

UNIVERSITY OF NOTTINGHAM



SCHOOL OF MATHEMATICAL SCIENCES

# Cell Signalling

Jake Powell

A thesis submitted to the University of Nottingham for the degree of  
DOCTOR OF PHILOSOPHY

ONE DAY

---

## Abstract

---

Original abstract: Intracellular Calcium oscillations is a versatile signalling mechanism responsible for many biological phenomena including immune responses and insulin secretion. There is now compelling evidence that whole-cell calcium oscillations are stochastic, arising from random molecular interactions at the subcellular level. This poses a significant challenge for modelling. Here, we utilise a probabilistic method that treats calcium oscillations as a point process. By employing an intensity function, we capture intrinsic cellular heterogeneity as well as inhomogeneous extracellular conditions, such as time-dependent stimulation. I will demonstrate how to simulate stochastic calcium spikes based on intensity functions. Furthermore, we will infer model parameters from real data using Bayesian methods, and utilise novel MCMC techniques to deal with priors on functions.

## Acknowledgements

Thank some folk.

---

## Contents

---

<b>Abstract</b>	<b>i</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Calcium Signalling . . . . .	2
1.1.1 Encoding and decoding $\text{Ca}^{2+}$ Patterns . . . . .	4
1.2 How do we generate spike sequences? . . . . .	5
1.2.1 Calcium Signalling Models . . . . .	5
1.2.2 $\text{Ca}^{2+}$ oscillation modelling . . . . .	5
1.2.3 $\text{Ca}^{2+}$ waves . . . . .	6
1.2.4 Stochastic Models . . . . .	7
<b>2 Model and Bayesian Inference</b>	<b>8</b>
2.1 The model . . . . .	8
2.2 Simulating Spikes . . . . .	12
2.3 Bayesian Inference . . . . .	14
2.4 Bayesian Computation . . . . .	18
2.4.1 Simple Model . . . . .	19
2.4.2 Piecewise Constant Prior . . . . .	23
2.4.3 Gaussian Process Prior . . . . .	34

---

2.4.4	Updating ISI Parameter . . . . .	40
2.5	Model Assessment . . . . .	42
2.6	Example with simulated data . . . . .	43
<b>3</b>	<b>Non-trivial Model Properties</b>	<b>51</b>
3.1	ISI Distribution . . . . .	51
3.1.1	Non-Identifiability issues with two parameter ISI distributions	52
3.1.2	Choosing one-parameter ISI distributions . . . . .	52
3.1.3	Gamma distribution . . . . .	53
3.1.4	Inverse Gaussian distribution . . . . .	54
3.1.5	Log Normal distribution . . . . .	55
3.1.6	Weibull distribution . . . . .	56
3.2	Speed of the GP . . . . .	57
3.2.1	Projection . . . . .	57
3.2.2	Spectral representation . . . . .	57
3.3	Estimating Length Scale . . . . .	59
<b>4</b>	<b>Refractory Period</b>	<b>60</b>
4.0.1	How it affects the model . . . . .	60
4.0.2	Estimating $T_{\min}$ . . . . .	61
<b>5</b>	<b>Application to Real Data</b>	<b>63</b>
5.1	Section . . . . .	63
<b>6</b>	<b>Conclusion</b>	<b>64</b>
6.0.1	Web application for simulating spikes . . . . .	64
6.0.2	Web application for viewing/analysing raw data . . . . .	67

---

## List of Figures

---

2.1	Fura-2 fluorescence intensity traces of a HEK293 cell stimulated initially by $20\mu\text{M}$ carbachol and then with $50\mu\text{M}$ carbachol at 3444s. Where spike times (red rugs) are recovered by thresholding (red line) the concentration values at $0.8\mu\text{M}$ . (Data taken from M Falcke for experiment details see K et al. (2014) ) . . . . .	9
2.2	(A) Rastor plot of spikes simulated from a Gamma ISI distribution with parameter value $\gamma = 10$ and intensity function $x(t) = 2 \cos \frac{t}{2} + \cos \frac{t}{4} + 2.8$ , between 0 and 20s. (B) Intensity function (red) alongside peristimulus-time histogram (blue) obtained from 1000 spike sequences simulated from the same distribution as (A). . . . .	15
2.3	Posterior results from the simple model. Each grey point represents one sample from the posterior distribution, where overlapping values become darker. The red dot is the value obtained using MLE. The histograms, top and right, represent the individual posterior distributions of $x$ and $\theta$ respectively. . . . .	21
2.4	Realisations from the PWC prior. Where $T = 20, k_{\max} = 10, \lambda = 5, \kappa = 1$ and $\mu = 1$ . In (A) the heights are independent and in (B) the heights follow the martingale structure. . . . .	25

- 
- 2.5 Illustration of the possible moves at each iteration of the RJMCMC. The black line corresponds to the original step function. The blue line shows a candidate function when a change point is removed. Red when a new change point is added. Cyan when a height is changed. Magenta when a change point moves. . . . . 27
- 2.6 Realisation of a Gaussian Process (top line), with mean 0 and SE covariance. The bell shape curves show the probability distributions of the values the realisation can take at times 0, 50, 100 and 150, where the probability is coloured such that light blue indicates low probability and magenta high probability. The dashed lines indicated the probability of the given value that the GP takes at that time. . . 34
- 2.7 Illustration of how varying hyper parameters changes realisations of a Gaussian Process with mean 0. We draw 3 samples from the following GP priors: (A)  $\sigma_f^2 = 1, l = 2, \sigma_v^2 = 0$  (B)  $\sigma_f^2 = 1, l = 1, \sigma_v^2 = 0$  (C)  $\sigma_f^2 = 2, l = 2, \sigma_v^2 = 0$  (D)  $\sigma_f^2 = 1, l = 2, \sigma_v^2 = 0.01$ . The magenta region indicates the 95% confidence intervals. In (B) we notice that decreasing the length scale leads the realisation to vary more frequently with time. In (C) increasing  $\sigma_f^2$  increases the range of values taken (variance). Finally, (D) shows noise is controlled by  $\sigma_v^2$ . . . . . 47
- 2.8 Draws from the GP prior for the intensity, where we assume the logarithm of  $x(t)$  has a GP prior. Where the black, red and blue lines correspond to length scales 10, 5 and 1. To illustrate that a priori we don't know the value of  $l$  and it will be fitted using MCMC. . . . . 48

2.9	Results from simulated data with constant intensity. (A) Rastor plot of 40 of the simulated spike sequences. (B) Posterior intensity function results for a single spike sequence, where the black line corresponds to the intensity the spikes were sampled from. In addition red, blue and magenta lines correspond to posterior intensity computed via PWC, GP and simple priors respectively, where the shaded areas correspond to the 95% confidence regions. (C) Histogram of the posterior samples of the ISI parameter $\gamma$ for the same spike sequence as (B) where the colours represent the priors PWC (red), GP (blue) and simple (magenta). (D) Scatterplot of the MLEs for 200 spike sequences, where each dot corresponds to the MLE of a single spike sequence. . . . .	49
2.10	Results from simulated data with time-varying intensity. (A) Rastor plot of 40 of the simulated spike sequences. (B,C) Posterior intensity function results for a single spike sequence, where the black line corresponds to the intensity the spikes were sampled from. In addition red, blue and magenta lines correspond to posterior intensity computed via PWC, GP and simple priors respectively, where the shaded areas correspond to the 95% confidence regions. (D) Scatterplot of average posterior intensity function for 40 spike sequences, where each red line corresponds to the mean posterior intensity function for a single spike sequence. . . . .	50
6.1	Examples of generated spike sequences using SimSpikes. . . . .	66
6.2	Images of the files downloaded from the step change intensity function SimSpikes. . . . .	67
6.3	Print-screen of main pain of ViewData app where the threshold input has been updated to be 0.7. . . . .	69
6.4	Print screen of App where threshold has been updated to '0.7,0.45,0.7,800,1000'. . . . .	70
6.5	Main Panel where we remove two transient regions. The first region between 0s and 200s. The second interval is between 3300s and 3500s. . . . .	71



---

6.6	Main panel where linear trends are removed. . . . .	72
6.7	Images of the files downloaded from the step change intensity function SimSpikes. . . . .	73

---

## List of Tables

---

# CHAPTER 1

---

## Introduction

---

Calcium ( $\text{Ca}^{2+}$ ) is the alkaline earth metal, first discovered by Sir Humphry Davy in 1908 by electrolysis of its oxide. ?, the name ‘Calcium’ is derived from the latin calx meaning lime as it was extracted by heating limestone.

$\text{Ca}^{2+}$  is needed for the normal function of cells and their survival. The importance of  $\text{Ca}^{2+}$  in animal cells was discovered by accident in 1883 by Sidney Ringer ?, predating its first isolation. In 1882, Ringer investigated the affects of sodium and potassium had on the contractions of isolated frog hearts, and found that the compounds effected the heart beating. ? However, he discovered that the laboratory ran out of distilled water during the experiment and a technician had used pipe water instead. Hence, he repeated the experiment using distilled water but found the results differed from the original experiment, and he hypothesised the observed effects were “due to the inorganic constituents of the pipe water” ?. That constituent was  $\text{Ca}^{2+}$ .

In the years since that discovery, it is known that in addition to influencing the heart beat,  $\text{Ca}^{2+}$  has 4 main biological roles :

- cofactor for enzymes or proteins,
- electrical (i.e. in the formation of action potentials in excitable cells),

- intracellular second messenger
- structural (i.e. in skeletal structures such as bones or shells).

In this thesis we consider  $\text{Ca}^{2+}$ 's role as an intracellular second messenger, or to be more precise we investigate how changes to the  $\text{Ca}^{2+}$  concentration inside a cell informs us of the applied stimulation. We begin by introducing intracellular  $\text{Ca}^{2+}$  signalling, before reviewing the literature of what mathematical models have been used to analyse the signals. Finally, we introduce the goals of our project.

## 1.1 Calcium Signalling

In its simplest form cells at rest have a  $\text{Ca}^{2+}$  concentration of  $100\text{nM}$  but activates when the level rises to  $\sim 1\mu\text{M}$ . This elevation of  $\text{Ca}^{2+}$  can regulate many processes due to its versatility, i.e. speed, amplitude and spatio-temporal patterning.

Calcium takes part in a variety of biological functions such as cell maturation, ?, steering bacterial infection Van Nhieu et al. (2003) and orchestrating fertilisation Santella et al. (2004); Whitaker (2006); Denninger et al. (2014). Along with these functions excessive  $\text{Ca}^{2+}$  can damage cells and even cause apoptosis ??.

The overarching idea of the  $\text{Ca}^{2+}$  signalling network can be split into 4 compartments:

1. Signalling is triggered by a stimulus that generates various  $\text{Ca}^{2+}$ -mobilising signals.
2. The  $\text{Ca}^{2+}$ -mobilising signals activate the ON mechanism that feeds  $\text{Ca}^{2+}$  into the cytoplasm
3.  $\text{Ca}^{2+}$  functions as a messenger to simulate numerous  $\text{Ca}^{2+}$ -sensitive processes.
4. The OFF mechanism, composed of pumps and exchangers, remove  $\text{Ca}^{2+}$  from the cytoplasm to restore the resting state.

We will illustrate the process by considering Human embryonic kidney (HEK) cells exposed to the stimulant Carbohol, which induces  $\text{Ca}^{2+}$  oscillations driven by  $\text{Ca}^{2+}$

release through the endoplasmic reticulum (ER) through inositol-1,4,5-triphosphate ( $\text{InsP}_3$ ) receptors ( $\text{InsP}_3\text{R}$ ), see figure –.

Carbohol acts as an agonist of the muscarinic  $\text{M}_3$  receptor in the extracellular medium, which itself is coupled with a G-protein which induces the activation of Phospholipase C (PLC). The increase of PLC causes subsequent production of  $\text{IP}_3$  (and diacyl-glycerol DAG), via the hydrolysis of PLC and phosphatidylinositol 4,5-bisphosphate ( $\text{PIP}_2$ ). Therefore Carbohol has caused the activation of  $\text{IP}_3$ , a  $\text{Ca}^{2+}$ -mobilising signal.

$\text{IP}_3$  then diffuses into the cell across the cell membrane, and binds to  $\text{IP}_3\text{Rs}$  situated on the ER. The ER is a compartment inside the cell that stores  $\text{Ca}^{2+}$ , on its membrane  $\text{IP}_3\text{R}$  are positioned in clusters with 5-30  $\text{IP}_3\text{R}$  channels per cluster. ?.

Describe the structure of  $\text{IP}_3\text{R}$ . The  $\text{IP}_3\text{R}$  opens in response to  $\text{IP}_3$  and  $\text{Ca}^{2+}$  binding to activation sites. The activated channels release large amount of  $\text{Ca}^{2+}$  into the cytoplasm.

Some of the released  $\text{Ca}^{2+}$  binds to a receptors inactivation site which closes the channel. The channel cannot reopen for some time after inactivation (the channel is in a refractory state) ??. After  $\text{IP}_3\text{R}$  closure cytosolic  $\text{Ca}^{2+}$  is removed through plasma membrane ATPase (PCMA) and sarco-plasmic reticulum  $\text{Ca}^{2+}$ ATPase (SERCA) pumps out of the cell and into the ER respectively ?. Cytosolic  $\text{Ca}^{2+}$  is also controlled by buffers - compounds that bind free  $\text{Ca}^{2+}$  such as ATP and calretinin - hence allowing  $\text{Ca}^{2+}$  concentration to remain in an operational range. Since  $\text{Ca}^{2+}$  is cytotoxic. The intracellular  $\text{Ca}^{2+}$  buffering capacity depends on the type of cell ???

Repeated increases and decreases in  $\text{Ca}^{2+}$  caused by the opening/closing of  $\text{IP}_3\text{R}$  along with  $\text{Ca}^{2+}$  pump activity creates  $\text{Ca}^{2+}$  oscillations. Depending on the cell type the period of oscillations varies from seconds to minutes, i.e. 10ms spike duration in toadfish swimbladder muscle or 1 day for circadian rhythms ?? and oscillations range from regular spiking to bursting ?.

Changes in cytosolic  $\text{Ca}^{2+}$  concentration generally doesn't occur uniformly across the cell, rather local  $\text{Ca}^{2+}$  changes can create  $\text{Ca}^{2+}$  events are often categorised into 3 groups: blips, puffs and waves. A Blip occurs when  $\text{Ca}^{2+}$  is released from a

single receptor, whereas a puff arises when a cluster of receptors is activated.  $\text{Ca}^{2+}$  can diffuse from one puff site to surrounding clusters activating the receptors and causing wave propagation. In astrocytes at least three puff sites need to be activated synchronously to cause a wave ?. Intracellular  $\text{Ca}^{2+}$  waves spread throughout the cell and can transfer information from one part of the cell to another.

### 1.1.1 Encoding and decoding $\text{Ca}^{2+}$ Patterns

Cells use spatio-temporal patterns to transmit information (both within the same cell and to surrounding cells) ??? and consequently to initiate an appropriate physiological response. ?. However, it is not yet fully understood how encoding and decoding mechanisms convey information. To date, it has been shown that  $\text{Ca}^{2+}$  responses can vary amongst different cell types ?, and even in the same cell type ?. Along with cell type, the agonist type ??? and its concentration ? also affects  $\text{Ca}^{2+}$  response.

The relationship between  $\text{Ca}^{2+}$  oscillations and agonist was first studied in 1986 by Woods et al. ? They found that  $\text{Ca}^{2+}$  oscillations depended on the hormone concentration used for stimulation in single, isolated rat hepatocyte cells. From this, they hypothesised that only the frequency of  $\text{Ca}^{2+}$  oscillations were effected by the agonist and both amplitude and width of the oscillations were unaffected, which was confirmed in 1987 ?.

Two hypothesis' for stimulus encoding followed from the fixed agonist experiments. Namely, amplitude and frequency modulation (AM and FM ?). AM proposes that the concentration of agonist increases the amplitude of  $\text{Ca}^{2+}$  signals, whereas FM claims that the frequency of the  $\text{Ca}^{2+}$  signals increase with the stimulus strength. Recent studies have shown that  $\text{Ca}^{2+}$  spike times scales exponentially with the stimulus strength for FM K et al. (2014).

## 1.2 How do we generate spike sequences?

The aim of our work is to fit statistic models to real spike data. However, before any model fitting it is best to understand how we get spike data from individual cells.

### 1.2.1 Calcium Signalling Models

Calcium signalling is often modelled using either deterministic or stochastic models. In this section we provide a short overview of these models. For a more detailed view, we refer readers to ?????.

### 1.2.2 $\text{Ca}^{2+}$ oscillation modelling

We have shown that cytosolic  $\text{Ca}^{2+}$  concentration,  $c(t)$  is controlled through  $\text{IP}_3\text{R}$  and  $\text{SERCA/PMCA}$  pumps.  $\text{IP}_3\text{R}$  controls  $\text{Ca}^{2+}$  release from the ER into the cytoplasm ( $J_{\text{IP}_3\text{R}}$ ). Whereas  $\text{SERCA/PMCA}$  pumps transport  $\text{Ca}^{2+}$  from the cytoplasm back into the ER ( $J_{\text{SERCA}}$ ) and outside the cell ( $J_{\text{PMCA}}$ ) respectively. Combining the fluxes, we can mathematically describe the  $\text{Ca}^{2+}$  concentration by

$$\frac{dc}{dt} = J_{\text{IP}_3\text{R}} - J_{\text{SERCA}} - J_{\text{PMCA}}. \quad (1.1)$$

We could include more terms such as the flux into the cytoplasm from  $\text{RyR}$ , or leak from the internal stores into the cytosol. One of the first realistic models for  $\text{Ca}^{2+}$  oscillations was produced by De Young and Keizer (DYK) in 1992 ?. The model became an archetype for more recent deterministic ??? and stochastic models ??.

The DYK model considers only two fluxes, one into the cytosol via the  $\text{IP}_3\text{R}$  channels and a leak from the internal store, and one out of the cytosol via a ATP-dependent pump (i.e. the  $\text{SERCA}$  pump). The  $\text{SERCA}$  pump is described by the Hill function of  $c(t)$

$$J_{\text{SERCA}} = \frac{vc^2}{k^2 + c^2}, \quad (1.2)$$

where parameters  $v, k$  represent maximum  $\text{Ca}^{2+}$  uptake and pump activation constant respectively ?.

Recall that studies have found that the  $\text{IP}_3\text{R}$  is activated when at least 3 of the 4 subunits are activated. Thus the DYK model assumes that an  $\text{IP}_3\text{R}$  consists of 3 identical independent subunits. Each subunit contains one  $\text{IP}_3\text{R}$  binding site, one  $\text{Ca}^{2+}$  activation site and one  $\text{Ca}^{2+}$  inhibition site. As such each subunit can be in one of 8 states (figure – ), which can be specified by the binary triplet  $ijk \in \{0,1\}^3$ , where the indexes correspond to the  $\text{IP}_3\text{R}$  site,  $\text{Ca}^{2+}$  activation site,  $\text{Ca}^{2+}$  inhibition site in order. An index of 0 means the site is unbound whereas 1 represents the site been bound. A subunit is activated when it is in state 110 i.e. the  $\text{IP}_3\text{R}$  site and  $\text{Ca}^{2+}$  activation site is bound and the  $\text{Ca}^{2+}$  Inhibition site is unbound. Hence the the channel is open when all 3 subunits are activated. The probability that the subunit is in state  $ijk$  is denoted by  $x_{ijk}$ . By applying mass-action kinetics we retrieve eight ODEs for each subunit state, for example

$$\frac{dx_{000}}{dt} = b_1x_{100} + b_5x_{010} + b_4x_{001} - x_{000}(a_1I + a_4c + a_5c). \quad (1.3)$$

However, since we must be in one of the 8 states we can replace one of the ODEs with the condition  $\sum x_{ijk} = 1$ .  $\text{IP}_3$  concentration can be fixed, however for more realistic results it is possible to include it's own dynamics, e.g ????.

Thus, with the  $\text{IP}_3\text{R}$  subunit open probability of  $x_{110}$ , the outward flux of the DYK model can be computed by

$$J_{\text{IP}_3\text{R}} = c_1 (v_1 x_{110}^3 + v_2) (c_{\text{er}} - c), \quad (1.4)$$

where the parameters  $c_1, c_{\text{er}}, v_1$  and  $v_2$  correspond to the ratio of ER to the cytosolic volume,  $\text{Ca}^{2+}$  concentration in the ER, maximal  $\text{Ca}^{2+}$  influx and  $\text{Ca}^{2+}$  leak respectively.

### 1.2.3 $\text{Ca}^{2+}$ waves

Whilst the DYK improved understanding of  $\text{Ca}^{2+}$  signalling toolbox, it only considers temporal changes in the cytosolic  $\text{Ca}^{2+}$  concentration. It is know (see figure – ) that  $\text{Ca}^{2+}$  patterns vary in both time and space. To incorporate this into (??) a diffusion term is added

$$\frac{\partial c}{\partial t} = D_c \frac{\partial^2 c}{\partial x^2} \quad (1.5)$$



### 1.2.4 Stochastic Models

## CHAPTER 2

---

### Model and Bayesian Inference

---

In this chapter we describe how calcium ( $\text{Ca}^{2+}$ ) spikes can be modelled as point processes, and how to create time-heterogeneous models from stationary inter-spike interval (ISI) dynamics, by so-called intensity functions. We then explain how to simulate spike sequences from the model, which will later be used to create surrogate data to check model fitting. Afterwards we explain how to use Bayesian inference to fit the model to spike data, by employing Markov chain Monte Carlo (MCMC). In particular, we give two different prior models for the intensity function and discuss how to sample from the posterior distribution of the model. Finally, we consider model assessment applying two complementary graphical techniques. We provide explicit guidance on how to simulate and fit model parameters for the Gamma ISI distribution, followed by an example using simulated data.

