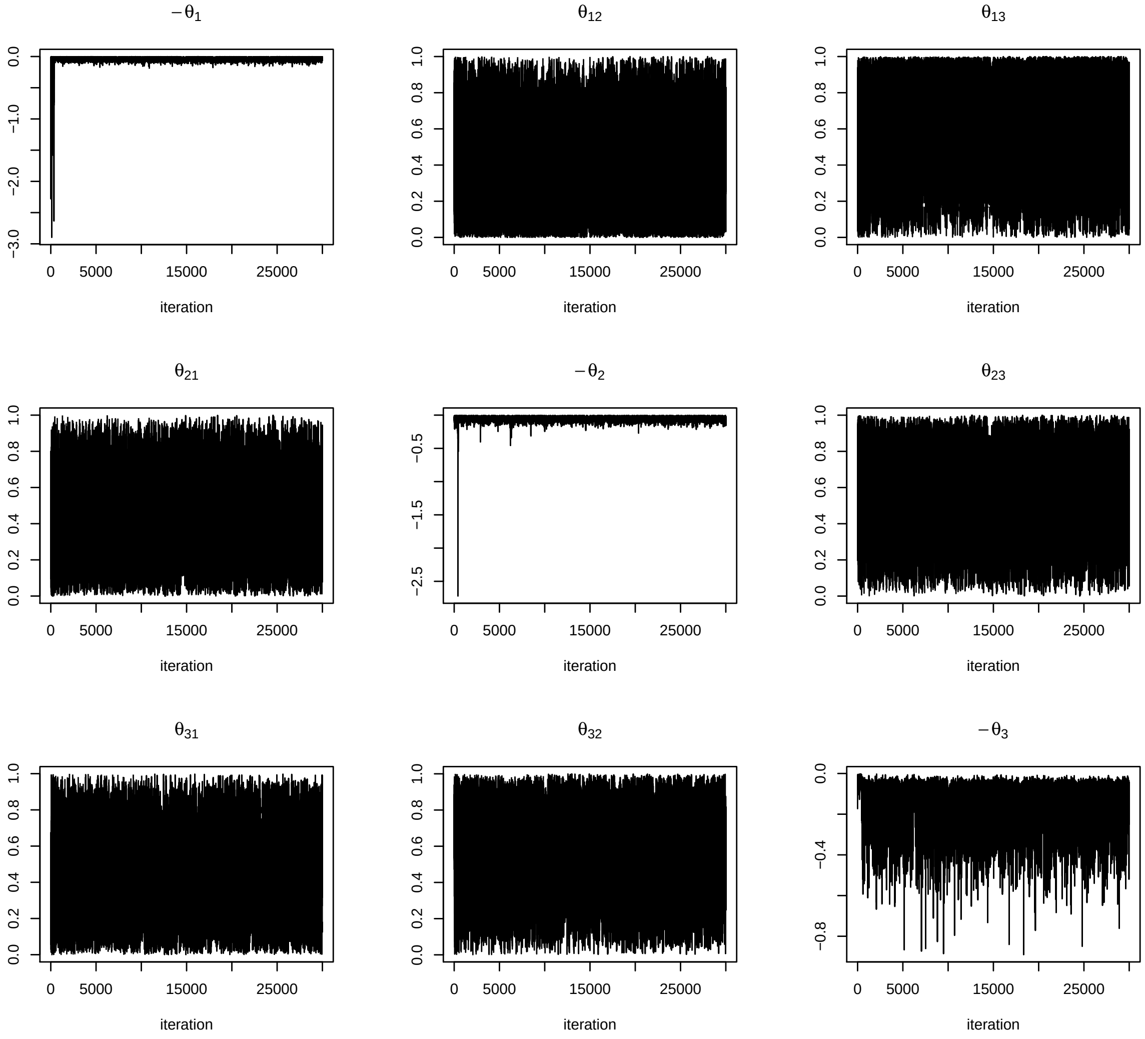

Figure C.6: Trace plots for $\theta$ in scenario 3.

