

UNIVERSITY OF NOTTINGHAM



SCHOOL OF MATHEMATICAL SCIENCES

**Statistical modelling of Ca^{2+}
oscillations in the presence of
single cell heterogeneity**

Jake Powell

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

29TH APRIL 2022

Abstract

Intracellular calcium oscillations are 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.

In this thesis, we develop a probabilistic approach that treats calcium oscillations as a stochastic point process. By employing an intensity function — a one dimension function over time which corresponds to the mean calcium spiking rate — we capture intrinsic cellular heterogeneity as well as inhomogeneous extracellular conditions, such as time-dependent stimulation.

We adopt a Bayesian approach to infer the model parameters from calcium oscillations. Under this approach we need to be able to infer the intensity function. One method is to use a parametric model for the intensity function. For example we could assume the intensity function has the linear form $x(t) = at + b$. Then the intensity function is reduced to only needing to infer the two parameters a and b . However, parametric models suffer from strict assumptions, in this case, for the shape of the intensity function. Therefore, to lessen such assumptions, we utilise a non-parametric approach. This requires a prior distribution over the space of functions. We use two such priors, namely Gaussian processes and piecewise constant functions.

We use Markov chain Monte Carlo (MCMC) techniques to sample from the posterior distribution to obtain estimates for our model parameters. Although advantageous — due to sampling from the true posterior distribution — MCMC algorithms

can experience issues relating to their computational cost and imprecise samplers. We discuss the issues arising for our particular model and data and develop methods to improve the functionality of the MCMC algorithms in this case. For example we discuss the difficulty of inferring the length scale of the Gaussian process when fitted from calcium oscillations.

An important mechanism of calcium oscillations is the refractory period, the minimum amount of time before the next calcium oscillation. Thus, it may be beneficial to explicitly include the refractory period as part of the model. We investigate the advantages and disadvantages of including the refractory period.

We fit the model to HEK293 cells and astrocytes challenged under a variety of stimulation protocols. We find that our model can accurately generate surrogate spike sequences with similar properties to those the model is fitted from. Therefore, the model can be used to cheaply create spike sequences that are synonymous to those found experimentally. Moreover, our model captures the similarity between calcium spike sequences obtained from step-change stimulus protocols and constant stimulus protocols. Combining intensity functions inferred from constant stimulus experiments closely follow the intensity function from a step change experiment. This implies it may be possible to build surrogate spike sequences for complex time-dependent stimulation protocols by combining results from simpler experiments.

Of particular interest are patterns found in the intensity function which describes the heterogeneity in the calcium oscillations over time. Common patterns could help to understand the different time scales of the calcium response. Standard approaches often fail in grouping intensity functions with similar shape. Therefore, we develop an approach to cluster intensity functions based on their shape alone by utilising the Haar basis.

In summary, we have developed novel statistical approaches based on the concept of stochastic point processes and non-standard MCMC techniques. We have successfully

applied these new methodologies to gain a deeper understanding into the stochastic nature of intracellular calcium oscillations, in particular how different cell types respond to a variety of stimulation protocols. In turn, this brings us one step closer to unravel the complex dynamics of this pivotal intracellular messenger which controls life from its very beginning to its end.

Publications

J. Powell, M. Falcke, A. Skupin, T. C. Bellamy, T. Kypraios, and R. Thul, “A statistical view on calcium oscillations,” in *Advances in Experimental Medicine and Biology*, pp. 799-826, Springer International Publishing, 2019.

Acknowledgements

First and foremost, I would like to thank my supervisors, Rüdiger Thul and Theo Kypraios, without whom none of this would have been possible. In particular, I would like to thank both for introducing me to the topic of this thesis, and the advice, enthusiasm and patience offered by both throughout.

Secondly, I am greatly indebted to my family for their unwaveringly encouragement and support, without which I would not be where I am today.

Thirdly, I'd like to thank the friends I have met during my time at the University of Nottingham. In particular, the maths department football group which gave me the opportunity to unwind and relax.

Finally, a special thank you to my girlfriend Lucy for all her love and support throughout .

This work was supported by the UK Engineering and Physical Sciences Research Council (EPSRC) grant EP/N50970X/1. I am grateful to Thomas Bellamy, Alexander Skupin, Colin Taylor and Martin Falcke for sharing Ca^{2+} concentration data to me.

Contents

1	Introduction	1
1.1	Ca ²⁺ signalling background	6
1.1.1	Calcium Signalling	7
1.1.2	Modelling Ca ²⁺ oscillations	12
1.1.3	Ca ²⁺ imaging	16
1.2	Point Process Background	18
1.2.1	Poisson Process	20
1.2.2	Point processes and time-dependent renewal theory in neuro- science	21
1.3	Structure of the Thesis	22
2	Model and Bayesian Inference	24
2.1	The model	24
2.2	Simulating Spikes	29
2.3	Bayesian Inference	31
2.4	Bayesian Computation	35
2.4.1	Simple Model	36
2.4.2	Piecewise Constant Prior	40

2.4.3	Gaussian Process Prior	53
2.4.4	Updating ISI Parameter	61
2.5	Example with simulated data	64
2.6	Summary	67
3	Non-trivial Model Properties	69
3.1	ISI Distribution	70
3.2	Mixing issues with the GP and their resolutions	83
3.2.1	Method	88
3.2.2	Choice of f	92
3.2.3	Proposal mechanism for GP	97
3.3	Estimating the Length Scale of a GP prior	99
3.3.1	Estimating l with multiple spike sequences	102
3.3.2	Estimating l with limited information	107
3.3.3	Summary	112
3.4	Speed of MCMC algorithms	112
3.4.1	The Mean Projection Approximation	116
3.4.2	Simulating using spectral representation	120
3.4.3	Summary	127
4	Refractory Period	128
4.1	Including the refractory period in the ISI model	129
4.1.1	Affect on the intensity function	130
4.2	Estimating T_{\min}	132
4.3	Advantages and disadvantages of including T_{\min}	136
5	Application to Real Data	142
5.1	Model Assessment	143
5.1.1	Kolmogorov-Smirnov and Quantile-Quantile plots	143
5.1.2	Issue with Q-Q and K-S plots	147
5.2	Constant Stimulus	150