### 2.1 The model

Before we create a model for  $\text{Ca}^{2+}$  spiking in cells, we first need to understand the raw data received from experimentalists. The raw data consists of time-series data for

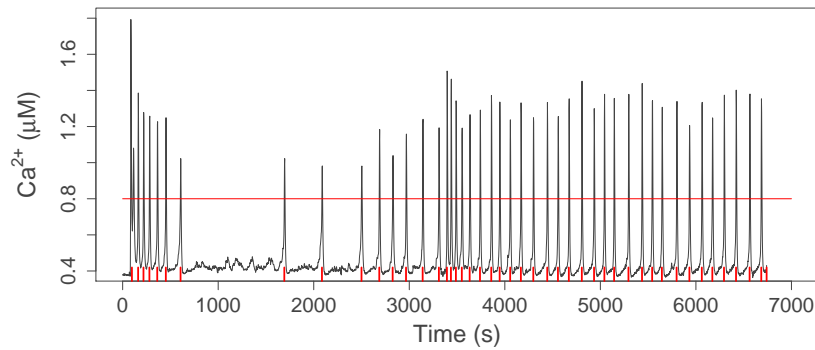


Figure 2.1: Fura-2 fluorescence intensity traces of a HEK293 cell stimulated initially by  $20\mu\text{M}$  carbachol and then with  $50\mu\text{M}$  carbachol at 3444s. Where spike times (red rugs) are recovered by thresholding (red line) the concentration values at  $0.8\mu\text{M}$ . (Data taken from M Falcke for experiment details see K et al. (2014) )

concentration of calcium in an individual cell, such as the black trace plot in figure 2.1, from which we can see the transient rises and falls in the  $\text{Ca}^{2+}$  concentration. We then transform the data into a series of spikes by thresholding the calcium concentration (red line in figure 2.1) and obtain the spike times (red rugs on x-axis). This is shown for one cell, however we have data from a variety of cells (HEK293, HeLa, Astrocyte, etc) under different stimuli. Hence our aim is to create a model to mimic the behaviour of  $\text{Ca}^{2+}$  spikes in cells, and create simulated data, in addition to fitting the model to individual cells and recovering the spiking rate and spike variance. Once we fit the model, we can ask questions such as, could we infer what type of cell or stimulus strength was used in the experiment?

A natural starting point to model  $\text{Ca}^{2+}$  spikes are point processes, where each spike is an event in the process. Point processes are fully defined by their inter-event distribution, or in our case inter-spike interval (ISI) distribution. Namely, the probability  $p(s, t)$  of a spike occurring at time  $t$  given that the last spike occurred at time  $s$ , i.e spikes occur at times  $s$  and  $t$  with no spikes in the interval  $(s, t)$ . ISIs

are obtained directly from experimental recordings, from which we can acquire the empirical distribution. From this we can choose a probability distribution to model the ISI, such as the Gamma, Weibull, inverse Gaussian distributions. For example the probability density function of the Gamma distribution is

$$p(s, t | \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} (t - s)^{\alpha-1} e^{-\beta(t-s)}, \quad (2.1)$$

where  $\alpha$  and  $\beta$  are the shape and rate parameters respectively and  $\Gamma$  denotes the Gamma function. Research has shown that directly after a spike in  $\text{Ca}^{2+}$  concentration the probability of a spike occurring shortly afterwards is low K et al. (2011). The downtime after a spike when no spikes occur is known as the refractory period  $t_{\text{refract}}$ . The Gamma distribution is a suitable choice since increasing both  $\alpha$  and  $\beta$  simultaneously, whilst retaining the ratio  $\alpha/\beta$ , has the affect of maintaining the mean of the distribution but decreasing the variance, which will in turn reduce the probability of a spike occurring in the refractory period. Another approach would be to directly incorporate the refractory period into the ISI probability density, i.e add a parameter  $t_{\text{refract}}$  such that if  $t < t_{\text{refract}}$  then  $p(s, t) = 0$ . This is done with the exponential distribution in A et al. (2011). The advantage of this approach is that since we assume spikes ISIs are independent and identically distributed the probability of an entire sequence of spikes factorises into individual ISIs. Suppose  $\text{Ca}^{2+}$  spikes occur at times  $y_1, y_2, \dots, y_n$  and denote the set of all spikes by  $\mathbf{y}$ . Then the probability density of the spikes sequence factorises to

$$p(\mathbf{y}) = p_1(0, y_1) p(y_1, y_2) \dots p(y_{n-1}, y_n) p_n(y_n, T), \quad (2.2)$$

where  $p_1(0, y_1)$  denotes the probability density of the first spike occurring at time  $y_1$  and  $p_n(y_n, T)$  denotes the probability density of no spikes occurring between  $y_n$  and the end of experiment time  $T$ . These events are separated as they do not constitute an ISI, and as such they are often modelled by different probability distributions, such as an exponential.

These models are often referred to as stationary, since the ISIs have identical distributions. Moreover, the ISI probability depends only on the time since the last

spike and not on the absolute experiment time. For example, suppose the first two spikes are separated by 100s, the probability of this occurring is the same whether the first spike occurs 50s or 1000s into the experiment. From figure 2.1 we notice that after the stimulus is exchanged the rate of spiking increases (approximately at 3500s), in this instance it appears that the spiking rate depends on the absolute time of the experiment. Furthermore, there are numerous biological reasons why the ISI probabilities do not remain constant over time (see chapter *intro*).

Thus we seek a method to mathematically describe the non-stationarity of the ISIs. One approach would be to add an explicit time-dependence into the ISI distribution by making the ISI parameters vary with time. However, the probability of a spike sequence need not factorise as (2.2), and we would have to consider the full multivariate probability density  $p(\mathbf{y}) = p(y_1, \dots, y_n)$ , which can present substantial difficulties.

A more practical approach was used by Barbieri et al in 2001 R et al. (2001) via a time-rescaling method which maps the original experiment time  $t$  to a new time  $z$  by

$$z(t) = \int_0^t x(\nu) d\nu, \quad (2.3)$$

where  $x(\cdot)$  is called the intensity function, and is strictly positive. Since  $x(\cdot)$  is positive it associates each  $t$  with a unique value of  $z$ . The intensity encompasses the history dependence, and once rescaled in time  $z$  the ISIs become independent, which allows the spike sequence to factorise into individual ISI contributions

$$p(\mathbf{y}|x) = g_1(0, z_1)g(z_1, z_2) \dots g(z_{n-1}, z_n)g_n(z_n, Z). \quad (2.4)$$

Where  $z_i = z(y_i)$  and  $Z = z(T)$  are the transformed spike times and experiment time respectively. We include the dependency on the intensity  $x(t)$  directly to emphasise that the transformation depends on the intensity function. The probability density  $g$  relates to the original ISI probability density  $p$  by

$$g(z_i, z_{i+1}|x) = \frac{dz}{dt} \bigg|_{y_{i+1}} p(z_i, z_{i+1}), \quad (2.5)$$

$$= x(y_{i+1})p(z_i, z_{i+1}). \quad (2.6)$$

To illustrate this with the Gamma ISI distribution we begin with the one-parameter Gamma with mean 1, and denote its parameter by  $\gamma$ . The probability density of a spike occurring at time  $y_i$  given the last spike occurred at  $y_{i-1}$  is

$$p(y_{i-1}, y_i | \gamma) = \frac{\gamma^\gamma}{\Gamma(\gamma)} (y_i - y_{i-1})^{\gamma-1} e^{-\gamma(y_i - y_{i-1})}. \quad (2.7)$$

Substituting (2.7) into (2.6) gives the new ISI probability density

$$p(y_{i-1}, y_i | x, \gamma) = \frac{\gamma x(y_i)}{\Gamma(\gamma)} [\gamma X(y_{i-1}, y_i)]^{\gamma-1} e^{-\gamma X(y_{i-1}, y_i)}, \quad (2.8)$$

where

$$X(s, t) = \int_s^t x(u) du, \quad \text{for any } s, t \in [0, T]. \quad (2.9)$$

From now on we drop the notation that the probability density is denoted by  $g$  in the transformed time and use  $p$  throughout. At this stage, it may appear that the intensity  $x(t)$  is devoid of biological meaning, and only a mathematical trick. However, in fact  $x(t)$  corresponds to the spiking rate independent of spike history. An example is shown in section 2 (figure 2.2B).

In summary, we model  $\text{Ca}^{2+}$  sequences as point processes that are fully described by their ISI distribution, which requires distribution type (Gamma, Weibull, etc) with parameter  $\theta$  and the intensity function  $x(t)$ . Above we have shown the case for the Gamma distribution, however the concept can be used for any probability distribution on  $(0, \infty)$  used to describe the ISI dynamics.

Notice that the more complex time-heterogeneous model contains the simpler stationary model. To visualise this suppose that the intensity function is constant  $x(t) = x^*$  for some  $x^* > 0$ , then the integrated intensity simplifies to  $X(s, t) = x^*(t-s)$ . Substituting this into (2.8) we retrieve the simple Gamma distribution with shape  $\alpha = \gamma$  and rate  $\beta = \gamma x^*$ , in (2.1).

## 2.2 Simulating Spikes

Suppose we are given the ISI distribution with parameter  $\theta$  and the intensity function  $x(t)$ , our goal is to create spike sequences from the model. There are numerous ways

to simulate spike sequences, such as time rescaling, an approach using a Bernoulli process and the inverse transform method EN et al. (2002); W et al. (2005); L (2013). For all ISI distributions bar the exponential we use the inverse transform method. Since the exponential distribution leads to an inhomogeneous Poisson process, we use a thinning method DJ and D (2003). This method begins by simulating from the homogeneous Poisson process with rate  $\max(x(t))$ , and then thinning (removing) spikes with probability  $x(y_i) / \max(x(t))$ .

The inverse transform method samples from the target distribution by initially sampling a number  $a$  from the uniform distribution on  $[0, 1]$ , interpreted as a probability. Then if the cumulative density function (CDF)  $F$  of the target distribution is known, return  $F^{-1}(a)$ , the inverse CDF evaluated at  $a$ . However, in our case the CDF does not have closed form. Hence we discretise time and numerically calculate the CDF.

Thus to simulate spike sequences we require the inputs: time length of experiment  $T$ , the ISI distribution, ISI parameter  $\theta$ , intensity  $x(t)$  defined on  $[0, T]$  and the discretisation of time defined by the number of steps  $K$ . With  $K$  steps the discretisation is defined by  $\mathbf{t} = \{t_i\}_{i=0}^K$  and  $t_i = iT/K$ . The method is described in Algorithm 1, where  $C$  stores the value of the CDF whilst we iterate through times  $t$  and  $y_{\text{cur}}$  stores the most recent spike time.

As stated previously,  $x(t)$  is the spiking rate independent of the spike history. Hence, if we simulate a large number of spike sequences from the same distribution and then average over all of the sequences this should return the intensity function.

In figure 2.2A we have simulated 50 spike sequences from the model defined by the intensity function  $x(t) = 2 \cos \frac{t}{2} + \cos \frac{t}{4} + 2.8$ , with Gamma ISI distribution with parameter  $\gamma = 10$ . In figure 2.2B using the same model we simulate 1000 sequences and binned the spikes (blue histogram). Since we used a large number of spike sequences this is equivalent to taking the average number of spikes across each small region, thus returning the expected number of spikes in each bin irrespective of the spike history. The strong agreement between the histogram and the intensity function (red line) confirms the interpretation of  $x(t)$ .

---

**Algorithm 1:** Simulating spike sequences.

---

**Input:**  $T$ ,  $K$ , ISI distribution,  $\theta$  and  $x(t)$ .

**Output:** A spike sequence  $\mathbf{y}$  between  $[0, T]$ .

Set  $t = 0$ ,  $y_{\text{cur}} = 0$ ,  $\mathbf{y} = \emptyset$  and  $h = T/K$ ;

**while**  $t < T$  **do**

Draw  $a \sim U(0, 1)$  and set  $C = 0$ ;

**while**  $C < a$  **do**

$t = t + h$ ;

**if**  $t > T$  **then**

STOP;

Numerically calculate  $C = \int_{y_{\text{cur}}}^t p(y_{\text{cur}}, u | x(u), \theta) du$ ;

Add  $t$  to the set of spike times  $\mathbf{y}$ , and set  $y_{\text{cur}} = t$ ;

**Return** spike times  $\mathbf{y}$ ;

---

## 2.3 Bayesian Inference

We have shown that given some parameters for a cell spiking we can simulate spike sequences. Now consider the reverse question, if given spike data can we infer the model parameters from which the spikes occurred. Informally, what does the data inform us about the parameters of the model? Consider a single cell and observe its  $\text{Ca}^{2+}$  concentration over the time period  $[0, T]$ . From this we determine the times  $0 < y_1 < \dots < y_N < T$  at which the  $\text{Ca}^{2+}$  spikes occur. Let  $\mathbf{y} = \{y_i\}_{i=1}^N$  denote the set of all spike times. Then the quantity of interest is the *posterior* distribution  $\pi(x, \theta | \mathbf{y})$ , which is the probability distribution of the parameters given the data. This approach has the benefit that we do not merely get a point estimate for our parameters but a joint distribution of  $x$  and  $\theta$ .

Consider the distribution for  $\theta$  the ISI parameter. If the distribution is strongly centred around the value  $\theta^*$  then we can be confident that  $\theta^*$  is a good estimate for  $\theta$ . However, picture a distribution with multiple maxima or large variance, this would imply less confidence in the estimate. This is an advantage of distributions over point



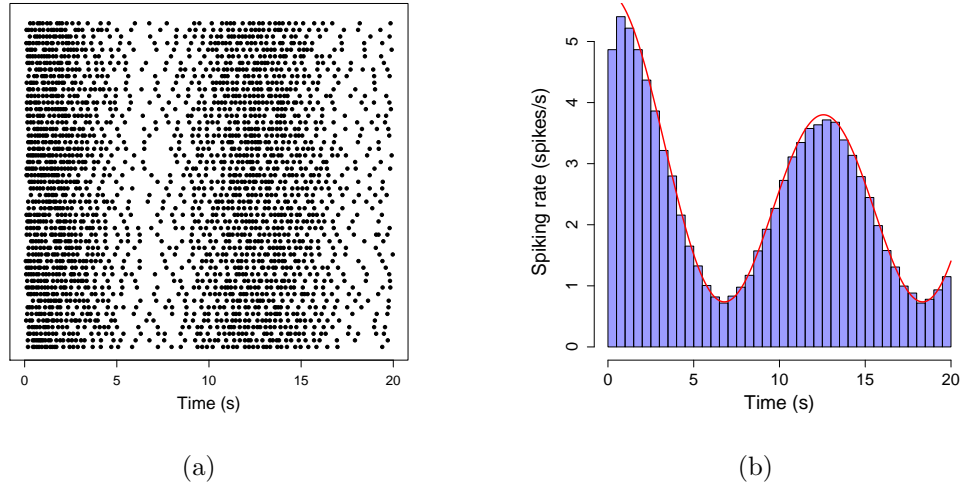


Figure 2.2: (A) Raster plot of spikes simulated from a Gamma ISI distribution with parameter value  $\gamma = 10$  and intensity function  $x(t) = 2 \cos \frac{t}{2} + \cos \frac{t}{4} + 2.8$ , between 0 and 20s. (B) Intensity function (red) alongside peristimulus-time histogram (blue) obtained from 1000 spike sequences simulated from the same distribution as (A).

estimates.

Furthermore, the joint posterior distribution is all we need to answer any question about the experiment or model dynamics, such as mean, variance or the interplay between ISI parameter and intensity function.

To calculate the posterior distribution we apply Bayes' theorem

$$\pi(x, \theta | \mathbf{y}) = \frac{\pi(\mathbf{y} | x, \theta) \pi(x, \theta)}{\pi(\mathbf{y})}. \quad (2.10)$$

On the right hand side we have: the likelihood function  $\pi(\mathbf{y} | x, \theta)$ , the prior  $\pi(x, \theta)$  and the normalising constant  $\pi(\mathbf{y})$ .

We need to derive an expression for the likelihood in our model. In our case this means the likelihood of obtaining the spike sequence given the parameters of the model, i.e. the probability of spike sequence  $\mathbf{y}$  given  $x(t)$  and  $\theta$ . This is exactly how we defined the model in section 1.1 (see (2.4) and (2.6)), and as such the joint

probability density of a spike sequence factorises into individual ISIs. Thus, the likelihood is given by

$$\pi(\mathbf{y}|x, \theta) = p_1(y_1|x)p_T(T, y_N|x) \prod_{i=2}^N p(y_{i-1}, y_i|x, \theta). \quad (2.11)$$

Here,  $p_1(y_1|x)$  represents the conditional probability of finding the first spike at time  $y_1$ . Furthermore,  $p_T(T, y_N|x)$  represents the conditional probability of finding no spike in the interval  $(y_N, T]$ . We assume  $p_1$  and  $p_T$  come from an inhomogeneous Poisson process, i.e.

$$p_1(y_1|x) = x(y_1)e^{-X(0, y_1)}, \quad p_T(T, y_N|x) = e^{-X(y_N, T)}, \quad (2.12)$$

where  $X(\cdot, \cdot)$  comes from (2.9).

For the Gamma ISI distribution putting together equations (2.8), (2.11) and (2.12) gives the likelihood to be

$$\begin{aligned} \pi(\mathbf{y}|x, \gamma) &= x(y_1)e^{-X(y_0, y_1)}e^{-X(y_N, T)} \\ &\times \prod_{i=2}^N \frac{\gamma x(y_i)}{\Gamma(\gamma)} [\gamma X(y_{i-1}, y_i)]^{\gamma-1} \exp(-\gamma X(y_{i-1}, y_i)). \end{aligned} \quad (2.13)$$

The *prior* contains our beliefs about the parameters before seeing the data. For example, if we strongly believed  $\theta$  to be close to the value  $\theta^*$  then we would choose a prior distribution that is centred around  $\theta^*$ , such as  $\pi(\theta) = \text{Gamma}(\kappa\theta^*, \kappa)$  for some value of  $\kappa$ . On the other hand, if we are unsure of the value of  $\theta$  we would choose a wider prior. Moreover, if we believe there is a positive correlation between say  $\theta_1$  and  $\theta_2$  then a priori we could choose a joint distribution with positive correlation such as the bivariate normal,  $\pi(\theta_1, \theta_2) = \mathcal{N}((\theta_1^*, \theta_2^*), \Sigma)$ , where  $\Sigma$  has positive correlation.

It is easy to think of priors for ISI parameter  $\theta$  as this is just a point variable, however it is harder to imagine a prior for the function  $x(\cdot)$ . We require a prior over the space of functions on  $[0, T]$ . The first method to combat this is to add a parametric form to  $x(t)$ , the most basic of which would be to assume it is constant,  $x(t) = x$ . This simplifies the problem greatly and a distribution on  $(0, \infty)$  would

suffice. To allow for time-heterogeneity more complex forms could be considered, such as a quadratic  $x(t) = at^2 + bt + c$ . Then to describe our model we would need priors on  $\{a, b, c\}$ , such as independent Gamma distributions. However, even with this improvement over the constant intensity, we are still limited to what shape  $x(t)$  takes. Furthermore, a priori there is little evidence to suggest what parametric form  $x(t)$  takes, and this could vary significantly from cell to cell.

Hence, we aim to use a non-parametric method for the prior as this allows more freedom on the shape that  $x(t)$  can take. We use two such methods, namely applying a Gaussian Processes (GP) - as suggested by Cunningham et al in 2008 JP et al. (2008) and a piecewise constant (PWC) prior. Details of how these priors work are in the subsequent sections 1.4.3 and 1.4.2 respectively.

Expanding the normalising constant we have

$$\pi(\mathbf{y}) = \int \pi(\mathbf{y}|x, \theta) \pi(x, \theta) dx d\theta, \quad (2.14)$$

which is an integral over all intensity functions and  $\theta$ . The high dimensionality of this integral would require efficient methods, as a direct integration would be computationally expensive if not impossible. Rather than compute the posterior directly, Tituaine et al in 2017 A et al. (t 17) determined the maximum of the distribution and its variance, by applying Laplace's approximation for the intensity and integrating out the ISI parameter. Whilst this is a sufficient method it does have its limitations, since it only approximates the intensity function. We adopt a different approach using Markov chain Monte Carlo (MCMC). The computational difficult part of the posterior calculation comes from the normalising constant. Thus, MCMC samples from  $p(x, \theta|\mathbf{y})$ , where we only need to know the distribution up to a multiplicative constant. Then from the samples we recover the possible values and their associated probabilities, without having to calculate a closed form solution. This is due to the fact that if a value is more likely then it will get sampled more often. The construction of the MCMC algorithms are detailed in the next section.

The method above can be generalised to input multiple spike sequences. Assuming the multiple spike sequences  $\{\mathbf{y}^i\}_{i=1}^M$  are all independent then the only effect is on

the likelihood, where we now have the product of all the individual spike sequences

$$\pi\left(\{\mathbf{y}^i\}_{i=1}^M|x,\theta\right)=\prod_{i=1}^M\pi\left(\mathbf{y}^i|x,\theta\right), \quad (2.15)$$

Multiple spike sequences may be considered if we believe the spikes all come from the same distribution.

## 2.4 Bayesian Computation

In this section we explore how to use MCMC methods to obtain the posterior distribution of the model parameters. The concept behind MCMC is that if the required probability density is known up to a multiplicative constant, then we can create a Markov chain whose stationary distribution is the probability density we desire. Once we create the chain, we start at some arbitrary initial value and iterate the Markov chain to create samples. Doing so many times, the samples converge onto the target stationary distribution. There are many standard MCMC methods (e.g Gibbs sampler, Metropolis-Hastings Algorithm G and E (1992); S and E (1995)), which can be used in a plethora of different problems. However, due to the fact that the intensity is a function of time we will require complex MCMC algorithms to sample from the desired posterior.

In layman's terms, an MCMC algorithm begins at some initial value, say  $\phi_0 = \phi_0$  where  $\phi$  can be and often is multi-dimensional. Suppose we are at iteration  $i$ , where the current value is  $\phi_i$ , we have a proposal regime that gives us a new candidate value  $\phi^*$ , often dependent on the current value  $\phi_i$ . Then we accept the candidate value with probability  $p_{acc}$ , relating to the probability that  $\phi^*$  is a better estimate than  $\phi_i$ , giving  $\phi_{i+1} = \phi^*$ , else  $\phi_{i+1} = \phi_i$ .