# Appendix D

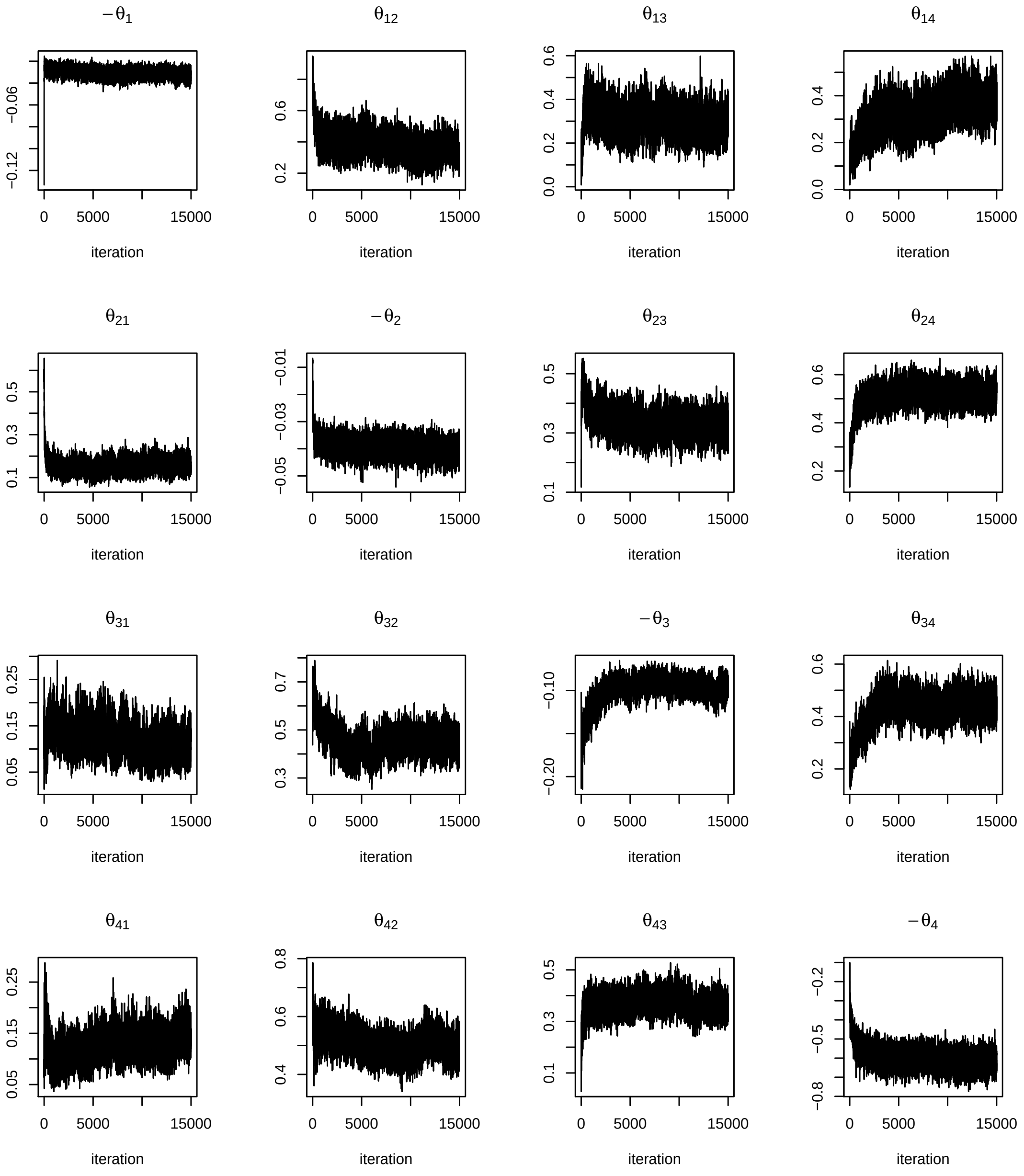

Figure D.1: Trace plots for $\theta$ in traffic accidents.

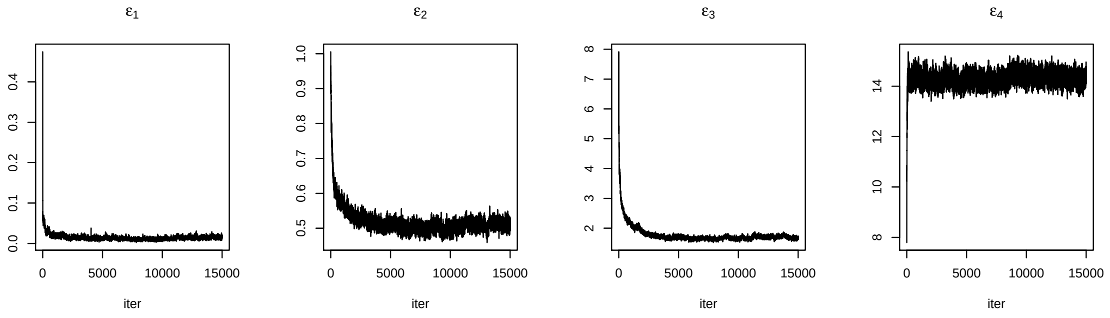

Figure D.2: Trace plots for $E$ in traffic accidents.