We utilise a Gibbs sampling method over  $x(t)$  and ISI parameter  $\theta$ . This means that rather than propose candidate values of both parameters together, we split the algorithm into two sections where we update  $x(t)$  first and then update  $\theta$ . So rather than sample from the full joint distribution we sample from the conditional distributions. One benefit of this method is that the conditional distributions are

often easier to deal with than the full joint distribution. The MCMC algorithm begins by choosing initial values  $x_0$  and  $\theta_0$  then iterating the process

*Step 1:* Update the  $x(t)$  applying either piecewise constant prior (section 1.4.2) or Gaussian Process prior (section 1.4.3);

*Step 2:* Update  $\theta$  using a Metropolis-Hastings algorithm (section 1.4.4).

Thus the algorithm returns samples for  $x$  and  $\theta$ , namely  $\{x_i\}_{i=1}^M$  and  $\{\theta_i\}_{i=1}^M$ , where  $M$  is the total number of iterations. It often takes the algorithm time to converge onto the target distribution, as such the first  $M_{\text{burn}}$  iterations are removed.

However, we begin in the next section by applying MCMC methods to the case  $x(t)$  is constant. This significantly reduces the difficulty of the problem, nevertheless it is a good starting point and is useful to compare with the time-dependent models later.

### 2.4.1 Simple Model

We begin by assuming the intensity function is constant,  $x(t) = x$ . As stated previously, this assumption gives rise to time-homogeneity. Furthermore, we get  $X(s, t) = x(t - s)$  for the integrated intensity function in the likelihood. The model is now simpler as we only need to estimate two parameters,  $x$  and the ISI parameter  $\theta$ . We use two different methods to achieve this, Maximum likelihood estimation (MLE) and MCMC.

We need to decide prior distributions for  $x$  and  $\theta$ , we choose to use Independent Gamma distributions for both  $x$  and  $\theta$ , i.e.

$$\pi(x) = \Gamma(\alpha_x, \beta_x) \quad \text{and} \quad \pi(\theta) = \Gamma(\alpha_\theta, \beta_\theta). \quad (2.16)$$

These Gamma distributions are often set to be non-informative with shape 1 and rate small. If we assume the ISIs follow a Gamma distribution then the likelihood

simplifies to

$$\pi(\mathbf{y}|x, \theta) = x e^{-x(y_1 - y_0)} e^{-x(T - y_N)} \times \prod_{i=2}^N \frac{\gamma x}{\Gamma(\gamma)} [\gamma x(y_i - y_{i-1})]^{\gamma-1} \exp(-\gamma x(y_i - y_{i-1})). \quad (2.17)$$

MLE is a method where we search the parameter space and return the values which give rise to the largest likelihood. This is done using a classical optimiser function (such as *optim* in R), and returns a point estimate. The benefit of this method is that it is quick and easy to implement. However, it only returns individual values for the parameters, rather than full distributions. For example suppose we have spikes  $\mathbf{y}$  and we want to fit the Gamma ISI distribution. MLE method works by solving for

$$(x^*, \gamma^*) = \max_{x, \gamma \in (0, \infty)} \pi(\mathbf{y}|x, \theta), \quad (2.18)$$

where  $\pi(\mathbf{y}|x, \theta)$  comes from (2.17).

For the MCMC method we use Gibbs sampling. We use random walk (RW)-Metropolis algorithms to update  $x$  and  $\theta$  separately. The RW-Metropolis algorithm is a special case of the Metropolis-Hastings algorithm that comprises of two steps: a proposal regime and whether to accept the proposal. Suppose at our current iteration we have a value  $z$  of the parameter of interest. We propose a new value  $z'$  using the proposal distribution  $q(z'|z)$  which is the conditional probability of proposing state  $z'$  given we are in state  $z$ . Then we accept  $z'$  with probability

$$p_{\text{acc}} = \min \left\{ 1, \frac{P(z')q(z|z')}{P(z)q(z'|z)} \right\}, \quad (2.19)$$

where  $P(z)$  is proportional to the target distribution (likelihood and prior). For RW-Metropolis the proposal distribution is Gaussian centred around the current value with variance given,  $z' \sim N(z, \sigma)$ . In this case the  $q$  ratio in the acceptance probability cancels leaving

$$p_{\text{acc}} = \min \left\{ 1, \frac{P(z')}{P(z)} \right\}. \quad (2.20)$$

We now explain how to use this framework, within a Gibbs sampler to sample from  $(x, \theta)$ . Since we update  $x$  and  $\theta$  separately, when updating  $x$   $\theta$  is constant and

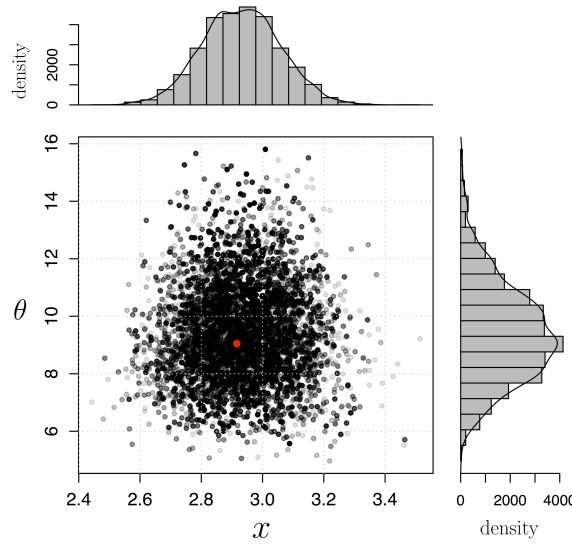


Figure 2.3: Posterior results from the simple model. Each grey point represents one sample from the posterior distribution, where overlapping values become darker. The red dot is the value obtained using MLE. The histograms, top and right, represent the individual posterior distributions of  $x$  and  $\theta$  respectively.

vice versa. Hence the target distribution reduces to the conditionals  $\pi(x|\theta, \mathbf{y})$  and  $\pi(\theta|x, \mathbf{y})$  and by using Bayes' theorem

$$\pi(x|\theta, \mathbf{y}) = \pi(\mathbf{y}|x, \theta)\pi(x) \quad \text{and} \quad \pi(\theta|x, \mathbf{y}) = \pi(\mathbf{y}|x, \theta)\pi(\theta). \quad (2.21)$$

The algorithm begins by selecting initial values  $x_{\text{cur}} = x_0$  and  $\theta_{\text{cur}} = \theta_0$ , sampling variances  $\sigma_x^2$  and  $\sigma_\theta^2$  and the number of iterations  $M$ . In each iteration, begin by updating  $x$ . Propose a new value  $x_{\text{can}} \sim N(x_{\text{cur}}, \sigma_x^2)$ , and accept the candidate value with probability  $\min\left(1, \frac{\pi(x_{\text{can}}|\theta_{\text{cur}}, \mathbf{y})}{\pi(x_{\text{cur}}|\theta_{\text{cur}}, \mathbf{y})}\right)$ . Next we update  $\theta$  by proposing a new value  $\theta_{\text{can}} \sim N(\theta_{\text{cur}}, \sigma_\theta^2)$ , and it is accepted probability  $\min\left(1, \frac{\pi(\theta_{\text{can}}|\mathbf{y}, x_{\text{cur}})}{\pi(\theta_{\text{cur}}|\mathbf{y}, x_{\text{cur}})}\right)$ . Often the calculations include large powers of variables, for example the Gamma ISI distribution (2.13) contains  $\gamma^\gamma$ , where  $\gamma$  can be large. To circumvent this issue the calculations are assessed on the logarithmic scale. The algorithm outputs a matrix of dimension  $2 \times M$ , where column  $i$  contains the values  $(x_i, \theta_i)$  the  $i$ th sample. Often, the initial

$M_{\text{burn}}$  iterations are removed to allow the Markov chain time to converge onto the stationary distribution.

In particular for the Gamma ISI distribution with parameter  $\gamma$ , we have the likelihood given by (2.17). Expanding the priors for  $x$  and  $\theta$  in (2.16) we have

$$\pi(x) = \frac{\beta_x^{\alpha_x}}{\Gamma(\alpha_x)} x^{\alpha_x-1} \exp\{-\beta_x x\} \quad \text{and} \quad \pi(\gamma) = \frac{\beta_\gamma^{\alpha_\gamma}}{\Gamma(\alpha_\gamma)} \gamma^{\alpha_\gamma-1} \exp\{-\beta_\gamma \gamma\}. \quad (2.22)$$

Recall we only require the density up to a multiplicative constant hence removing constants and rearranging we get the acceptance probability for  $x$

$$p_{\text{acc}}^x = \frac{x_{\text{can}}^{\alpha_x} \exp\{-x_{\text{can}}(\beta_x + y_1 + (T - y_N))\} \prod_{i=2}^N x_{\text{can}}^{\gamma} \exp\{-\gamma x_{\text{can}}(y_i - y_{i-1})\}}{x_{\text{cur}}^{\alpha_x} \exp\{-x_{\text{cur}}(\beta_x + y_1 + (T - y_N))\} \prod_{i=2}^N x_{\text{cur}}^{\gamma} \exp\{-\gamma x_{\text{cur}}(y_i - y_{i-1})\}}, \quad (2.23)$$

$$= \frac{x_{\text{can}}^{\alpha_x + (N-1)\gamma} \exp\{-x_{\text{can}}(\beta_x + y_1 + (T - y_N) + \gamma(y_N - y_1))\}}{x_{\text{cur}}^{\alpha_x + (N-1)\gamma} \exp\{-x_{\text{cur}}(\beta_x + y_1 + (T - y_N) + \gamma(y_N - y_1))\}}, \quad (2.24)$$

where  $\gamma$  is equal to the current value at the start of the iteration. In the case of the Gamma ISI distribution it is possible to simplify the method stated above. Note that the conditional  $\pi(x|\gamma, \mathbf{y})$  (i.e the numerator in (2.24)) is the kernel of a Gamma distribution. This implies that when we update  $x$  we can draw from distribution  $\Gamma(\alpha_x + (N-1)\gamma, \beta_x + y_1 + (t - y_N) + \gamma(y_N - y_1))$ . However, we use RW-Metropolis for  $x$  here as this is required for other ISI distributions.

Similarly the acceptance probability for  $\gamma$  is

$$p_{\text{acc}}^\gamma = \frac{\gamma_*^{\alpha_\gamma-1} e^{-\beta_\gamma \gamma_*} \prod_{i=2}^N \frac{(\gamma_* x)^{\gamma_*}}{\Gamma(\gamma_*)} [(y_i - y_{i-1})]^{\gamma_*-1} \exp(-\gamma_* x (y_i - y_{i-1}))}{\gamma^{\alpha_\gamma-1} e^{-\beta_\gamma \gamma} \prod_{i=2}^N \frac{(\gamma x)^{\gamma}}{\Gamma(\gamma)} [(y_i - y_{i-1})]^{\gamma-1} \exp(-\gamma x (y_i - y_{i-1}))}, \quad (2.25)$$

$$= \frac{\gamma_*^{\alpha_\gamma-1} \exp\{-\gamma_*(\beta_\gamma + y_n - y_1)\} \frac{(\gamma_* x)^{(N-1)\gamma_*}}{\Gamma(\gamma_*)^{N-1}} \prod_{i=2}^N (y_i - y_{i-1})^{\gamma_*-1}}{\gamma^{\alpha_\gamma-1} \exp\{-\gamma(\beta_\gamma + y_n - y_1)\} \frac{(\gamma x)^{(N-1)\gamma}}{\Gamma(\gamma)^{N-1}} \prod_{i=2}^N (y_i - y_{i-1})^{\gamma-1}}, \quad (2.26)$$

where we swap notation  $\gamma_{\text{cur}} = \gamma$  and  $\gamma_{\text{can}} = \gamma_*$  for clarity in the equation, and  $x$  is the current value in the iteration. Due the large number of exponents and exponentials the calculations are computed on the logarithmic scale. We draw  $u \sim U[0, 1]$ , then accept the candidate value if  $\log u < \log p_{\text{acc}}$ .

In figure 2.3 we give an example for simulated data with 58 spikes generated between 0 and 20s, from the model defined by  $x(t) = 3$  and  $\gamma = 9$ . We apply the



MCMC algorithm beginning with  $x_0 = 1$  and  $\gamma_0 = 1$  and both variances  $\sigma_x, \sigma_\gamma$  also set to 1. We ran the algorithm for 40000 iterations and removed the first 10000. Each grey dot represents the values of a single iteration of the algorithm, such that if dots overlap the colour becomes darker. We also calculate the MLE which corresponds to the red point in the scatter plot. We see that we recover the model parameters well, with most of the density centred about the true values. Furthermore, the MLE occurs at approximately the values with the most probability from the MCMC as expected.

### 2.4.2 Piecewise Constant Prior

We now generalise the simple model, where the intensity function can now take multiple values via a step function. We assume that a priori the intensity function is piecewise constant (PWC) and is defined by the times (change points) where the intensity changes and the values of the intensity between these times (heights). This is a reasonable assumption to make in step-change experiments, where the stimulus is exchanged during the experiment. Moreover, if cells experienced a constant stimulation, it is reasonable to assume  $x(t)$  remains constant over large periods but the value may change due to biological phenomena, such as a change in ER  $\text{Ca}^{2+}$  load. Even if a priori we believe the intensity varies continuously, every continuous function can be approximated by a step function, taking the number of change points to be large.

To apply Bayesian inference we require a prior distribution over the space of PWC functions and require an MCMC method to sample from the posterior. Notice that with this functional form the number of parameters that describe the intensity function changes depending on the number of change points. For example if  $x(t) = 1$ , we have 0 change points and 1 height and a total of 1 parameter, whereas if  $x(t) = 0.5$  if  $t < 20$  and 1.5 otherwise, we have 1 change point and 2 heights and a total of 3 parameters. Hence we require an MCMC method which allows jumps between parameter spaces with varying dimensions. We implement a Reversible jump MCMC (RJMCMC) technique PJ (1995), which extends standard MCMC to allow for parameter spaces of

varying dimension. In our case, the sample space is  $\mathcal{X} = \cup_k \mathcal{X}_k$ , where  $\mathcal{X}_k$  corresponds to the parameter space with  $k$  change points and  $k + 1$  heights. The method works by either changing parameter space or parameter values in the current space at each iteration, full details are in the preceding section.

The advantage of this method compared to the Gaussian Process prior (see section 1.4.3) is that it requires less computational demand. However in vitro the belief is that cells are exposed to time varying (continuous) stimuli and the intensity function should inherit this behaviour, thus with PWC we can only approximate this continuous change with step functions.

#### 2.4.2.1 Prior Distribution

We assume that the intensity function  $x(t)$  on  $[0, T]$  is a step function. Suppose that there are  $k$  change points  $0 = s_0 < s_1 < \dots < s_k < s_{k+1} = T$  and that the height on each interval  $[s_j, s_{j+1})$  is  $h_j$  for  $j = 0, 1, \dots, k$ . Hence we have

$$x(t) = \sum_{j=0}^k h_j \mathbf{1}_{[s_j, s_{j+1})}(t). \quad (2.27)$$

We assume that a priori  $k$  has a Poisson distribution with rate  $\lambda$  conditioned on  $k \leq k_{\max}$

$$p(k) = \frac{\lambda^k e^{-\lambda}}{k!} \bigg/ \sum_{i=0}^{k_{\max}} \frac{\lambda^i e^{-\lambda}}{i!}. \quad (2.28)$$

This means that the most likely number of change points is  $\lambda$  and we limit the maximum number to be  $k_{\max}$ . Suppose we have  $k$  change points  $s_1, \dots, s_k$  a priori they are distributed as the even-numbered order statistics of  $2k + 1$  points uniformly and independently distributed on  $[0, T]$ . To visualise imagine we were to simulate the position of the change points, we would sample  $2k + 1$  points in the interval  $[0, T]$ . Then order the points from smallest to largest to obtain  $\{t_i\}_{i=1}^{2k+1}$ , and set  $s_i = t_{2i}$  for  $i \in 1, \dots, k$ . This prior for the change points was used by Green (1995) to probabilistically penalise short intervals and improve the spacing of the change points.

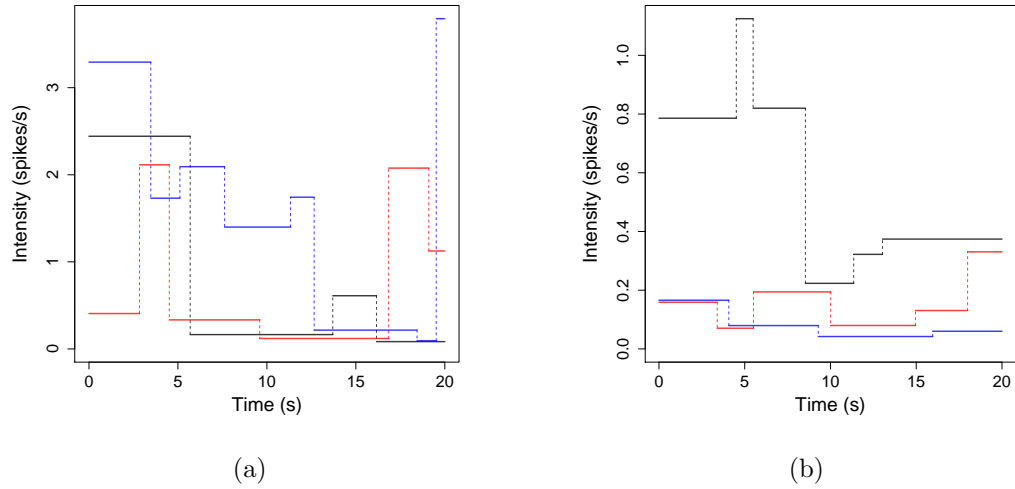


Figure 2.4: Realisations from the PWC prior. Where  $T = 20$ ,  $k_{\max} = 10$ ,  $\lambda = 5$ ,  $\kappa = 1$  and  $\mu = 1$ . In (A) the heights are independent and in (B) the heights follow the martingale structure.

For the heights we have two options for the prior. Firstly,  $h_0, h_1, \dots, h_k$  have independent  $\Gamma(\kappa, \mu)$  distributions. Secondly, the heights can follow a martingale structure (Following Arjas and Gasbarra (1994) E and D (1994)) where we assume that  $h_0 \sim \text{Gamma}(\kappa, \mu)$  and that given  $h_0, \dots, h_{i-1}$ ,  $h_i \sim \text{Gamma}(\kappa, \mu_i)$  where  $\mu_i = \kappa/h_{i-1}$  so that  $\mathbb{E}[h_i|h_{i-1}] = h_{i-1}$ . This should have the affect of smoothing  $x(t)$ . In Figure 2.4 we see that the heights with the martingale structure do indeed vary less then those assumed to be independent. Indeed, we see in figure 1.4 that with the martingale structure consecutive heights have more similar values than if the heights are assumed to be independent.

### 2.4.2.2 Posterior Computation

In this section we will show how the RJMCMC proposes new step functions, and whether or not to accept the new function. In Algorithm 2 we have an outline of the MCMC algorithm where a priori the intensity function is PWC. Furthermore, the

algorithm states when to update the ISI parameter, with more details in section 1.4.4. Note that the algorithm requires the following inputs: spikes  $\mathbf{y}$ ; the ISI distribution (e.g. Gamma, Weibull. etc); the number of iterations to compute  $M$  along with training period of  $M_{\text{burn}}$  iterations; the parameters of the PWC prior  $\lambda, k_{\text{max}}, \kappa, \mu$ ; the variance  $\sigma_\theta^2$  used in the RW-Metropolis algorithm and the prior type for heights (either independent or martingale structure). It outputs  $M$  samples of  $\mathbf{x}(t)$ , where each sample is composed of position of change points and corresponding heights, where the number of change points and heights can vary in each sample. In addition to  $M$  samples of the ISI parameter  $\theta$ .

We will begin by considering the heights to be independent a priori, and discuss how the martingale structure alters the formulas afterwards. At each iteration of the algorithm we make one of three types of updates:

1. Birth of a change point.
2. Death of a change point.
3. Within model updates (moving existing change points and change heights).

These occur respectively with probabilities  $b_k, d_k$  and  $1 - b_k - d_k$ , where

$$b_k = c \min\{1, p(k+1)/p(k)\}, \quad d_k = c \min\{1, p(k-1)/p(k)\}. \quad (2.29)$$

Above  $p(k)$  is the probability mass function of a Poisson random variable with rate  $\lambda$  conditioned with maximum value  $k_{\text{max}}$  and  $c$  is chosen such that  $\max_k(b_k + d_k) = 0.9$  PJ (1995). This is done to make sure within model changes occur at a frequent rate, such each iteration is not always a change in parameter space.

For all of the updates we need to calculate the likelihood ratio, i.e given the data the likelihood that the data comes from the given intensity function  $x(t)$  and ISI parameter  $\theta$ . So suppose we initially have intensity function  $x(t)$  and candidate intensity function is  $x'(t)$  then the likelihood ratio is

$$(\text{likelihood ratio}) = \frac{\pi(\mathbf{y}|x', \theta)}{\pi(\mathbf{y}|x, \theta)}. \quad (2.30)$$

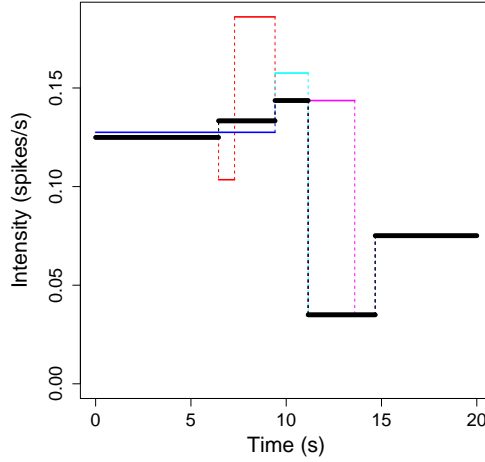


Figure 2.5: Illustration of the possible moves at each iteration of the RJMCMC. The black line corresponds to the original step function. The blue line shows a candidate function when a change point is removed. Red when a new change point is added. Cyan when a height is changed. Magenta when a change point moves.

Note that  $\theta$  takes the same value in both numerator and denominator, thus any terms purely in  $\theta$  cancel. In particular for the Gamma ISI distribution the likelihood function  $p(\mathbf{y}|x, \theta)$  is given in (2.13), with the ISI parameter becoming  $\gamma$ . This gives the ratio

$$\frac{x'(y_1)e^{-X'(y_0, y_1)}e^{-X'(y_N, T)} \prod_{i=2}^N x'(y_i) [X'(y_{i-1}, y_i)]^{\gamma-1} \exp(-\gamma X'(y_{i-1}, y_i))}{x(y_1)e^{-X(y_0, y_1)}e^{-X(y_N, T)} \prod_{i=2}^N x(y_i) [X(y_{i-1}, y_i)]^{\gamma-1} \exp(-\gamma X(y_{i-1}, y_i))}, \quad (2.31)$$

where  $X'$  is the integrated candidate intensity function.

#### 2.4.2.3 Step 1. Birth of a change point

Firstly, we want to describe how to calculate the position of the new change point and how this changes the heights. Assume initially we have change points  $s_1, \dots, s_k$  and corresponding heights  $h_0, \dots, h_k$ . A new change point  $s^*$  is chosen uniformly on

$[0, T]$ . The new change point lies in an existing interval  $[s_j, s_{j+1})$  with probability 1. So our set of change points becomes  $\{s_1, \dots, s_j, s^*, s_{j+1}, \dots, s_k\}$ .

Next we want to propose the new heights. We will replace height  $h_j$  (this corresponds to the interval  $[s_j, s_{j+1})$ ) with two new heights  $h'_j$  and  $h'_{j+1}$ , using a weighted geometric mean so that

$$(s^* - s_j) \log(h'_j) + (s_{j+1} - s^*) \log(h'_{j+1}) = (s_{j+1} - s_j) \log(h_j) \quad (2.32)$$

and

$$\frac{h'_{j+1}}{h'_j} = \frac{1 - u}{u} \quad (2.33)$$

where  $u \sim U(0, 1)$ . Hence the candidate heights are  $h_0, \dots, h_{j-1}, h'_j, h'_{j+1}, h_{j+1}, \dots, h_k$ .

The acceptance probability takes the form

$$p_{\text{acc}} = \min\{1, (\text{likelihood ratio}) \times (\text{prior ratio}) \times (\text{proposal ratio}) \times (\text{Jacobian})\}. \quad (2.34)$$

The likelihood and prior terms correspond to the target distribution, the proposal ratio is analogous to the transition ratio in the Metropolis-Hastings algorithm (2.19), and the Jacobian is the transformation of coordinates from the original to the new space, e.g from  $k$  changepoints to  $k + 1$ .

Begin by deriving the prior ratio. In words we have:

$$(\text{prior ratio}) = (\text{prior ratio for heights}) \times (\text{prior ratio for change points}). \quad (2.35)$$

Consider the heights. We assumed that the heights come from independent  $\Gamma(\kappa, \mu)$  distributions. Thus, the probability of heights  $h_0, \dots, h_k$  is

$$p(h_0, \dots, h_k) = \prod_{j=0}^k \frac{\mu^\kappa}{\Gamma(\kappa)} h_j^{\kappa-1} e^{-\mu h_j}. \quad (2.36)$$

Consequently, the ratio takes the form  $p(h_0, \dots, h'_j, h'_{j+1}, \dots, h_k) / p(h_0, \dots, h_j, \dots, h_k)$ , notice that all heights apart from  $h_j, h'_j$  and  $h'_{j+1}$  cancel in the division. Hence, the ratio reduces to  $p(h'_j, h'_{j+1})/p(h_j)$ . Solving this gives:

$$(\text{prior ratio for heights}) = \frac{\left( \left( \frac{\mu^\kappa}{\Gamma(\kappa)} \right)^2 (h'_j)^{\kappa-1} e^{-\mu h'_j} (h'_{j+1})^{\kappa-1} e^{-\mu h'_{j+1}} \right)}{\left( \frac{\mu^\kappa}{\Gamma(\kappa)} h_j^{\kappa-1} e^{-\mu h_j} \right)}, \quad (2.37)$$

$$= \frac{\mu^\kappa}{\Gamma(\kappa)} \left( \frac{h'_j h'_{j+1}}{h_j} \right)^{\kappa-1} \exp\{-\mu(h'_j + h'_{j+1} - h_j)\}. \quad (2.38)$$

Next consider the change points. Each individual change point is uniformly distributed on  $[0, T]$ , i.e  $s_i \sim U[0, T]$ , with density  $p(s_i) = 1/T$  for  $i = 1, \dots, k$ . Hence considering the joint distribution we get

$$p(s_1, \dots, s_k) = (2k+1)! \int \dots \int \frac{1}{T^{2k+1}} \mathbb{1}_{\{s_0 < t_1 < s_1 < t_3 < s_2 < t_5 \dots\}} dt_1 dt_3 \dots dt_{2k+1}, \quad (2.39)$$

$$= \frac{(2k+1)!}{T^{2k+1}} \int \dots \int \mathbb{1}_{\{s_0 < t_1 < s_1 < t_3 < s_2 < t_5 \dots\}} dt_1 dt_3 \dots dt_{2k+1}, \quad (2.40)$$

$$= \frac{(2k+1)!}{T^{2k+1}} (s_1 - s_0) \dots (s_{j+1} - s_j) \dots (s_{k+1} - s_k), \quad (2.41)$$

where  $s_0 = 0$  and  $s_{k+1} = T$ , are the start and end times of the experiment. Furthermore,  $(2k+1)!$  comes from the number of permutations of  $2k+1$  points. Hence the prior probability of having  $k$  change points at times  $\{s_i\}_{i=1}^k$  is  $(2.28) \times (2.41)$ , where we leave the conditioned Poisson as  $p(k)$  for simplicity in the following equations. Similar to the heights ratio, all the factors of the form  $(s_{i+1} - s_i)$ ,  $i = 1, \dots, k$  cancel except for the intervals that have been changed by the introduction of the new point  $s^*$ . Thus giving

$$(\text{prior ratio for change points}) = \frac{\left( p(k+1) \frac{(2k+3)!}{T^{2k+3}} (s^* - s_j)(s_{j+1} - s^*) \right)}{\left( p(k) \frac{(2k+1)!}{T^{2k+1}} (s_{j+1} - s_j) \right)}, \quad (2.42)$$

$$= \frac{p(k+1)}{p(k)} \frac{(2k+2)(2k+3)}{T^2} \frac{(s^* - s_j)(s_{j+1} - s^*)}{(s_{j+1} - s_j)}. \quad (2.43)$$

Putting together equation (2.38) and (2.43) we get that the prior ratio is

$$\begin{aligned} (\text{Prior ratio}) &= \frac{p(k+1)}{p(k)} \frac{(2k+2)(2k+3)}{T^2} \frac{(s^* - s_j)(s_{j+1} - s^*)}{(s_{j+1} - s_j)} \\ &\quad \times \frac{\mu^\kappa}{\Gamma(\kappa)} \left( \frac{h'_j h'_{j+1}}{h_j} \right)^{\kappa-1} \exp\{-\mu(h'_j + h'_{j+1} - h_j)\}. \end{aligned} \quad (2.44)$$

Next we want to compute the proposal ratio. As stated previously, this is analogous to the ratio  $q(x'|x)/q(x|x')$ , where  $x$  is the current value of the intensity function and  $x'$  the candidate value. In this case where we are now jumping between different dimensions this becomes  $j(x')/(j(x)q(u))$ , where  $j(x)$  is the probability of choosing  $x$  and  $q(u)$  is the choice of the parameter  $u$  used to generate the new point in the space. Hence,  $j(x')$  is the move from  $x'$  to  $x$  which is a death at the point  $s^*$  and  $j(x)$  is the move from  $x$  to  $x'$  which is the birth of point  $s^*$ .

$$(\text{Proposal Ratio}) = \frac{j(x')}{j(x)q(u)} = \frac{d_{k+1} \frac{1}{k+1}}{b_k \frac{1}{T-0} \frac{1}{1-0}} = \frac{d_{k+1} T}{b_k (k+1)}. \quad (2.45)$$

The Jacobian calculates the change in co-ordinates from our original space to the candidate space. Due to the nature of adding a single change point this reduces to the change from  $h_j$  (and  $u$ ) to  $h'_j$  and  $h'_{j+1}$ . We first need to rearrange equations (2.32) and (2.33) to  $h'_j$  and  $h'_{j+1}$  in terms of  $h_j$  and  $u$ . This is done initially taking the exponential

$$(h'_j)^{(s^* - s_j)} (h'_{j+1})^{(s_{j+1} - s^*)} = h_j^{(s_{j+1} - s_j)}. \quad (2.46)$$

Then we substitute  $h'_{j+1} = h'_j \frac{1-u}{u}$ , thus getting the exponent for  $h'_j$  to be  $(s_{j+1} - s_j)$ .

Hence, rearranging for  $h'_j$  we get

$$h'_j = h_j \left( \frac{u}{1-u} \right)^{\frac{s_{j+1} - s^*}{s_{j+1} - s_j}}. \quad (2.47)$$

Similarly, for  $h'_{j+1}$  we have

$$h'_{j+1} = h_j \left( \frac{1-u}{u} \right)^{\frac{s^* - s_j}{s_{j+1} - s_j}}. \quad (2.48)$$

For ease of notation we let  $r = (s_{j+1} - s^*)/(s_{j+1} - s_j)$ , and ergo  $1 - r = (s^* - s_j)/(s_{j+1} - s_j)$  from now on.



$$J = \begin{vmatrix} \frac{\partial h'_j}{\partial h_j} & \frac{\partial h'_j}{\partial u} \\ \frac{\partial h'_{j+1}}{\partial h_j} & \frac{\partial h'_{j+1}}{\partial u} \end{vmatrix}, \quad (2.49)$$

$$= \begin{vmatrix} \left(\frac{u}{1-u}\right)^r & h_j r \left(\frac{u}{1-u}\right)^{r-1} \frac{1}{(1-u)^2} \\ \left(\frac{1-u}{u}\right)^{1-r} & h_j (1-r) \left(\frac{1-u}{u}\right)^{-r} \frac{-1}{u^2} \end{vmatrix}, \quad (2.50)$$

$$= h_j (1-r) \frac{u^{2r-2}}{(1-u)^{2r}} + h_j r \frac{u^{2r-2}}{(1-u)^{2r}}, \quad (2.51)$$

$$= h_j \frac{u^{2r-2}}{(1-u)^{2r}}, \quad (2.52)$$

$$= h_j \frac{[u^{r-1}u + u^{r-1}(1-u)]^2}{(1-u)^{2r}}, \quad (2.53)$$

$$= \frac{[h_j \left(\frac{u}{1-u}\right)^r + h_j \left(\frac{1-u}{u}\right)^{1-r}]^2}{h_j}, \quad (2.54)$$

$$= \frac{(h'_j + h'_{j+1})^2}{h_j}. \quad (2.55)$$

#### 2.4.2.4 Step 2. Death of a change point

Select a change point  $s_{j+1}$  uniformly at random from the existing change points for deletion, and propose new height  $h'_j$  over the interval  $(s_j, s_{j+2}) = (s'_j, s'_{j+1})$  by

$$(s_{j+1} - s_j) \log(h_j) + (s_{j+2} - s_{j+1}) \log(h_{j+1}) = (s'_{j+1} - s'_j) \log(h'_j). \quad (2.56)$$

The acceptance probability is obtained by inversion of the terms for the corresponding birth probability. Hence the prior ratio is

$$\begin{aligned} \frac{p(k-1)}{p(k)} \frac{T^2}{2k(2k+1)} \frac{(s'_{j+1} - s'_j)}{(s_{j+2} - s_{j+1})(s_{j+1} - s_j)} \\ \times \frac{\Gamma(\kappa)}{\mu^\kappa} \left( \frac{h'_j}{h_j h_{j+1}} \right)^{\kappa-1} \exp\{-\mu(h'_j - h_j - h_{j+1})\}, \end{aligned} \quad (2.57)$$

the proposal ratio is

$$\frac{b_{k-1}k}{d_k T}, \quad (2.58)$$

and the Jacobian is

$$\frac{h'_j}{(h_j + h_{j+1})^2}. \quad (2.59)$$

#### 2.4.2.5 Step 3. Within model updates

When we are not proposing a birth or a death, the following within model updates take place.

Move an existing change point: provided that  $k > 0$  a change point  $s_j$  is chosen uniformly at random from the existing change points and a new position  $s'_j$  is proposed uniformly at random on  $[s_{j-1}, s_{j+1}]$  and accepted with probability

$$\min \left\{ 1, (\text{likelihood ratio}) \times \frac{(s_{j+1} - s'_j)(s'_j - s_{j-1})}{(s_{j+1} - s_j)(s_j - s_{j-1})} \right\} \quad (2.60)$$

Change a height: a height  $h_j$  is chosen uniformly at random from the existing heights and a new height  $h'_j$  is proposed such that  $\log(h'_j/h_j)$  is uniformly distributed on  $[-1/2, 1/2]$ , and accepted with probability

$$\min \left\{ 1, (\text{likelihood ratio}) \times (h'_j/h_j)^\kappa \exp\{-\mu(h'_j - h_j)\} \right\} \quad (2.61)$$

The second term in equations (2.60) and (2.61) come from the prior distribution for the change points and heights respectively.

In the steps above we have used the assumption that the heights are a priori independent and identically distributed. We will now discuss how the calculations change with the martingale structure. This change does not affect the mechanism of which move is used in an iteration nor how we propose the candidate function. Changes only occur via the prior ratio in the acceptance probability.

First, consider a birth of a new point. Recall that the original heights are  $h_0, \dots, h_k$  and the proposed heights are  $h_0, \dots, h_{j-1}, h'_j, h'_{j+1}, h_{j+1}, \dots, h_k$  and the only change comes from replacing height  $h_j$  with heights  $h'_j$  and  $h'_{j+1}$ , where for the martingale structure we have  $h_i \sim \Gamma(\kappa, \frac{\kappa}{h_{i-1}})$ , for  $i > 0$  and  $h_0 \sim \Gamma(\kappa, \mu)$ . Notice that even with the more complex structure most of the terms will cancel in the prior ratio. In fact, the probability of each height only depends on the previous height, thus the ratio simplifies to  $p(h'_j, h'_{j+1}, h_{j+1})/p(h_j, h_{j+1})$ . We will need to split into 2 cases, for when  $j < k$  and  $j = k$  since  $h_{j+1}$  does not exist if  $j = k$ . Furthermore, care needs to be taken when only one height exists, as the ratio simplifies further to  $p(h'_1, h'_2)/p(h_1)$ .

First let's consider the simpler case when  $j = k$ , explicitly writing the prior ratio gives the following

$$\frac{\left(\frac{(\kappa/h_{k-1})^\kappa}{\Gamma(\kappa)} (h'_k)^{\kappa-1} \exp\left\{-\frac{\kappa}{h_{k-1}} h'_k\right\}\right) \left(\frac{(\kappa/h'_k)^\kappa}{\Gamma(\kappa)} (h'_{k+1})^{\kappa-1} \exp\left\{-\frac{\kappa}{h'_k} h'_{k+1}\right\}\right)}{\left(\frac{(\kappa/h_{k-1})^\kappa}{\Gamma(\kappa)} h_k^{\kappa-1} \exp\left\{-\frac{\kappa}{h_{k-1}} h_k\right\}\right)} \quad (2.62)$$

Then collecting terms and simplifying gives

$$(\text{Prior ratio}) = \frac{\kappa^\kappa}{\Gamma(\kappa)} \frac{1}{h_k} \left(\frac{h'_{k+1}}{h'_k}\right)^{\kappa-1} \exp\left\{\kappa \left(\frac{h_k - h'_k}{h_{k-1}} - \frac{h'_{k+1}}{h'_k}\right)\right\} \quad (2.63)$$

Now consider the case where  $j < k$ , for simplicity define  $h_{-1} = \mu$  because  $h_0 \sim \Gamma(\kappa, \mu)$ .

$$\frac{\left(\frac{(\kappa/h_{j-1})^\kappa}{\Gamma(\kappa)} (h'_j)^{\kappa-1} \exp\left\{-\frac{\kappa}{h_{j-1}} h'_j\right\}\right) \left(\frac{(\kappa/h'_j)^\kappa}{\Gamma(\kappa)} (h'_{j+1})^{\kappa-1} \exp\left\{-\frac{\kappa}{h'_j} h'_{j+1}\right\}\right) \times \left(\frac{(\kappa/h'_{j+1})^\kappa}{\Gamma(\kappa)} (h_{j+1})^{\kappa-1} \exp\left\{-\frac{\kappa}{h'_{j+1}} h_{j+1}\right\}\right)}{\left(\frac{(\kappa/h_{j-1})^\kappa}{\Gamma(\kappa)} (h_j)^{\kappa-1} \exp\left\{-\frac{\kappa}{h_{j-1}} h_j\right\}\right) \left(\frac{(\kappa/h_j)^\kappa}{\Gamma(\kappa)} (h_{j+1})^{\kappa-1} \exp\left\{-\frac{\kappa}{h_j} h_{j+1}\right\}\right)} \quad (2.64)$$

Simplifying

$$\frac{\kappa^\kappa}{\Gamma(\kappa)} \frac{h_j}{h'_j + h'_{j+1}} \exp\left\{\kappa \left(-\frac{h'_j - h_j}{h_{j-1}} - \frac{h'_{j+1}}{h'_j} - \frac{h_{j+1}}{h'_{j+1}} + \frac{h_{j+1}}{h_j}\right)\right\} \quad (2.65)$$

Similar to before, for a death of a change point we just inverse the prior for the birth. There is no affect for changing the value of a change point. Next consider

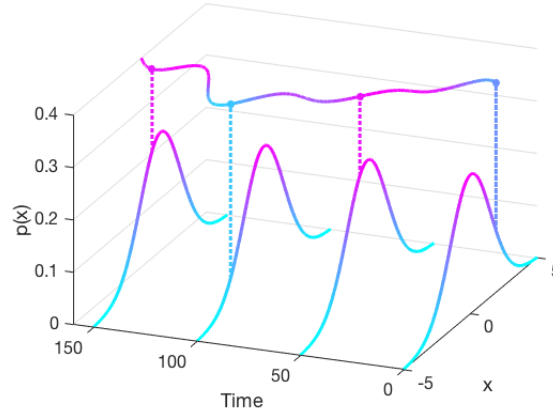


Figure 2.6: Realisation of a Gaussian Process (top line), with mean 0 and SE covariance. The bell shape curves show the probability distributions of the values the realisation can take at times 0, 50, 100 and 150, where the probability is coloured such that light blue indicates low probability and magenta high probability. The dashed lines indicated the probability of the given value that the GP takes at that time.

changing a height, again we must split into two cases when  $j < k$  and  $j = k$  repeating the argument above we get the prior ratio to be

$$\left(\frac{h'_j}{h_j}\right)^{\kappa-1} \exp\left\{-\frac{\kappa}{h_{j-1}}(h'_j - h_j)\right\}, \quad (2.66)$$

for  $j = k$ , and

$$\frac{h_j}{h'_j} \exp\left\{-\frac{\kappa}{h_{j-1}}(h'_j - h_j) - \kappa h_{j+1} \left(\frac{1}{h'_j} - \frac{1}{h_j}\right)\right\}, \quad (2.67)$$

for  $j < k$ . This replaces  $(h'_j/h_j)^\kappa \exp\{-\mu(h'_j - h_j)\}$  in (2.61).

### 2.4.3 Gaussian Process Prior

A Gaussian process (GP) is a collection of random variables where any finite number of which have a multivariate Gaussian distribution. A GP is fully specified by its

mean and covariance function. Often the mean is taken to be 0, in which case a GP is defined solely by its covariance function. There are many potential covariance functions that could be used such as: constant, linear, or Gaussian noise C and C (2006). However, we use the widely applied standard squared exponential (SE) covariance function,

$$\Sigma(t_1, t_2) = \sigma_f^2 \exp \left[ -\frac{(t_1 - t_2)^2}{2l^2} \right] + \delta(t_1 - t_2) \sigma_v^2. \quad (2.68)$$

This is because realisations from the GP with SE covariance are smooth when  $\sigma_v^2 = 0$ , which is a property we desire for the intensity function, to mimic the varying stimulus that cells experience in vitro. Three hyper parameters define the shape of the intensity function, as can be seen in figure 2.7.  $\sigma_f^2$  determines the function's average distance from the mean (fig 2.7A, 2.7C);  $l$  controls the 'smoothness' of the intensity, where smaller  $l$  leads to an intensity that varies more (fig 2.7A, 2.7B) and  $\sigma_v^2$  is additional Gaussian noise (fig 2.7A, 2.7D).

A Gaussian process can be viewed as a probability distribution over the space of functions on  $[0, T]$ , the domain of interest. For example in figure 2.6 the top line is a draw from the GP with mean 0 and SE covariance function, each curve underneath corresponds to the normal distribution that that point comes from, where magenta corresponds to a higher probability of the value occurring and blue a lower probability.

To sample from a GP we discretise time by partitioning the interval into  $\mathbf{t} = \{t_i\}_{i=1}^M$ , where  $t_i = i\Delta$  such that  $\Delta$  is the time step used. From these points we can retrieve the covariance matrix  $\Sigma = \{\Sigma(t_i, t_j)\}_{t_i, t_j \in \mathbf{t}}$ , assuming the GP has mean  $\mu$ , a vector of length M, we can approximate a realisation from the GP by sampling from the multivariate normal with mean  $\mu$  and covariance matrix  $\Sigma$ .

Furthermore, we can calculate the probability of a function  $f(t)$  by discretising to  $\mathbf{f} = \{f_i\}_{i=1}^M$  with  $f_i = f(t_i)$  and evaluating the probability density function for the Multivariate normal distribution, i.e  $\mathcal{N}(\mathbf{f}|\mu, \Sigma)$ . Next we consider how to use a GP as a prior for our intensity function  $x(t)$ , as noted above we need to discretise the intensity function for practical purposes, thus  $x(t)$  becomes  $\mathbf{x} = \{x_i\}_{i=1}^M$ , where

$x_i = x(t_i)$ . Often the time step  $\Delta$  is taken to be the inverse of the recording frame rate of the experiment such that spikes occur on the discretised grid. However, care needs to be taken if the spike data is altered before fitting the model, such as removing linear trends. In this situation, the altered spike times do not occur on the grid created from the experiment frame rate. To combat this, we extend the grid to include the spike times, for example if the original grid is  $\mathbf{t} = \{t_i\}_{i=1}^M$ , we extend to  $\tilde{\mathbf{t}} = \mathbf{t} \cup \mathbf{y}$ , where  $\mathbf{y}$  are the spike times.