# Bibliography

[1] Amarasingham, A., Chen, T. L., Geman, S., Harrison, M. T. and Sheinberg, D. L. (2006) Spike count reliability and the Poisson hypothesis. *The Journal of Neuroscience.* **26**, 801-809.

[2] Basu, S. and Dassios, A. (2002) A Cox process with log-normal intensity. *Insurance: mathematics and economics.* **31**, 297-302.

[3] Brémaud, P. (1981) *Point Processes and Queues: Martingale Dynamics.* New York: Springer-Verlag.

[4] Cox, D.R. (1955) Some statistical methods connected with series of events. *Journal of the Royal Statistical Society, Series B.* **17**, 129-164.

[5] Cox, D. R. and Lewis, P. A. W. (1966) *The Statistical Analysis of Series of Events.* London: Methuen.

[6] Cunningham, J., Yu, B., Shenoy, K. and Sahani, M. (2008) Inferring neural firing rates from spike trains using Gaussian processes. *Advances in Neural Information Processing Systems.* **20**, 329-336.

[7] Daley, D. J. and Vere-Jones, D. (2003) *An Introduction to the Theory of Point Processes.* Volume I: Elementary Theory and Methods. 2 ed. New York: Springer-Verlag.

[8] Dassios, A. and Jang, J. (2003) Pricing of catastrophe reinsurance e derivatives using the Cox process with shot noise intensity. *Finance and Stochastics.* **7**, 73-95

[9] Gamerman, D. and Migon, H. S. (1999) *Statistical Inference: an integrated approach.* London: Arnold.

[10] Gonçalves, F. B. and Gamerman, D. (2015). Exact Bayesian inference in spatiotemporal Cox processes driven by multivariate Gaussian processes. In preparation.

[11] Hobolth, A. and Stone, E. A. (2009) Simulation from endpoint-conditioned, continuous-time Markov chains on a finite state space, with applications to molecular evolution. *The Annals of Applied Statistics.* **3**, 1024-1231.

[12] James, B. R. (2011) *Probabilidade: um curso em nível intermediário.* 3 ed. Rio de Janeiro: IMPA.

[13] Lando, D. (1998) On Cox processes and credit risky securities. *Review of Derivatives Research.* **2**, 99-120.

[14] Lewis, P. A. W. and Shedler, G. S. (1979) Simulation of nonhomogeneous Poisson process by thinning. *Naval Research Logistics Quarterly.* **26**, 403-413.

[15] Mengersen, K. L. and Tweedie, R. L. (1996) Rates of convergence of the Hastings and Metropolis algorithms. *Annals of Statistics.* **24**, 101-121.

[16] Møller, J., Syversveen, A. R. and Waagepetersen, R. P. (1998) Log Gaussian Cox processes. *Scandinavian Journal of Statistics.* **25**, 451-482.

[17] Møller, J. and Waagepetersen, R. P. (2003) *Statistical Inference and Simulation for Spatial Point Processes.* London: Chapman and Hall.

[18] Norris, J. R. (1998) *Markov Chains.* Cambridge, UK: Cambridge University Press.

[19] Rolski, T., Schmidli, H., Schmidt, V. and Teugels, J. (1999) *Stochastic Processes for Insurance and Finance.* Chichester: John Wiley.

[20] Ross, S. M. (1996) *Stochastic processes.* 2 ed. New York: Wiley.

[21] Serfozo, R. F. (1972) Conditional Poisson processes. J*ournal of Applied Probability.* **9**, 288-302.

[22] Shao, J. (2003) *Mathematical Statistics.* 2 ed. New York: Springer texts in statistics.

[23] Smith, J. A. and Karr, A. F. (1983) A point process model of summer season rainfall occurrences. *Water Resources Research.* **19**, 95-103.

[24] Tong Zhou, G., Schafer, W. R. and Schafer, R. W. (1998). A three-state biological point process model and its parameter estimation. *Signal Processing, IEEE Transactions on.* 46(10), 2698-2707.

[25] Taylor, H. M. and Karlin S. (1998) *An introduction to stochastic modeling.* 3 ed. San Diego: Academic Press.