For the Gamma ISI distribution we obtain the likelihood function with the discretisation to be

$$\pi(\mathbf{y}|\mathbf{x}, \gamma) = x_1 e^{-\hat{X}_{0,1}} e^{-\hat{X}_{n,n+1}} \prod_{i=2}^N \frac{\gamma x_{l_i}}{\Gamma(\gamma)} \left[ \gamma \hat{X}_{i-1,i} \right]^{\gamma-1} e^{-\gamma \hat{X}_{i-1,i}}, \quad (2.69)$$

where  $\hat{X}_{i,j} = \Delta \sum_{k=l_i}^{l_j} x_k$  with  $l_0 = 0$ , and  $l_{n+1} = T/\Delta$ . Since we are only considering updating  $x$  we can remove terms purely in  $\gamma$ , to get

$$\pi(\mathbf{y}|\mathbf{x}, \gamma) = x_1 e^{-\hat{X}_{0,1}} e^{-\hat{X}_{n,n+1}} \prod_{i=2}^N x_{l_i} \left[ \hat{X}_{i-1,i} \right]^{\gamma-1} e^{-\gamma \hat{X}_{i-1,i}}. \quad (2.70)$$

Note that we require the intensity function to be positive, as it is impossible to have a negative spiking rate. Thereby, we will assume that the logarithm of the intensity function has a Gaussian Process prior with SE covariance, that is

$$\log \mathbf{x} \sim \mathcal{N}(\mathbf{0}, \Sigma), \quad (2.71)$$

where  $\Sigma$  is defined in (2.68).

It is not necessary to take the logarithm, however when sampling the posterior distribution of  $x(t)$  many proposed intensity functions would not be positive and would be rejected instantaneously, requiring a longer mixing time.

Hence in general, with the GP prior the unknowns are the model parameters and the hyper parameters of the GP, i.e  $\{x, \theta, \sigma_f^2, l, \sigma_n^2\}$ . However, we elect to fix  $\sigma_f^2 = 1000$  to allow for a large range of intensity values, equivalent to adding a non-informative prior about the value of the intensity. To determine realisations from the GP we need to be able to sample from a large multivariate normal distribution which requires inversion of the covariance matrix. We want to set  $\sigma_n^2 = 0$  to retain

the smoothness property of the intensity function, however doing such renders the matrix computationally singular. Hence, we set  $\sigma_n^2 = 0.0001$ , small, hereby making the inversion possible with negligible random noise. This is equivalent to adding a nugget term to the Gaussian process A (2010). A priori we have little information to inform us on a sensible value for  $l$ . For this reason we keep it as a model parameter. Thus for the GP prior we introduce a new variable  $l$  which we also need to predict. As  $\mathbf{x}$  requires a complex MCMC algorithm we sample  $l$  separately from  $x$ , utilising Gibbs Method.

In algorithm 3 we give an outline of the method used, where the required inputs are: the spikes  $\mathbf{y}$ ; the ISI distribution used (e.g. Gamma, Weibull, etc); the experiment length  $T$ ; the number of samples we need  $M$  and the initial number of iterations to remove  $M_{\text{burn}}$ ; the discretisation  $\mathbf{t}$  used for the GP; step-size parameter  $\omega$  used in the under-relaxed method and the variances  $\sigma_\theta^2, \sigma_l^2$  used in the RW-Metropolis algorithms for  $\theta$  and  $l$  respectively..

In the upcoming sections we describe in detail the under-relaxed method used to update  $\mathbf{x}$ , followed by the RW-Metropolis algorithm used for  $l$ .

#### 2.4.3.1 Step 1. Posterior Computation for $\mathbf{x}$

We use a under-relaxed algorithm, based on a Metropolis-Hastings method RM (1999). Begin by choosing an initial value  $\mathbf{x}_0$ . As  $x(t)$  describes the spiking rate, we set  $\mathbf{x}_0$  to be the mean spike rate from the experiment. For example if an experiment ran for 100s and a total of 58 spikes were recorded then  $\mathbf{x}_0 = \mathbf{0.58}$ , the vector of length  $M$  with all values equal 0.58. Consider an iteration, with current intensity  $\mathbf{x}_{\text{cur}}$ , we propose new intensity  $\mathbf{x}_{\text{can}}$  by  $\mathbf{x}_{\text{can}} = e^{\mathbf{x}_{\text{can}}^*}$  where

$$\mathbf{x}_{\text{can}}^* = \sqrt{1 - \omega^2} \log(\mathbf{x}_{\text{cur}}) + \omega \boldsymbol{\nu}, \quad \boldsymbol{\nu} \sim \mathcal{N}(\mathbf{0}, \Sigma). \quad (2.72)$$

Above  $\omega \in [-1, 1]$  is a step-size parameter, where the closer  $\omega$  is to zero the more the candidate value will be similar to the current value. Whereas, if  $\omega$  is close to one the candidate value will be similar to a draw from the prior distribution. We use a value close to zero, often  $\omega = 0.01$ , to improve the mixing of the MCMC (i.e

candidate values get accepted more often).

Next we need to determine the acceptance probability,

$$p_{\text{acc}} = \min \left\{ 1, \frac{\pi(\mathbf{y}|\mathbf{x}_{\text{can}}, \theta) \pi(\mathbf{x}_{\text{can}}) q(\mathbf{x}_{\text{cur}}|\mathbf{x}_{\text{can}})}{\pi(\mathbf{y}|\mathbf{x}_{\text{cur}}, \theta) \pi(\mathbf{x}_{\text{cur}}) q(\mathbf{x}_{\text{can}}|\mathbf{x}_{\text{cur}})} \right\} \quad (2.73)$$

where for the Gamma ISI distribution  $\pi(\mathbf{y}|\mathbf{x}, \theta)$  comes from (2.70), the prior  $\pi(\mathbf{x})$  is defined in (2.71) and  $q(r|s)$  is the probability of transitioning from intensity  $s$  to  $r$ . Due to the proposal regime this reduces to only the marginals since the prior and  $q$  ratios cancel out.

For simplicity we show that the prior and transition probabilities cancels for the case when  $x$  is a constant. Note from (2.72) we have  $q(x_{\text{can}}|x_{\text{cur}}) = \sqrt{1-\omega^2} \log(x_{\text{cur}}) + \omega\nu$ , where  $\nu \sim N(0, \sigma^2)$ , this means the transition from the current to the candidate value is normally distributed about  $\sqrt{1-\omega^2} \log(x_{\text{cur}})$  with variance  $\omega^2 \sigma^2$ , i.e  $q(x_{\text{can}}|x_{\text{cur}}) = N(\sqrt{1-\omega^2} \log(x_{\text{cur}}), \omega^2 \sigma^2)$ . In this case the prior on  $\log(x)$  is a gaussian centred at 0 with variance  $\sigma^2$ . Substituting this into (2.73) gives

$$\frac{\pi(\mathbf{y}|\mathbf{x}_{\text{can}}, \theta) \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ \frac{-\log(x_{\text{can}})^2}{2\sigma^2} \right\}}{\pi(\mathbf{y}|\mathbf{x}_{\text{cur}}, \theta) \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ \frac{-\log(x_{\text{cur}})^2}{2\sigma^2} \right\}} \frac{\frac{1}{\sqrt{2\pi\omega^2\sigma^2}} \exp \left\{ \frac{-(\log(x_{\text{cur}}) - \sqrt{1-\omega^2} \log(x_{\text{can}}))^2}{2\omega^2\sigma^2} \right\}}{\frac{1}{\sqrt{2\pi\omega^2\sigma^2}} \exp \left\{ \frac{-(\log(x_{\text{can}}) - \sqrt{1-\omega^2} \log(x_{\text{cur}}))^2}{2\omega^2\sigma^2} \right\}}. \quad (2.74)$$

Then we note that the contributions by the prior and transition probabilities cancel; by combining the exponentials, equating the denominator and expanding the square term in the transition probabilities, thereby leaving

$$p_{\text{acc}} = \min \left\{ 1, \frac{\pi(\mathbf{y}|\mathbf{x}_{\text{can}}, \theta)}{\pi(\mathbf{y}|\mathbf{x}_{\text{cur}}, \theta)} \right\}. \quad (2.75)$$

The advantage of this proposal mechanism is that the acceptance probability does not require any contributions from the multivariate normal, and in particular the computational expensive matrix inversion. Since most ISI distributions contain large exponents we convert to the logarithmic scale and accept candidate value  $\mathbf{x}_{\text{can}}$  if

$$\log u < \log \pi(\mathbf{y}|\mathbf{x}_{\text{can}}, \theta) - \log \pi(\mathbf{y}|\mathbf{x}_{\text{cur}}, \theta), \quad (2.76)$$



where  $u \sim U[0, 1]$ . For the Gamma ISI distribution taking the log of equation (2.70) and substituting into (2.76), we accept if

$$\log u < \log(x'_1) - \hat{X}'_{0,1} - \hat{X}'_{n,n+1} \sum_{i=2}^N \log(x'_{l_i}) + (\gamma - 1) \log(\hat{X}'_{i-1,i}) - \gamma \hat{X}'_{i-1,i} \\ - \left( \log(x_1) - \hat{X}_{0,1} - \hat{X}_{n,n+1} \sum_{i=2}^N \log(x_{l_i}) + (\gamma - 1) \log(\hat{X}_{i-1,i}) - \gamma \hat{X}_{i-1,i} \right) \quad (2.77)$$

### 2.4.3.2 Step 2. Estimating the length scale

We begin by applying Baye's theorem once more where initially we consider both  $\mathbf{x}$  and  $l$

$$\pi(\mathbf{x}, l | \mathbf{y}) \propto \pi(\mathbf{y} | \mathbf{x}, l) \pi(\mathbf{x}, l), \quad (2.78)$$

$$= \pi(\mathbf{y} | \mathbf{x}) \pi(\mathbf{x} | l) \pi(\mathbf{x}) \pi(l), \quad (2.79)$$

where (2.79) is obtained by conditioning the likelihood on  $l$ , and noticing that the likelihood (2.70), is not directly influenced by  $l$ . Furthermore, our priors for  $\mathbf{x}$  and  $l$  are independent. Since we only require the marginal up to multiplicative constant we have

$$\pi(l | \mathbf{x}, \mathbf{y}) \propto \pi(\mathbf{x} | l) \pi(l). \quad (2.80)$$

Note that  $l$  is defined on  $(0, \infty)$  and a priori we have little idea what value  $l$  will take. Hence we apply a non-informative exponential prior is suitable, i.e  $l \sim \exp(\lambda)$ , where we take  $\lambda$  small.

Recall that  $\log \mathbf{x} \sim \mathcal{N}(\mathbf{0}, \Sigma)$ , where  $\Sigma$  depends on the hyperparameters  $\{l, \sigma_f^2, \sigma_n^2\}$  and that  $\sigma_f^2$  and  $\sigma_n^2$  are fixed. Thus we have

$$\pi(\mathbf{x} | l) \propto \frac{1}{|\Sigma|^{1/2}} \exp \left[ -\frac{1}{2} \mathbf{x}_*^T \Sigma^{-1} \mathbf{x}_* \right], \quad (2.81)$$

where  $\mathbf{x}_* = \log \mathbf{x}$ . Hence, collectively we get

$$\pi(l | \mathbf{y}, \mathbf{x}) \propto \frac{1}{|\Sigma|^{1/2}} \exp \left[ -\frac{1}{2} \mathbf{x}_*^T \Sigma^{-1} \mathbf{x}_* \right] e^{-\lambda l} \quad (2.82)$$

We implement a RW-Metropolis algorithm to sample the length scale. We choose initial value  $l_0$ , then at each iteration we propose a candidate value  $l_{can}$  generated from a normal distribution centred about the current value with variance  $\sigma_l^2$ , i.e  $l_{can} \sim N(l_{cur}, \sigma_l^2)$ . Recall for this algorithm the transition probabilities cancel as per (2.19). Thus by taking the logarithm  $l_{can}$  is accepted if

$$\log u < \log \pi(l_{can}|\mathbf{y}, \mathbf{x}) - \log \pi(l_{cur}|\mathbf{y}, \mathbf{x}) \quad (2.83)$$

where  $u \sim U[0, 1]$ .

#### 2.4.4 Updating ISI Parameter

In the above two sections we have considered how to sample the posterior for the intensity function  $x(t)$ , whilst we consider the ISI parameter  $\theta$  to be fixed. Now we consider how to sample the ISI parameter  $\theta$ , where we fix the value of  $x(t)$  whilst updating  $\theta$ . Applying Bayes' theorem we have

$$\pi(\theta, x|\mathbf{y}) = \frac{\pi(\mathbf{y}, |x, \theta)\pi(\theta, x)}{\pi(\mathbf{y})}. \quad (2.84)$$

Since we use a Gibbs sampler, we concentrate on the conditional distribution  $\pi(\theta|x, \mathbf{y})$ . The priors on  $x$  and  $\theta$  are independent and we only require the distribution up to a multiplicative constant giving

$$\pi(\theta|\mathbf{x}, \mathbf{y}) \propto \pi(\mathbf{y}|x, \theta)\pi(\theta). \quad (2.85)$$

We require a prior distribution on  $\theta$ . For all ISI distributions their parameters are defined on  $(0, \infty)$ . Hence any distribution on  $(0, \infty)$  could be suitable. However, throughout we use a non-informative Gamma prior, i.e  $\theta \sim \text{Gamma}(1, 0.01)$ , unless specified otherwise. Since we are only updating  $\theta$  we remove any terms that are purely in  $x$  in the likelihood  $\pi(\mathbf{y}|x, \gamma)$ , as these will cancel when we take the likelihood ratio. For clarification, this is shown below for the Gamma ISI distribution where we have  $\theta = \gamma$ , and we use the more generalised prior  $\gamma \sim \text{Gamma}(\alpha, \beta)$

$$\pi(\gamma|x, \mathbf{y}) \propto \pi(\mathbf{y}|x, \gamma)\pi(\gamma), \quad (2.86)$$

$$\begin{aligned} &= \textcolor{red}{x}(y_1)e^{-X(y_0, y_1)}e^{-X(y_N, T)} \prod_{i=2}^N \frac{\gamma \textcolor{red}{x}(y_i)}{\Gamma(\gamma)} [\gamma X(y_{i-1}, y_i)]^{\gamma-1} e^{-\gamma X(y_{i-1}, y_i)} \\ &\quad \times \frac{\gamma^\beta}{\Gamma(\alpha)} \gamma^{\alpha-1} e^{-\beta\gamma}, \end{aligned} \quad (2.87)$$

$$\propto \gamma^{\alpha-1} e^{-\beta\gamma} \prod_{i=2}^N \frac{\gamma^\gamma}{\Gamma(\gamma)} [X(y_{i-1}, y_i)]^{\gamma-1} e^{-\gamma X(y_{i-1}, y_i)}. \quad (2.88)$$

For the Gamma ISI distribution the terms in red were removed, as they do not depend on  $\gamma$ . For computational ease we will work on the logarithmic scale, as there are several exponents above. Hence taking the logarithm we have

$$\begin{aligned} \log \pi(\gamma|x, \mathbf{y}) &\propto (\gamma(N-1) + \alpha - 1) \log \gamma - (N-1) \log \Gamma(\gamma) - \beta\gamma \\ &\quad + \sum_{i=2}^N \left[ (\gamma-1) \log(X(y_{i-1}, y_i)) - \gamma X(y_{i-1}, y_i) \right]. \end{aligned} \quad (2.89)$$

In order to sample the ISI parameter  $\theta$  we use a RW-Metropolis Algorithm. We begin with initial value  $\theta_0$ , which depends on the ISI distribution. For the Gamma ISI the initial value is taken to be 1. Suppose the current sampled value is  $\theta_{\text{cur}}$ , in the next iteration we propose a new candidate value  $\theta_{\text{can}}$  drawn from  $N(\theta_{\text{cur}}, \sigma_\theta^2)$ , Where  $\sigma_\theta^2$  is chosen ad hoc. Since we follow a RW-Metropolis algorithm, the transition probabilities cancel and the acceptance probability reduces to the likelihood ratio, that is

$$p_{\text{acc}} = \min \left\{ 1, \frac{\pi(\theta_{\text{can}}|x, \mathbf{y})}{\pi(\theta_{\text{cur}}|x, \mathbf{y})} \right\}. \quad (2.90)$$

Converting to the logarithmic scale we accept the candidate value if

$$\log u < \log \pi(\theta_{\text{can}}|x, \mathbf{y}) - \log \pi(\theta_{\text{cur}}|x, \mathbf{y}), \quad (2.91)$$

where  $u \sim U[0, 1]$ .

## 2.5 Model Assessment

Suppose that we have calculated the mean posterior intensity function  $\hat{x}(t)$  and mean ISI parameter  $\hat{\theta}$  for a spike sequence. Recall that  $\text{Ca}^{2+}$  spiking is modelled by a point process. In the literature, there is a time rescaling theorem that states: any point process with an integrable conditional intensity function can be transformed into a homogeneous Poisson process with unit rate EN et al. (2002); Daley and Vere-Jones (2008). By applying the theorem we can assess the quality of the estimates by comparing the transformed ISI times to i.i.d exponential random variables with rate 1.

The conditional intensity function  $q(t)$  is defined by

$$q(t) = \lim_{\Delta t \rightarrow 0} \frac{\mathbb{E} \{N(t, t + \Delta t) | H_t\}}{\Delta t}, \quad (2.92)$$

where  $H_t$  denotes the spike history up to time  $t$  and  $N(s, t)$  is the number of spikes in the interval  $(s, t)$ . The conditional intensity  $q(t)$  function specifies the instantaneous rate at which spikes occur given the history of the process up to time  $t$ . For the inhomogeneous Poisson model we have  $q(t) = x(t)$ , as it is history independent. It is more convenient to define the condition intensity function in terms of the ISI probability density

$$q(t) = \frac{p(y_k, t | x, \theta)}{1 - \int_{y_k}^t p(u, y_k | x, \theta) du}, \quad (2.93)$$

for  $t > y_k$  and  $k = 1, 2, \dots, N$ . R et al. (2001); Daley and Vere-Jones (2008)

The integrated conditional intensity function  $R(t) = \int_0^t q(u) du$  is continuous and non-decreasing. This stems from the fact that the intensity function  $x(t)$  is non-negative, and thus  $q(t)$  is also non-negative. This means that the mapping  $t \rightarrow R(t)$  is injective. By the time rescaling theorem, rescaled spike times  $R(y_i)$   $i = 1, \dots, N$  come from a Poisson process with unit rate. To check if our model agrees with this introduce  $\tau_k$  as the inter event times of the transformed point process:

$$\tau_k = R(y_k) - R(y_{k-1}). \quad (2.94)$$

If the ISI model is correct the  $\tau_k$  will be independent and identically distributed exponential random variables with rate 1, i.e  $\tau_k \sim E(1)$ . EN et al. (2002); Daley and

Vere-Jones (2008)

For the model assessment we use two graphical methods, namely Quantile-Quantile (Q-Q) and Kolmogorov-Smirnov (K-S) plots EN et al. (2002); MB and R (1968). Q-Q plots are produced by plotting experimental quantiles (ordered  $\tau_k$ ) against model quantiles ( $\tilde{\tau} = -\log(1 - s_k)$ , where  $s_k = \frac{k-0.5}{N}$ ). If the model is correct then experimental quantiles against model quantiles should form a straight line through the origin with slope 1.

K-S plots are produced by transforming the  $\tau_k$  into  $u_k = 1 - e^{-\tau_k}$ , then  $u_k$  are i.i.d uniform random variables on  $[0,1]$ . If the model is correct plotting  $u_k$  against  $s_k = \frac{k-0.5}{N}$  should be scattered around the line through the origin with slope 1.

These two graphical methods are complementary, since Q-Q plots are more sensitive at the tails of the distribution and K-S plots are more sensitive near the centre of the distribution, as they are based on the cumulative distribution.

Above we have assumed we have only one data set, i.e. one spike sequence with corresponding posterior  $\hat{x}(t)$  and  $\hat{\theta}$ . However, we can repeat the analysis with multiple data sets to ascertain whether our method is functioning properly for multiple spike sequences under different conditions. We can either plot all quantiles of multiple spikes on the same plot and visualise how the points lie around the line of slope one. Or for each data set calculate the slope of the quantiles, then compose a box plot of all the data sets, this should be centred around the value one with small range / interquartile range.

## 2.6 Example with simulated data

We now apply the techniques discussed to simulated data. We simulate spike sequences using algorithm 1 from a Gamma ISI distribution with parameter  $\gamma = 10$  with intensity constant at  $x(t) = 2$ , for 20s, where  $K = 8000$  steps were used. The expected number of spikes in one sequence is  $XX$  spikes. We then use these sequences to infer model parameters using constant, PWC and GP priors for the intensity function assuming the

ISIs are Gamma distributed. In all instances we give  $\gamma$  a  $\Gamma(1, 0.01)$  prior distribution. When  $x$  is assumed constant a priori it is distributed as a  $\Gamma(1, 0.01)$ . The inputs for the simple model are: 20000 iterations with a training period of 10000 iterations, and variances  $\sigma_x = \sigma_\theta = 1$ . For the PWC prior the inputs are:  $M = 10000$ ,  $M_{\text{burn}} = 5000$ ,  $k_{\text{max}} = 30$ ,  $\lambda = 20$ ,  $\kappa = 1$ ,  $\mu = 2$  and  $\sigma_\theta^2 = 1$ .

For the GP prior the inputs are:  $T = 20$ ,  $M = 6000$ ,  $M_{\text{burn}} = 4000$ ,  $\mathbf{t} = \{0, 0.05, 0.1, \dots, 20\}$ ,  $\omega = 0.01$ ,  $\sigma_\theta^2 = 1$  and  $\sigma_t^2 = 0.5$ .

Figure 2.9 (A) is a raster plot of 40 simulated spike sequences. In figure 2.9(B,C) we have the posterior results for a single spike sequence, which can be seen as the rugs in (B). In (B) we show the posterior results for  $x$  and in (C) the posterior results for  $\gamma$ . The colours red, blue and magenta correspond to the posterior results for PWC prior, GP prior and constant  $x$  respectively. We find that the posterior distributions  $x$  and  $\theta$  are close to the model parameters ( $x = 2$ ,  $\theta = 10$ ) the spike sequence was simulated from, as expected. Note that the intensities are all higher (2.2) than the value simulated from (black line), this is due to the simulated spike sequence having 44 spikes. Although not plotted the MLE values for the sequence were  $x = 2.214$  and  $\theta = 11.76$ . Even though the PWC and GP prior allows for large changes in the intensity, we see that both remain approximately at 2.2, with slight variation due to variation in the ISIs. In (C) we plot the histograms of  $\gamma$  generated by MCMC for the three priors, we see that whilst the distributions vary slightly all are centered around the true value of  $\gamma = 10$ . We have shown we recover parameter values for one sequence, however this could just be luck. To consolidate that we recover the model parameters we calculate the MLE for 200 spike sequences. In (D) we plot the MLEs for each individual spike sequence. We notice that the dots are scattered around the model parameters as expected. The variance of the dots around the model parameters we simulated from is due to the random nature of the simulated spikes.

Consider a time-dependent intensity function, we choose  $x(t) = 2 \cos(t/2) + 1.1$ , where the simulation time remains at 20s. We repeat the steps for the constant intensity above, simulating spike sequences and then fitting the model with these

spike sequences. The inputs for inference remains the same for each prior. In figure 2.10 (A) we show 40 simulated spike sequences with the time-dependent intensity function. The raster plot clearly shows that the spiking intensity varies in time in a wave manner. In figure 2.10(B,C) we show the posterior distribution for the intensity for two different spike sequences, shown as rugs in both plots. The black line corresponds to the intensity the spikes were simulated from. Both the posterior distributions from PWC (red) and GP (blue) priors recover the intensity.

**Algorithm 2:** Inference for PWC prior.

**Input:** Spikes  $\mathbf{y}$ , ISI distribution,  $M, M_{\text{burn}}, \lambda, k_{\text{max}}, \kappa, \mu, \sigma_{\theta}^2$  and the prior type for heights.

**Output:**  $M$  samples of the intensity  $x(t)$ , given by heights  $\mathbf{h}$  and change points  $\mathbf{s}$ , and ISI parameter  $\theta$ .

Calculate  $c$  such that  $\max_k(b_k + d_k) = 0.9$ ;

Set initial values  $\mathbf{h}^{(0)}, \mathbf{s}^{(0)}$  and  $\theta^{(0)}$ ;

**for**  $i$  *in* 1 *to*  $(M + M_{\text{burn}})$  **do**

    Update the intensity function.

        Calculate  $b_k$  and  $d_k$ ;

        Draw  $u \sim U(0, 1)$ ;

**(Step 1) if**  $u < b_k$  **then**

            Propose a new change point and corresponding heights;

            Accept the proposed values with probability given by the acceptance probability, update  $\mathbf{h}^{(i+1)}$  and  $\mathbf{s}^{(i+1)}$  ;

**(Step 2) else if**  $b_k < u < b_k + d_k$  **then**

            Propose a death of a current change point and corresponding heights;

            Accept the proposed values with probability given by the acceptance probability, update  $\mathbf{h}^{(i+1)}$  and  $\mathbf{s}^{(i+1)}$ ;

**(Step 3) else**

            Propose a move for an existing change point;

            Accept the proposed values with probability given by the acceptance probability, update  $\mathbf{s}^{(i+1)}$ ;

            Propose a change of an existing height;

            Accept the proposed values with probability given by the acceptance probability, update  $\mathbf{h}^{(i+1)}$ ;

    Update the ISI parameter.

        Use RW-Metropolis algorithm to update  $\theta$  (section 1.4.4);

Return  $\mathbf{h}, \mathbf{s}$  and  $\theta$  with initial  $M_{\text{burn}}$  iterations removed;



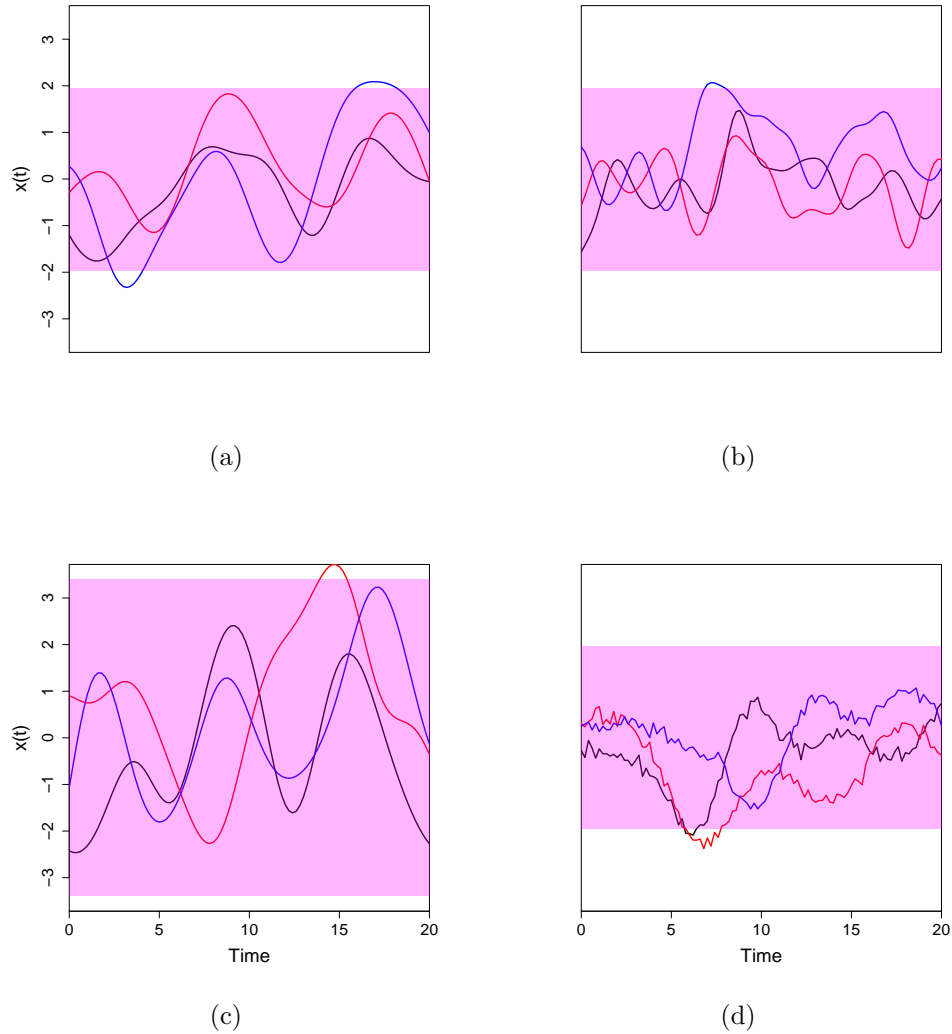


Figure 2.7: Illustration of how varying hyper parameters changes realisations of a Gaussian Process with mean 0. We draw 3 samples from the following GP priors: (A)  $\sigma_f^2 = 1, l = 2, \sigma_v^2 = 0$  (B)  $\sigma_f^2 = 1, l = 1, \sigma_v^2 = 0$  (C)  $\sigma_f^2 = 2, l = 2, \sigma_v^2 = 0$  (D)  $\sigma_f^2 = 1, l = 2, \sigma_v^2 = 0.01$ . The magenta region indicates the 95% confidence intervals. In (B) we notice that decreasing the length scale leads the realisation to vary more frequently with time. In (C) increasing  $\sigma_f^2$  increases the range of values taken (variance). Finally, (D) shows noise is controlled by  $\sigma_v^2$ .

---

**Algorithm 3:** Inference for GP prior.

---

**Input:**  $\mathbf{y}$ , ISI distribution,  $T$ ,  $M$ ,  $M_{\text{burn}}$ ,  $\mathbf{t}$ ,  $\omega$ ,  $\sigma_\theta^2$ ,  $\sigma_l^2$ **Output:**  $M$  samples of intensity  $\mathbf{x}$ , ISI parameter  $\theta$  and length scale  $l$ .Set initial values  $\mathbf{x}^{(0)}$ ,  $\theta^{(0)}$  and  $l^{(0)}$ ;**for**  $i$  *in* 1 *to*  $(M + M_{\text{burn}})$  **do**    **Step 1.** Propose  $\mathbf{x}_{\text{can}}$  and calculate acceptance probability using underrelaxed method, update  $\mathbf{x}^{(i)}$ ;    **Step 2.** Propose  $l_{\text{can}}$  and calculate acceptance probability using RW-Metropolis, update  $l^{(i)}$ ;    **Step 3.** Update  $\theta_{\text{can}}$  and calculate acceptance probability using RW-Metropolis, update  $\theta^{(i)}$  (section 1.4.4);Return:  $\mathbf{x}$ ,  $\theta$  and  $l$  with initial  $M_{\text{burn}}$  iterations removed;

---

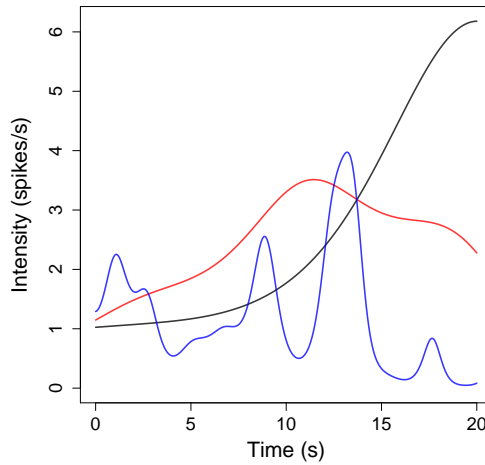


Figure 2.8: Draws from the GP prior for the intensity, where we assume the logarithm of  $x(t)$  has a GP prior. Where the black, red and blue lines correspond to length scales 10, 5 and 1. To illustrate that a priori we don't know the value of  $l$  and it will be fitted using MCMC.

---

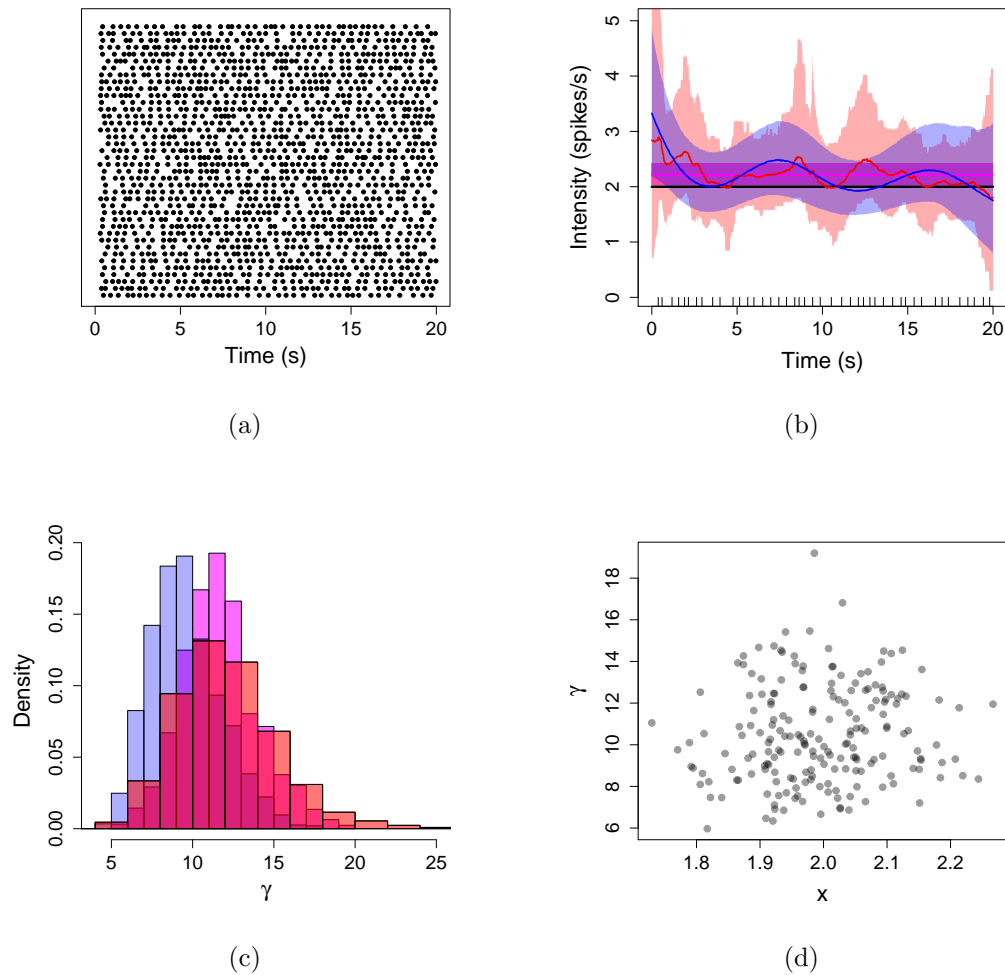


Figure 2.9: Results from simulated data with constant intensity. (A) Raster plot of 40 of the simulated spike sequences. (B) Posterior intensity function results for a single spike sequence, where the black line corresponds to the intensity the spikes were sampled from. In addition red, blue and magenta lines correspond to posterior intensity computed via PWC, GP and simple priors respectively, where the shaded areas correspond to the 95% confidence regions. (C) Histogram of the posterior samples of the ISI parameter  $\gamma$  for the same spike sequence as (B) where the colours represent the priors PWC (red), GP (blue) and simple (magenta). (D) Scatterplot of the MLEs for 200 spike sequences, where each dot corresponds to the MLE of a single spike sequence.

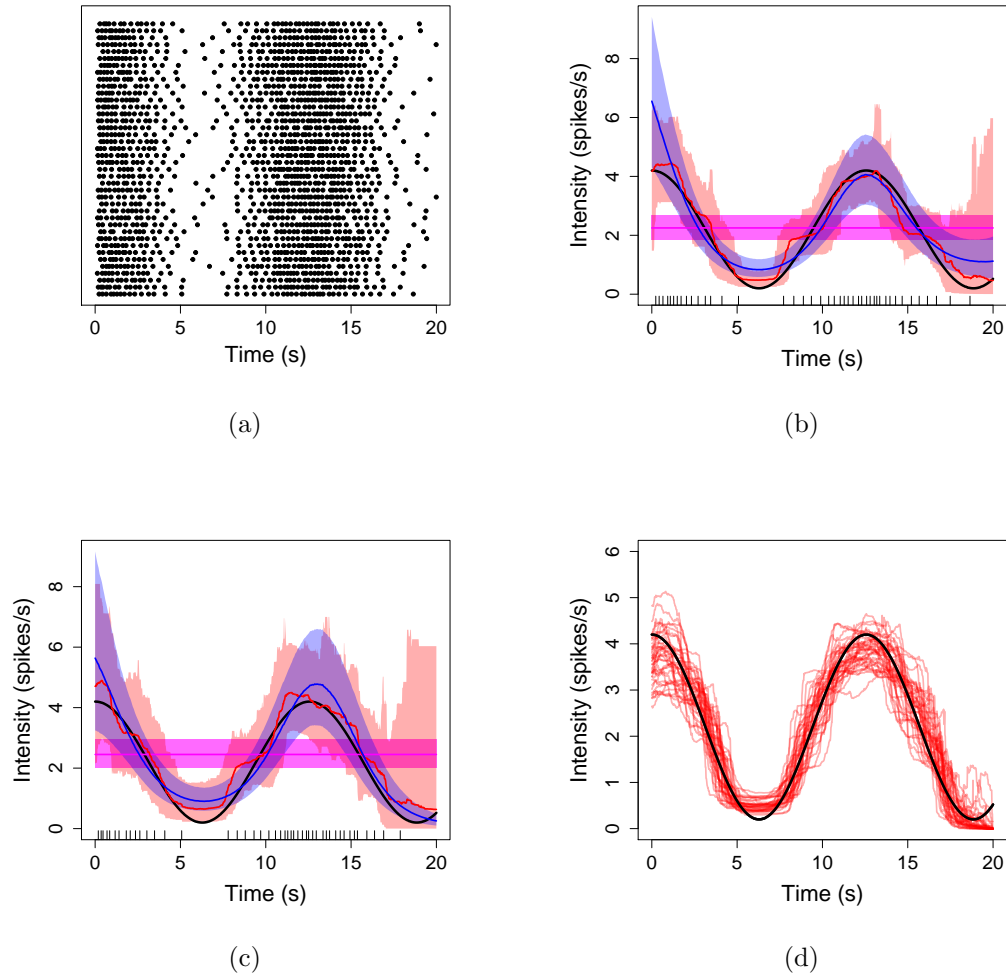


Figure 2.10: Results from simulated data with time-varying intensity. (A) Raster plot of 40 of the simulated spike sequences. (B,C) Posterior intensity function results for a single spike sequence, where the black line corresponds to the intensity the spikes were sampled from. In addition red, blue and magenta lines correspond to posterior intensity computed via PWC, GP and simple priors respectively, where the shaded areas correspond to the 95% confidence regions. (D) Scatterplot of average posterior intensity function for 40 spike sequences, where each red line corresponds to the mean posterior intensity function for a single spike sequence.

## CHAPTER 3

---

### Non-trivial Model Properties

---

The model we have created applies methods well-known in the field of statistics, however applying these methods to our model brings unique issues/decisions. In this chapter we will discuss these complications and explain and justify the decisions we make going forward.

#### 3.1 ISI Distribution

In chapter 2 we have assumed that the ISI distribution follows an inhomogeneous gamma distribution as detailed in equation -. This is the model that was used to best effect by Titulane et al when applying similar methods to HEK293 cells stimulated with Carbohol. In their paper they tested three potential ISI distributions, which they named the inhomogeneous Poisson

$$p(y_i, y_{i-1} | \gamma) =,$$

the inhomogeneous Gamma

$$p(y_i, y_{i-1} | \gamma) =,$$

and the inhomogenous Inverse Gaussian

$$p(y_i, y_{i-1} | \gamma) = .$$

These distributions were chosen because ... . However, this is clearly not a list of all possible distributions, and it is possible that another non-tested distribution would give a better fit. Thus, we will extend the number of ISI distributions to be considered to include a Log Normal distribution and a Weibull distribution. We want to consider these distributions because ... .

By the introduction of the intensity function we can see that the above distributions are generalisations of one-parameter Gamma and Inverse Gaussian distributions, namely  $\text{Gamma}(\gamma, \gamma)$  and  $\text{InverseGaussian}(-, -)$ . A natural question would be to ask whether we can generalise further and allow for the distributions two-parameter versions, such as the  $\text{Gamma}(\alpha, \beta)$ . However, by generalising the ISI distribution we create non-identifiability issues, where by scaling the ISI parameters with the intensity function we

### 3.1.1 Non-Identifiability issues with two parameter ISI distributions

To begin we will consider the Gamma distribution. Thus suppose the ISI distribution is defined by

$$p(y_i, y_{i-1} | x, \alpha\beta) = \frac{\beta x(y_i)}{\Gamma(\alpha)} [\beta X(y_{i-1}, y_i)]^{\alpha-1} \exp(-\beta X(y_{i-1}, y_i)), \quad (3.1)$$

where

$$X(s, t) = \int_s^t x(u) du, \quad \text{for any } s, t \in [0, T]. \quad (3.2)$$

### 3.1.2 Choosing one-parameter ISI distributions

Thus, we for each of the generalised two-parameter ISI distributions we need to choose a parameterisation to remove the non-identifiability issues.

Thus, the ISI distributions we will consider are the following:

### 3.1.3 Gamma distribution

Throughout we have assumed that the Gamma ISI distribution is as follows

$$p(y_i, y_{i-1}) = \frac{\gamma x(y_i)}{\Gamma(\gamma)} [\gamma X(y_{i-1}, y_i)]^{\gamma-1} \exp(-\gamma X(y_{i-1}, y_i)). \quad (3.3)$$

Which one can see a generalisation of a  $\text{Gamma}(\gamma, \gamma)$ . We have used this distribution because it was used in . However, can we generalise further and consider a  $\text{Gamma}(\alpha, \beta)$  ISI distribution? The answer is no. This is because the parameters  $x$  and  $\beta$  are non-identifiable.

Suppose our ISI interval is given by the following  $(\alpha, \beta)$  Gamma distribution.

$$p(y_i, y_{i-1}) = \frac{\beta x(y_i)}{\Gamma(\alpha)} [\beta X(y_{i-1}, y_i)]^{\alpha-1} \exp(-\beta X(y_{i-1}, y_i)), \quad (3.4)$$

where

$$X(s, t) = \int_s^t x(u) du, \quad \text{for any } s, t \in [0, T]. \quad (3.5)$$

Now we will show that this is equivalent to a  $(\gamma, \gamma)$  Gamma distribution by scaling the intensity function.

Let  $\gamma = \alpha$  and  $\kappa = \frac{\beta}{\alpha}$ , thus giving  $\beta = \gamma\kappa$ . Substitute into (2) gives

$$p(y_i, y_{i-1}) = \frac{\gamma\kappa x(y_i)}{\Gamma(\gamma)} [\gamma\kappa X(y_{i-1}, y_i)]^{\gamma-1} \exp(-\gamma\kappa X(y_{i-1}, y_i)). \quad (3.6)$$

Now define new intensity function by  $\tilde{x} = \kappa x$ , hence giving

$$\tilde{X}(s, t) = \int_s^t \tilde{x}(u) du = \int_s^t \kappa x(u) du = \kappa X(s, t), \quad \text{for any } s, t \in [0, T]. \quad (3.7)$$

Put this into (3) gives

$$p(y_i, y_{i-1}) = \frac{\gamma \tilde{x}(y_i)}{\Gamma(\gamma)} [\gamma \tilde{X}(y_{i-1}, y_i)]^{\gamma-1} \exp(-\gamma \tilde{X}(y_{i-1}, y_i)), \quad (3.8)$$

which we see is a  $(\gamma, \gamma)$  Gamma distribution.

Hence, where we have introduced a two-variable Gamma distribution we cannot retrieve initial choices of variables. I.e  $\beta$  is replaced as a variable by the intensity function. The choice of having  $\alpha = \beta$  is a sensible choice as this allows the expected number of spikes to not depend on the ISI parameters and only depends on  $x$ .

### 3.1.4 Inverse Gaussian distribution

Similarly we have not considered the full generalised Inverse Gaussian ISI distribution

$$p(y_i, y_{i-1}) = x(y_i) \left( \frac{\lambda}{2\pi X(y_{i-1}, y_i)^3} \right)^{0.5} \exp \left[ - \frac{\lambda(X(y_{i-1}, y_i) - \mu)^2}{2\mu^2 X(y_{i-1}, y_i)} \right]. \quad (3.9)$$

However, we again get non-identifiable variables, in this case if we multiple all the parameters by a constant we get an identical distribution.

Suppose we have an Inverse Gaussian ISI distribution with parameters  $(\lambda, \mu)$  and intensity function  $x$ . We want to show this is identical to an  $IG((k\lambda, k\mu))$  with intensity  $kx$  for any constant  $k$ . Firstly, note that the constant comes out the front of of the integral of the intensity function.

$$\int_s^t \kappa x(u) du = \kappa \int_s^t x(u) du = \kappa X(s, t), \quad \text{for any } s, t \in [0, T]. \quad (3.10)$$



Hence, we get

$$p(y_i, y_{i-1} | kx, k\lambda, k\mu) = kx(y_i) \left( \frac{k\lambda}{2\pi(kX(y_{i-1}, y_i))^3} \right)^{\frac{1}{2}} \exp \left[ - \frac{k\lambda(kX(y_{i-1}, y_i) - k\mu)^2}{2(k\mu)^2 kX(y_{i-1}, y_i)} \right], \quad (3.11)$$

$$= kx(y_i) \left( \frac{1}{k^2} \frac{\lambda}{2\pi X(y_{i-1}, y_i)^3} \right)^{\frac{1}{2}} \exp \left[ - \frac{k\lambda(k(X(y_{i-1}, y_i) - \mu))^2}{2k^3\mu^2 X(y_{i-1}, y_i)} \right], \quad (3.12)$$

$$= \frac{k}{k} x(y_i) \left( \frac{\lambda}{2\pi X(y_{i-1}, y_i)^3} \right)^{\frac{1}{2}} \exp \left[ - \frac{k^3 \lambda (X(y_{i-1}, y_i) - \mu)^2}{k^3 2\mu^2 X(y_{i-1}, y_i)} \right], \quad (3.13)$$

$$= x(y_i) \left( \frac{\lambda}{2\pi X(y_{i-1}, y_i)^3} \right)^{\frac{1}{2}} \exp \left[ - \frac{\lambda (X(y_{i-1}, y_i) - \mu)^2}{2\mu^2 X(y_{i-1}, y_i)} \right], \quad (3.14)$$

$$= p(y_i, y_{i-1} | x, \lambda, \mu) \quad \text{as required.} \quad (3.15)$$

### 3.1.5 Log Normal distribution

Next we shall consider the full Log Normal distribution

$$p(y_i, y_{i-1}) = \frac{x(y_i)}{X(y_{i-1}, y_i) \sigma \sqrt{2\pi}} \exp \left\{ - \frac{(\log X(y_{i-1}, y_i) - \mu)^2}{2\sigma^2} \right\} \quad (3.16)$$

However as in the above two cases we get non-identifiable variables. In this case, if we multiply the intensity by a factor of  $\kappa$  this is identical to subtracting  $\log \kappa$  to  $\mu$ . Suppose we have an Log Normal ISI distribution with parameters  $(\mu, \sigma)$  and intensity function  $\kappa x$ . We want to show this is identical to an  $\text{LN}((\mu + \log \kappa), \sigma)$  with intensity  $\kappa x$  for any constant  $\kappa$ .

$$p(y_i, y_{i-1} | \kappa x, \mu, \sigma) = \frac{\kappa x(y_i)}{\kappa X(y_{i-1}, y_i) \sigma \sqrt{2\pi}} \exp \left\{ -\frac{(\log \kappa X(y_{i-1}, y_i) - \mu)^2}{2\sigma^2} \right\}, \quad (3.17)$$

$$= \frac{x(y_i)}{X(y_{i-1}, y_i) \sigma \sqrt{2\pi}} \exp \left\{ -\frac{(\log X(y_{i-1}, y_i) + \log \kappa - \mu)^2}{2\sigma^2} \right\} \quad (3.18)$$

$$= \frac{x(y_i)}{X(y_{i-1}, y_i) \sigma \sqrt{2\pi}} \exp \left\{ -\frac{(\log X(y_{i-1}, y_i) - (\mu - \log \kappa))^2}{2\sigma^2} \right\} \quad (3.19)$$

$$= p(y_i, y_{i-1} | x, \mu - \log \kappa, \sigma) \quad \text{as required.} \quad (3.20)$$

### 3.1.6 Weibull distribution

Next we shall consider the Weibull distribution

$$p(y_i, y_{i-1} | k, \lambda) = \frac{x(y_i)k}{\lambda} \left( \frac{X(y_{i-1}, y_i)}{\lambda} \right)^{k-1} \exp \left\{ -\left( \frac{X(y_{i-1}, y_i)}{\lambda} \right)^k \right\} \quad (3.21)$$

However again we get non-identifiable variables. In this case, if we multiply the intensity by a factor of  $m$  this is identical to dividing  $\lambda$  by  $m$ .

Suppose we have an Weibull ISI distribution with parameters  $(k, \lambda)$  and intensity function  $mx$ . We want to show this is identical to an Weibull( $k, \frac{\lambda}{m}$ ) with intensity  $x$  for any constant  $m$ .

$$p(y_i, y_{i-1} | mx, k, \lambda) = \frac{mx(y_i)k}{\lambda} \left( \frac{mX(y_{i-1}, y_i)}{\lambda} \right)^{k-1} \exp \left\{ -\left( \frac{mX(y_{i-1}, y_i)}{\lambda} \right)^k \right\} \quad (3.22)$$

$$= \frac{x(y_i)k}{\lambda/m} \left( \frac{X(y_{i-1}, y_i)}{\lambda/m} \right)^{k-1} \exp \left\{ -\left( \frac{X(y_{i-1}, y_i)}{\lambda/m} \right)^k \right\} \quad (3.23)$$

$$= p(y_i, y_{i-1} | x, k, \lambda/m) \quad \text{as required.} \quad (3.24)$$

## 3.2 Speed of the GP

In this section we discuss how a naive implementation of the GP prior in chapter 1 will lead to a computationally expensive algorithm. We will explain different methods to improve the algorithm, and justify our implementation.

In this framework we assume a priori that the intensity function  $x(t)$  comes from a GP. To allow for computations we discretise time, often set to the frame rate of experimental data. Thus, the prior reduced to a multivariate normal distribution, over a large number of indexes, often greater than 1000 steps. To sample from the posterior intensity we utilise an under-relaxed method, where in each iteration we propose a candidate intensity  $x_{\text{can}}$  depending on the current value  $x_{\text{cur}}$  by

$$x_{\text{can}} = \sqrt{1 - \omega^2} \log(x_{\text{cur}}) + \omega \nu, \quad \nu \sim \mathcal{N}(\mathbf{0}, \Sigma)$$

Thus, in each iteration we must generate a draw ( $\nu$ ) from the multivariate normal distribution. This requires the inversion and determinant of the covariance matrix to be calculated, whose dimension is the discretisation chosen for the intensity, thus we need to be able to invert  $1000 \times 1000$  matrices, quickly. There exists different methods to compute the matrix decomposition such as via eigenvalues?? or a choleski decomposition. However, these methods tend to be time consuming once the dimension increases.

We have considered two different methods to improve the time taken, namely by implementing a projection when computing likelihoods and by expressing the GP by its spectral decomposition and simulating by a fast fourier transform (FFT).

### 3.2.1 Projection

Explain how projection works.

### 3.2.2 Spectral representation

What is spectral density function? Define/explain. In the case of a GP with 0 mean and squared exponential covariance function (write equation?) the spectral density is

given by

$$S(s) = \sigma^2 (2\pi\ell^2)^{1/2} \exp(-2\pi^2\ell^2 s^2)$$

However, not all kernels have a closed form for the spectral density, as such we can numerically calculate  $S$  via a FFT. Explain more.

Since A zero-mean GP is a real-values 1D-IV stationary stochastic process we can use the following theorem to express the gaussian process as an infinite series. *To every real-valued 1D-IV stationary stochastic process  $f_0(t)$  with mean value equal to zero and two-sided power spectral density function  $S_{f_0, f_0}(\omega)$ , two mutually orthogonal real processes  $u(\omega)$  and  $v(\omega)$  with orthogonal increments  $du(\omega)$  and  $dv(\omega)$  can be assigned such that*

$$f_0(t) = \int_0^\infty [\cos(\omega t) du(\omega) + \sin(\omega t) dv(\omega)]$$

By the infinite series representation we can simulate the GP by the following series as  $N \rightarrow \infty$

$$f(t) = \sqrt{2} \sum_{n=0}^{N-1} A_n \cos(\omega_n t + \Phi_n)$$

where

$$A_n = (2S(\omega_n)\Delta\omega)^{1/2}, \quad n = 0, 1, 2, \dots, N-1$$

$$\omega_n = n\Delta\omega, \quad n = 0, 1, 2, \dots, N-1$$

$$\Delta\omega = \omega_c/N$$

and

$$A_0 = 0 \quad \text{or} \quad S(\omega_0 = 0) = 0$$

In equation -  $\omega_c$  represents an upper cutoff frequency for which the spectral density is assumed to be zero for any larger frequency. To estimate  $\omega_c$  we use the following

---

**Algorithm 4:** Simulating Gaussian Process via spectral decomposition.

---

**Input:**  $N, \delta t, k(t_1, t_2)$

**Output:** Draw from GP

Set  $t = 0, y_{\text{cur}} = 0, \mathbf{y} = \emptyset$  and  $h = T/K$ ;

**while**  $t < T$  **do**

Draw  $a \sim U(0, 1)$  and set  $C = 0$ ;

**while**  $C < a$  **do**

$t = t + h$ ;

**if**  $t > T$  **then**

STOP;

Numerically calculate  $C = \int_{y_{\text{cur}}}^t p(y_{\text{cur}}, u | x(u), \theta) du$ ;

Add  $t$  to the set of spike times  $\mathbf{y}$ , and set  $y_{\text{cur}} = t$ ;

**Return** spike times  $\mathbf{y}$ ;

---

criteria

$$\int_0^{\omega_c} S(\omega) d\omega = (1 - \epsilon) \int_0^{\infty} S(\omega) d\omega$$

To further improve the computation time we can rewrite this infinite series to a format which allows for the use of fast Fourier transform. Rewriting we get

$$f(p\Delta t) = \Re \left( \sum_{n=0}^{M-1} B_n \exp [i(n\Delta\omega)(p\Delta t)] \right)$$

and  $B_n$

The limitations of sampling from the GP in this method is that the step size, and number of steps is no longer independent. However, by choosing the value of  $M$  in a smart way we can still maintain the correct step size. Via

### 3.3 Estimating Length Scale

## CHAPTER 4

---

### Refractory Period

---

#### 4.0.1 How it affects the model

We will consider having a Gamma ISI throughout, although the same concept has been used for Poisson, inverse Gaussian and Weibull. Previously, without the inclusion of  $T_{\min}$  we have the following ISI distribution.

$$p(y_i, y_{i-1}|x) = \frac{\gamma x(y_i)}{\Gamma(\gamma)} [\gamma X(y_{i-1}, y_i)]^{\gamma-1} e^{-\gamma X(y_{i-1}, y_i)}, \quad (4.1)$$

where

$$X(s, t) = \int_s^t x(u) du, \quad \text{for any } s, t \in [0, T], \quad (4.2)$$

where  $\gamma > 0$  is the ISI parameter and  $\Gamma$  is the Gamma function.

Now with the inclusion of  $T_{\min}$  the ISI distribution changes to

$$p(y_i, y_{i-1}|x) = \begin{cases} 0 & \text{if } y_i - y_{i-1} < T_{\min} \\ \frac{\gamma x(y_i)}{\Gamma(\gamma)} [\gamma X(y_{i-1}, y_i)]^{\gamma-1} e^{-\gamma X(y_{i-1}, y_i)} & \text{o/w} \end{cases} \quad (4.3)$$

However, we can rewrite this equation and alter the definition of  $X$  to account for

the refractory period. Thus,

$$p(y_i, y_{i-1}|x) = \frac{\gamma x(y_i)}{\Gamma(\gamma)} [\gamma \tilde{X}(y_{i-1}, y_i)]^{\gamma-1} e^{-\gamma \tilde{X}(y_{i-1}, y_i)}, \quad (4.4)$$

where

$$\tilde{X}(s, t) = \int_{s+T_{\min}}^t x(u) du, \quad \text{for any } s, t \in [0, T], \quad (4.5)$$

Next we want to calculate the likelihood for an entire spike sequence. In general we have:

$$p(\mathbf{y}|x) = p_1(y_1|x) p_T(T, y_N|x) \prod_{i=2}^N p(y_i, y_{i-1}|x). \quad (4.6)$$

Above we have dealt with the ISI distribution, however we need to decide how  $T_{\min}$  affects  $p_1$  and  $p_T$ . We assume that  $T_{\min}$  has no effect on  $p_1$ , since we do not know the time of the last spike before the initial spike. Hence, the refractory period may already of occurred. Thus,

$$p_1(y_1|x) = x(y_1) e^{-X(0, y_1)} \quad (4.7)$$

$T_{\min}$  however does effect  $p_T$ . If the time difference between the last spike and end time is less then the refractory period then  $p_T$  can be disregarded as it has no effect on the likelihood. If the difference is longer then the refractory period then we have

$$p_T(T, y_N|x) = e^{-\tilde{X}(y_N, T)}, \quad (4.8)$$

Thus when calculating the likelihood for  $x$  we do not need to change the formula, we only need to change our calculation of  $X$  by including  $T_{\min}$  to the bottom of the integrand. However we will need to distinguish the case whether to include a term for  $p_T$  or not.

#### 4.0.2 Estimating $T_{\min}$

We estimate  $T_{\min}$  using MCMC, where if we are estimating other parameters ( $x(t)$ /ISI parameters) we use Gibbs sampling and sample from  $T_{\min}$  separately.

$T_{\min}$  only affects the likelihood via  $X(y_{i-1}, y_i)$  hence we get the marginal for each ISI distribution reduces to :

Distribution	log marginal
Poisson	$-X(y_{i-1}, y_i)$
Gamma	$(\gamma - 1) \log X(y_{i-1}, y_i) - \gamma X(y_{i-1}, y_i)$
Inverse Gaussian	1
Log Normal	1
Weibull	



## CHAPTER 5

---

### Application to Real Data

---

words

#### 5.1 Section

words

And a formula too (a complicated one... Ohm's law)

$$V = \rho \frac{l}{a} I. \tag{5.1}$$

And also a more complicated one to show the maths font:

$$Q = \mathcal{L} + \sum_{i=1}^N \prod_{j=1}^N a_{ij}. \tag{5.2}$$

## CHAPTER 6

---

### Conclusion

---

#### 6.0.1 Web application for simulating spikes

I have created an application that can be used to generate surrogate spike sequences. The app generates spike sequences using the methods described in section -. The application can be found at <https://shiny.maths.nottingham.ac.uk/pmxjp8/SimSpikes/>. Once opened you will see a panel explaining how to use the app and a sidebar with a number of inputs. The inputs are as follows:

- ISI distribution,
- ISI parameter value,
- end time,
- refractory period,
- intensity function,
- how many sequences.

Once the desired inputs have been entered you press the get spikes button. This will generate the specified number of spike sequences. The main panel will be updated with a plot of the intensity function and underneath the plot there is the raster plot of the generated spike sequences (with a maximum of 20 shown). Furthermore, underneath the plot the first generated sequence is written. Click the toggle button to swap between instructions and results window if needed. Pressing the download button will create a folder on your local computer, once extracted the folder will contain two files: a text document containing the inputs and a .csv file containing each spike sequence as a new column.

#### 6.0.1.1 Example of using the app

I will give two examples of using the application, one to generate spikes from an oscillating intensity and one from a step change. In the first example I set the parameters as follows: ISI distribution to Gamma, ISI parameter to 32, end time to 40, refractory period to 0, intensity function to  $\cos(t/2.3) + 1.2$  and the number of sequences to 1. Pressing get spikes gives the pane shown in 6.1A. Where the intensity is plotted with the spike sequence shown below by the black points. Furthermore, the sequence is written explicitly below the plot.

For the step change I decided to change the parameters to as follows: ISI distribution to Weibull, ISI parameter to 7, end time to 40, refractory period to 0, intensity function to  $c(t[t < 20] * 0 + 1.6, t[t > 20] * 0 + 0.5)$  and the number of sequences to 25. The input for the intensity may look confusing, however this is the format required for R to understand the step function. The intensity function is generated by applying the input provided to the vector  $t$ , which is the sequence starting from 0 to the inputted end time in 8000 equal steps. Thus, in the case of the step function we split the  $t$  vector into two for values less than 20 by  $t[t < 20]$  and similarly for greater than 20. Then to get the constant intensity we multiply this vector by 0 and add the value we want the intensity to be. In our case we set the intensity to 1.6 between 0 and 20 and 0.5 between 20 and 40. Then to join the two vectors back together we use  $c()$  which is used by R to concatenate the vectors together. This

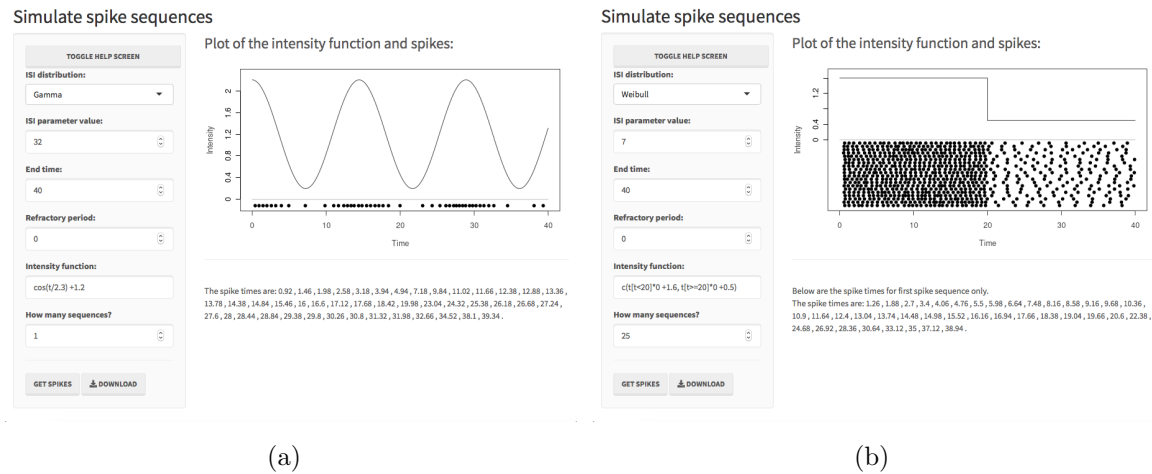


Figure 6.1: Examples of generated spike sequences using SimSpikes.

method can easily be used in combination with functions to create more complicated intensity functions as inputs. For example  $\cos(t[t < 20]) + 1.4$  sets the intensity to be  $\cos(t) + 1.4$  between the times 0 and 20.

Clicking the get spikes button is shown in 6.1B. We see that since we increased the number of sequences to generate there are more spike sequences shown as rasters underneath the plot of the intensity function, however only one sequence is still printed below the plot. Clicking the download button creates a new folder on your local computer called SimSpikes. Opening the folder you are greeted by two files one called details.txt and another called spikes.csv. Figure 6.2(A) shows the contents for details.txt, which included all the inputs used to generate the spike sequences. This is useful in case when using the simulated spikes at a later date you have forgotten the inputs used to them. Figure 6.2(A) is a screenshot of the spikes.csv in Excel, you can see that each spike sequence is stored as a column in the table. Each column contains the generated spike sequence adjoined with the simulation end time. Since a generated sequence will not have the same number of spikes, columns contain NA values after the sequence.

```

The inputs for the simulated spike sequences were
End time = 40
Intensity function = c(t[t<20]*0 +1.6, t[t>=20]*0 +0.5)
ISI distribution is Weibull with parameter 7
Refractory period was set to 0
Number of sequences is 25.

```

(a)

(b)

Figure 6.2: Images of the files downloaded from the step change intensity function SimSpikes.

## 6.0.2 Web application for viewing/analysing raw data

I have created an application that can be used to view the raw time series data and then threshold to retrieve spike sequences. The application can be found at <https://shiny.maths.nottingham.ac.uk/pmxjp8/ViewData/>. The application was created with the aim at easily viewing the raw time series data with a user-friendly method of thresholding the data to calculate spike times. Once the application is opened you are greeted with a panel explaining how the application works and a sidebar where inputs can be altered. To begin you need to input the raw data, which needs to be formatted in a ‘.csv’ file such that the first column corresponds to the time indexing and each following column corresponds to the calcium concentration at the times for a single cell. If the uploaded file has missing data the upload will fail and the user is prompted with the reason.

Once the upload has been successful a plot of calcium concentration against time will appear for the first cell, with an option on the sidebar to toggle between the different cells. By changing inputs on the sidebar the user can threshold the data which can be seen on the time series plot, further to this a new plot showing the ISIs will appear and the spike times written below. Once some of the data has been analysed the user can download the spikes and corresponding threshold details by clicking the download button which will save the results on the users home computer,

which can be reloaded into the app at a later date.

### 6.0.2.1 Example of using the app

A guide to using the app appears on loading the webpage. A run through of an example is provided here where we upload some data and retrieve spike times.

Firstly, we click on the 'INPUT DATA' button, and a pop up window appears. We then click browse and choose the file 'data.csv' from our local PC. Once the file is successfully uploaded we click 'ok' to finish the upload. If we have already uploaded, analysed and downloaded from the application we can also upload 'spike details' using the second upload box, which will then present the details from the previously analysed data.

Returning to just uploading data once we have clicked 'ok', we are shown the plot of calcium concentration against time for cell 1 - as seen in figure ???. On the sidebar under cell number we can toggle which cell's information we look at, the current cell number can be seen in this section of the sidebar and is also printed above the plot along with the number of cells that the data contains. The x and y axes for the concentration plot can be changed by unchecking the default axes buttons on the sidebar where you are then prompted to enter the range of values you want to view. This function is useful to zoom in on the plot for a better view.

Now we will consider thresholding the data for cell 1. In the sidebar there is section labelled 'Thresholding', under this section we can tell the application the heights at which we wish to threshold the data, as well as removing transient sections or removing linear trends in the data. We may want to remove transient regions since after applying a stimulus there is often a time window where the calcium concentration is non-stationary and we only want to consider spikes that are in a stationary regime. Furthermore, the user may want to remove linear trends as they could be caused by experimental effects such as de-sensitisation of the tracer placed in the cell and doesn't reflect the true nature of the calcium spiking rate.

Looking at cell one it seems reasonable to threshold the spikes at the concentration value 0.7. We enter the value into the 'Threshold at?' in the sidebar and the

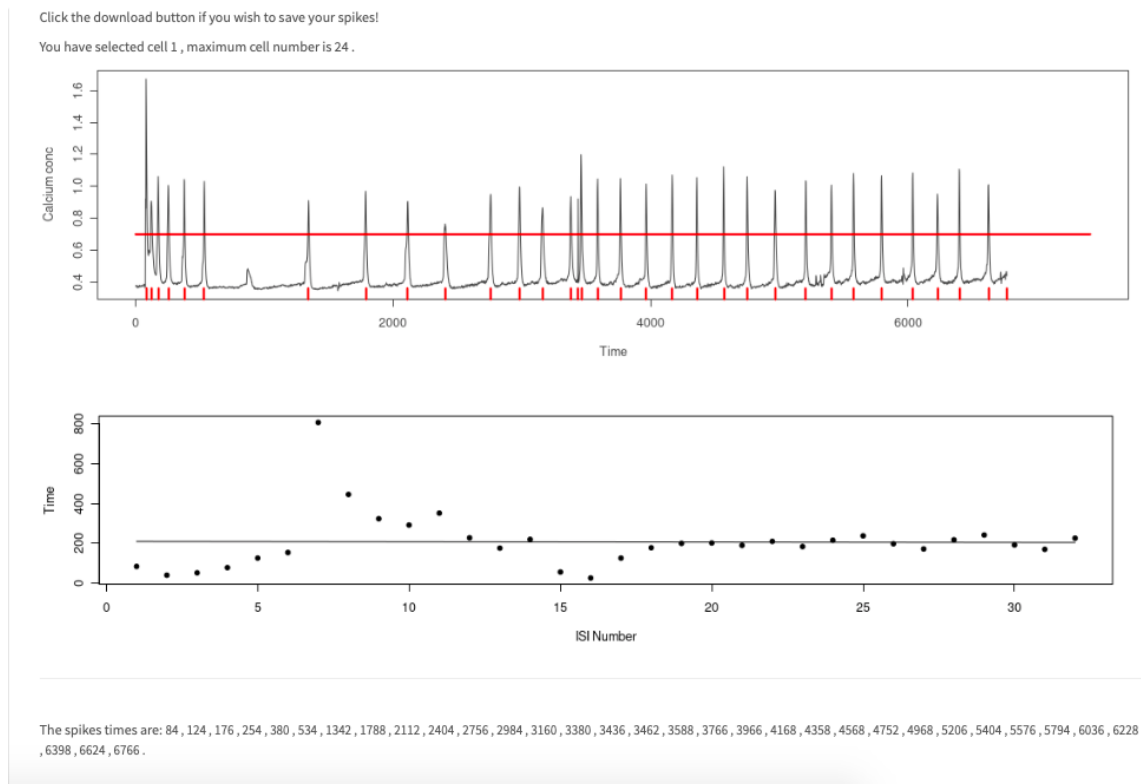


Figure 6.3: Print-screen of main pain of ViewData app where the threshold input has been updated to be 0.7.

application updates and shows figure 6.3. We see that the threshold line shows up on the concentration plot as a red line, and the red ticks on the x-axis shows the spike times. A new plot is produced under the concentration plot, which shows the ISI times for the thresholded data, which can aid in seeing any patterns in the spikes that are harder to visualise from the concentration plot. For example, if the ISI times slowly decreased this would be easier to see in ISIs rather than the raw spike times. Furthermore, in the ISI plot we fit the linear line of best fit to show the linear trends in the ISIs. Then underneath the ISI plot, the spike times are explicitly written, which can makes the spikes easily accessible via a copy and paste rather than having to download the data for all the cells.

Consider again the plot in figure ?, if we look between times 800 and 1000 we see a small spike in the calcium concentration that we may want to include as a spike time,

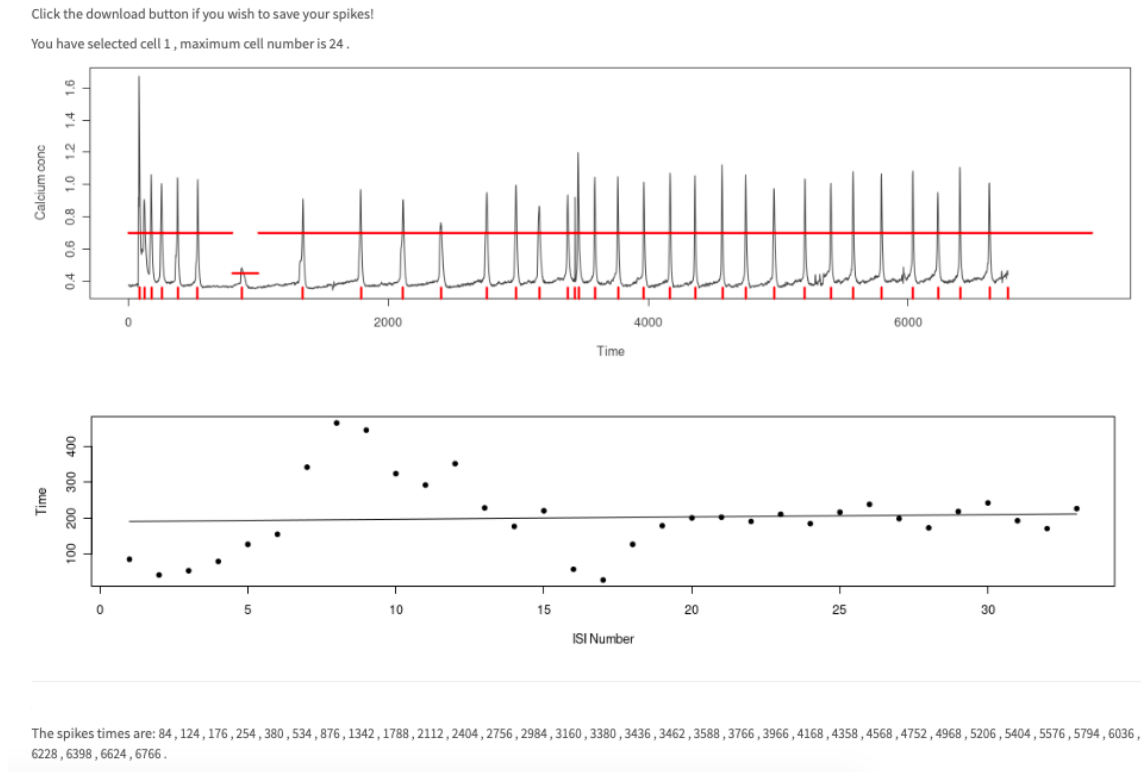


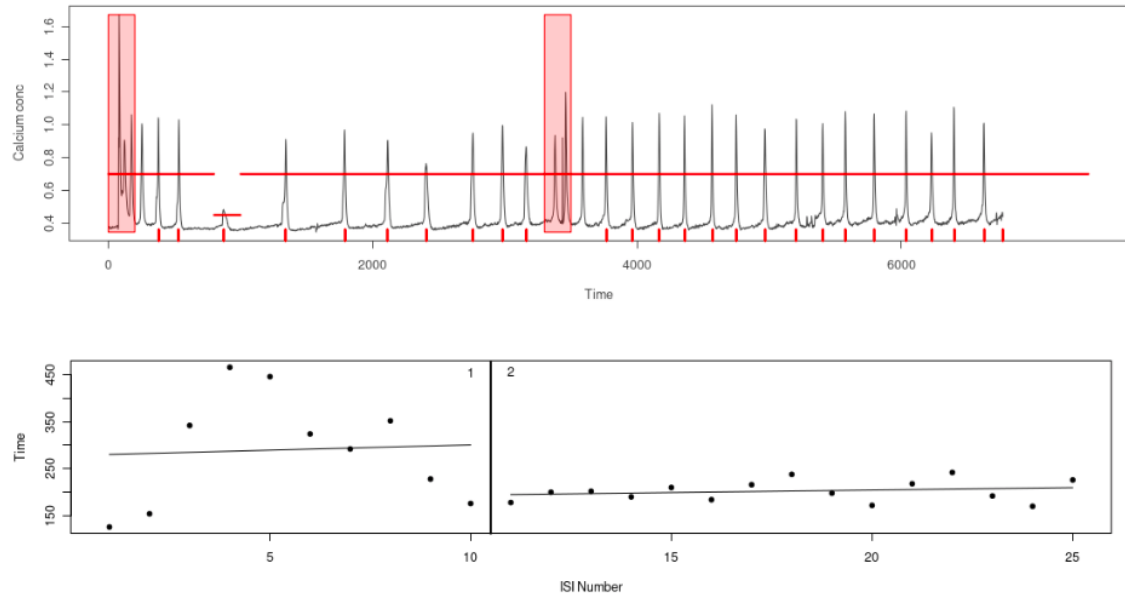
Figure 6.4: Print screen of App where threshold has been updated to ‘0.7,0.45,0.7,800,1000’.

however if we lowered the threshold time we may lose some of the other spikes (i.e at the beginning when the concentration is higher then the little bump). So, to include this value we can enter multiple threshold values by entering ‘0.7,0.45,0.7,800,1000’ into the threshold at? box. This tells the app that between times 0 and 800 you want to threshold at 0.7, between 800 and 1000 threshold at 0.45 and threshold at 0.7 for time greater than 1000. In figure 6.4 we see the red line has updated for our new threshold value and a new red tick is produced the new spike we have included.

Continuing with cell one the data comes from a step change experiment and we may want to remove the transient sections at the start and middle of the experiment. By checking the transient period box in the sidebar we are prompted to enter the transient regions at either the start, middle or end of the experiment. Looking at cell one we choose to transient periods to be between 0 and 200 at the start and between 3300 and 3500 in the middle. In figure 6.5 we see the affect of changing these values.



Click the download button if you wish to save your spikes!  
 You have selected cell 1 , maximum cell number is 24 .



The spikes times are: 126 , 280 , 622 , 1088 , 1534 , 1858 , 2150 , 2502 , 2730 , 2906 , 3084 , 3284 , 3486 , 3676 , 3886 , 4070 , 4286 , 4524 , 4722 , 4894 , 5112 , 5354 , 5546 , 5716 , 5942 , 6084 .

Figure 6.5: Main Panel where we remove two transient regions. The first region between 0s and 200s. The second interval is between 3300s and 3500s.

In the concentration plot red boxed now appear over the inputted transient periods. Notice that the red ticks now occur for spikes outside of the red boxes except for the first spike outside of these regions this is because these first spikes are taken as the start time of the sections. For example in the first section we say the region 0s to 200s is transient and the next spike outside this region previously occurred at 254s. Since removing transient regions is *donebecause* we want to consider stationary spikes, the region between the cutoff and the next spike is dependent on the cutoff, however by removing the first spike and setting the first spike time as time zero all the following ISI are true ISI. In the ISI plot the plot is now split into two windows one for the spikes before the middle transient period and one for afterwards, with each having its own linear trend line.

Considering the relationships in the ISI plot both appear to have a minimal positive trend, we can remove these linear trends by checking the 'Remove linear

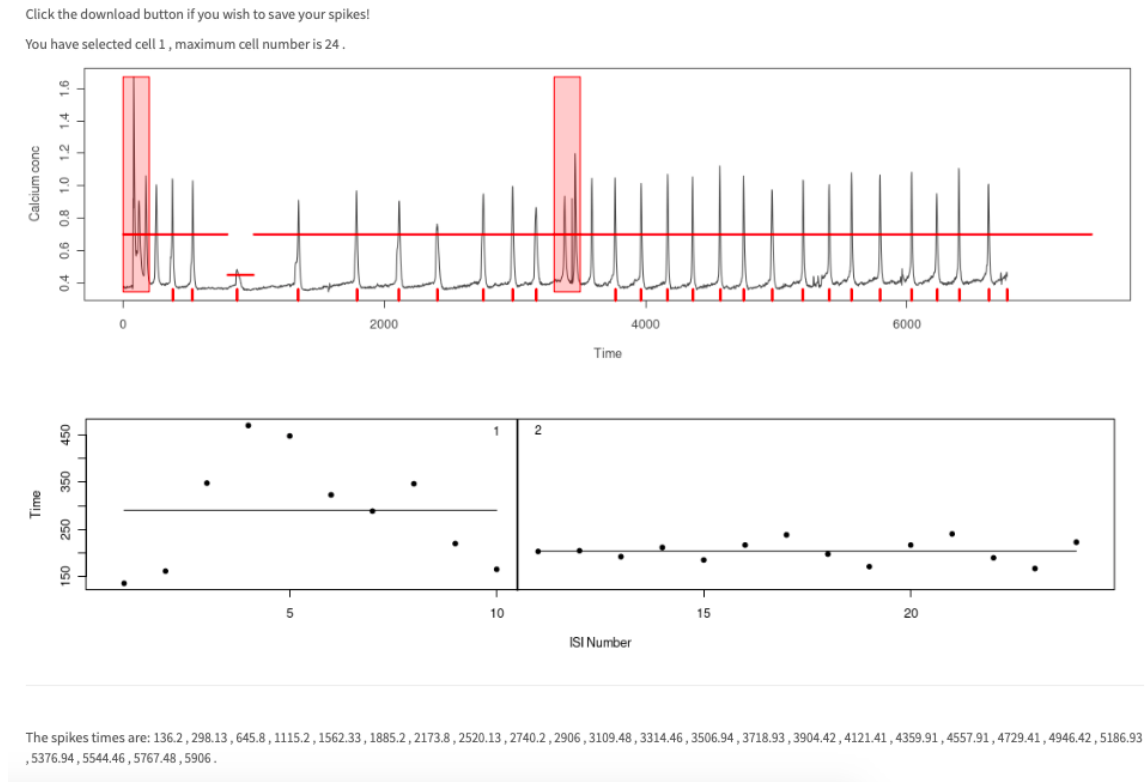
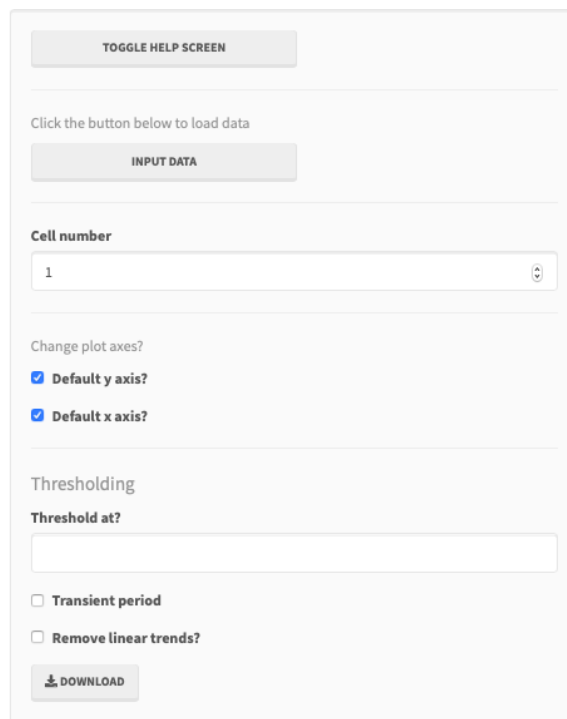


Figure 6.6: Main panel where linear trends are removed.

trends?’ box. In figure 6.6 we have removed the linear trends we see this has no affect on the concentration plot but now in the ISI plot both linear fits are flat showing there are no linear trends in the adjusted data.

We have now decided that we have successfully got the spikes for cell one and we wish to download the spikes for later use. Clicking on the ‘Download’ button will either save to your download folder automatically or prompt the user for a download. In the downloaded folder there will be three files. The first file entitled ‘Details.txt’ stores the data file from which we analysed the raw data in this case ‘Data.csv’. The second file is called ‘spikes.csv’ which stores the resultant spike sequences for each cell as a column in the file, see figure ???. Cells which have not been thresholded will have a column of NA’s in the file. The third file ‘store.csv’, contains the threshold details for each cell in the analysis. This is the file that can be uploaded at the start of the program along with the raw data to show the threshold details used, or can be used as a basis to update the threshold if mistakes were initially made.



TOGGLE HELP SCREEN

Click the button below to load data

INPUT DATA

Cell number

1

Change plot axes?

☒ Default y axis?

☒ Default x axis?

Thresholding

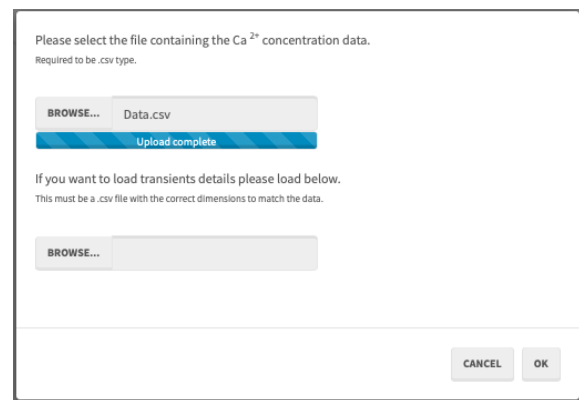
Threshold at?

☐ Transient period

☐ Remove linear trends?

DOWNLOAD

(a)



Please select the file containing the  $\text{Ca}^{2+}$  concentration data.  
Required to be .csv type.

BROWSE... Data.csv

Upload complete

If you want to load transients details please load below.  
This must be a .csv file with the correct dimensions to match the data.

BROWSE...

CANCEL OK

(b)

Figure 6.7: Images of the files downloaded from the step change intensity function SimSpikes.

---

## Bibliography

---

- A, P. (2010). The role of the nugget term in the gaussian process method. *mODa 9 – Advances in Model-Oriented Design and Analysis*, pages 149–156.
- A, S., H, K., U, W., M, W., H, S., Sc, T., and et al (2011). How does intracellular  $\text{Ca}^{2+}$  oscillate: by chance or by the clock? *Biophysical Journal*, 94(6):2404–2411.
- A, T., W, C., N, R., TC, B., and R, T. (Oct 17). A Bayesian approach to modelling heterogeneous calcium responses in cell populations. *PLoS Computational Biology*, 13(10):e1005794.
- C, R. and C, W. (2006). *Gaussian Processes in Machine Learning*. MIT Press.
- Daley, D. J. and Vere-Jones, D. (2008). *An Introduction to the Theory of Point Processes. Volume II: General Theory and Structure*. Springer New York, New York, NY.
- Denninger, P., Bleckmann, A., Lausser, A., Vogler, F., Ott, T., Ehrhardt, D. W., Frommer, W. B., Sprunck, S., Dresselhaus, T., and Grossmann, G. (2014). Male–female communication triggers calcium signatures during fertilization in arabidopsis. *Nature Communications*, 5(1).

- DJ, D. and D, V.-J. (2003). *An Introduction to the Theory of Point Processes. Vol. I.* Probability and its Applications (New York). Springer-Verlag, New York, second edition. Elementary theory and methods.
- E, A. and D, G. (1994). Nonparametric Bayesian Inference from Right Censored Data, using the Gibbs Sampler. *Statistica Sinica*, 4(2):505–524.
- EN, B., R, B., V, V., RE, K., and LM, F. (2002). The time rescaling theorem and its application to neural spike train data analysis. *Neural Computation*, 14(2):325–346.
- G, C. and E, G. (1992). Explaining the Gibbs Sampler. *The American Statistician*, 46(3):167–174.
- JP, C., KV, S., and M, S. (2008). Fast gaussian process methods for point process intensity estimation. In *Proceedings of the 25th International Conference on Machine Learning*, pages 192–199. Stanford University.
- K, T., IF, S., SC, T., CW, T., I, P., and M, F. (2011). Timescales of IP<sub>3</sub>-evoked Ca<sup>2+</sup> spikes emerge from Ca<sup>2+</sup> puffs only at the cellular level. *Biophysical Journal*, 101(11):2638–2644.
- K, T., SC, T., G, M., VL, P., A, M., AP, T., A, S., CW, T., and M, F. (2014). Reliable encoding of stimulus intensities within random sequences of intracellular Ca<sup>2+</sup> spikes. *Science Signaling*, 7(331):ra59–ra59.
- L, D. (2013). *Non-Uniform Random Variate Generation*. Springer-Verlag New York.
- MB, W. and R, G. (1968). Probability plotting methods for the analysis of data. *Biometrika*, 55(1):1–17.
- PJ, G. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732.
- R, B., MC, Q., LM, F., MA, W., and EN, B. (2001). Construction and analysis of non-Poisson stimulus-response models of neural activity. *The Journal of Neuroscience Methods*, 105(1):25–37.

- RM, N. (1999). Regression and Classification using Gaussian Process Priors. In JM, B., JO, B., AP, D., and AFM, S., editors, *Bayesian Statistics 6*, pages 475–501. Oxford University Press.
- S, C. and E, G. (1995). Understanding the Metropolis-Hastings algorithm. *The American Statistician*, 49(4):327–335.
- Santella, L., Lim, D., and Moccia, F. (2004). Calcium and fertilization: the beginning of life. *Trends in Biochemical Sciences*, 29(8):400–408.
- Van Nhieu, G. T., Clair, C., Bruzzone, R., Mesnil, M., Sansonetti, P., and Combettes, L. (2003). Connexin-dependent inter-cellular communication increases invasion and dissemination of shigella in epithelial cells. *Nature Cell Biology*, 5(8):720–726.
- W, T., UT, E., MR, F., JP, D., and EN, B. (2005). A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of Neurophysiology*, 93(2):1074–1089.
- Whitaker, M. (2006). Calcium at fertilization and in early development. *Physiological Reviews*, 86(1):25–88.