5.2.1	Astrocytes	150
5.2.2	HEK293 cells	163
5.2.3	Summary	170
5.3	Time-dependent Stimulus	171
5.3.1	Step-change stimulus	171
5.3.2	Wave stimulus	177
5.3.3	Pulse stimulus	181
5.4	Summary	185
6	Clustering Intensity Functions	191
6.1	The Haar basis	199
6.2	Extracting features	203
6.2.1	Calculating the magnitude of the features	211
6.3	Clustering	213
6.4	Application to HEK293 cells	218
6.4.1	Effects of altering Ca^{2+} spike times on features	220
6.4.2	Constant stimulus experiments	224
6.4.3	Step change experiments	230
6.5	Summary	233
7	Web applications of our methods	236
7.1	Web application for viewing/analysing raw data	237
7.1.1	Example of using the app	237
7.2	Web application for simulating spikes	243
7.2.1	Example of using the app	244
7.3	Web application for finding features in intensity functions	247
7.3.1	Example of using the app	247
8	Conclusion	252
8.1	Overview	252
8.2	Future work	259

Appendices	262
A Non-identifiable distributions	263
B Complementary figures for application to real data	265
B.1 Astrocytes challenged glutamate	265
B.2 HEK293 cells challenged with constant carbachol	270
B.3 HEK293 cells challenged with a step change in carbachol	273
B.4 HEK293 cells challenged with three waves of carbachol	280

List of Figures

1.1	Ca ²⁺ oscillations of HEK293 cells for a variety of stimulation protocols as shown by the bars. (A) Two cells challenged with 10μM carbachol. (B) A cell initially challenged with 50μM carbachol and then with 100μM carbachol at 3380s. (C) A cell challenged with three waves of carbachol. This is shown in the coloured bar where the lighter colour corresponds to a larger concentration of carbachol. The red ticks on the time axis correspond to the Ca ²⁺ spike times.	2
1.2	(A) Diagram is adapted from [1]. (B) Graph taken from [2]. (C) Image taken from [3]. (A) Illustration of Ca ²⁺ oscillations generated by the IP ₃ Ca ²⁺ pathway. (B) Open probability of IP ₃ Rs as a function of Ca ²⁺ concentration shown for varying IP ₃ concentration. Filled dots show experimentally recorded values and simulation results by the solid lines. (C) 3D IP ₃ R structure drawn from two perspectives: the cytosol and along the membrane plane.	8

-
- 1.3 Figure taken from [4]. Illustration of Ca^{2+} events, where IP_3R channels that are active and release Ca^{2+} are shown in green. (A) represents a blip, where Ca^{2+} is released from a single receptor. (B) represents a puff, where all receptors from a single cluster are activated. (C) represents a wave, where neighbouring clusters are activated by CICR. 11
- 1.4 Figure taken from [5]. Illustration of Ca^{2+} imaging. Images of chondrocytes in a cartilage explant that were dyed with the Ca^{2+} indicator Fluo-8 AM. Bright green regions correspond to a larger Ca^{2+} concentration. Responsive cells — cells whose Ca^{2+} concentration changes — are marked with a white arrow. 17
- 1.5 Visualisation of the different ways to define a point process. The red lines correspond to event times. 19
- 2.1 Fura-2 fluorescence intensity trace of a HEK293 cell stimulated initially by $20\mu\text{M}$ carbachol and then with $50\mu\text{M}$ carbachol at 3444s. The spike times (red ticks) are obtained by thresholding (red line) the change in fluorescence at 0.8. The data was kindly provided by Dr. Falcke, for experiment details see [6]. 25
- 2.2 (A) Raster plot of simulated spike sequences in $[0\text{s}, 20\text{s}]$ from an inhomogeneous Gamma ISI distribution with ISI parameter $\gamma = 10$ and intensity function $x(t) = 2 \cos \frac{t}{2} + \cos \frac{t}{4} + 2.8$. (B) Comparison of the intensity function (red) and the histogram (blue) obtained from 1000 spike sequences simulated from the ISI distribution used in (A). . . . 31
- 2.3 Posterior Gamma ISI distribution of the simple model fitted to a single simulated Ca^{2+} spike sequence. Each grey point represents one sample from the posterior distribution — consisting of intensity function x and ISI parameter θ — 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 θ respectively. 39

-
- 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. 42
- 2.5 Illustration of the possible moves at each iteration of the RJMCMC. The black line corresponds to the original step function. The blue, red, cyan and magenta lines correspond to candidate intensity functions when: a change point is removed, a new change point is added, a height is changed and a change point moves, respectively. 45
- 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. . . 52
- 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 σ_f^2 increases the range of values taken (variance). Finally, (D) shows noise is controlled by σ_v^2 54
- 2.8 Samples from the intensity function prior distribution, where its logarithm has a GP prior. To illustrate the affect of the length scale on the prior distribution the black, red and blue lines correspond to length scales 10, 5 and 1, respectively. 58

-
- 2.9 Illustration of fitting an inhomogeneous Gamma ISI distribution to spike sequences simulated from a model with a constant intensity function. (A) Raster plot of 40 simulated spike sequences. (B) Mean and 95% credible region of the posterior intensity function distribution inferred from a single spike sequence. The black line represents the intensity function used to simulate spike sequences. (C) Histogram of the ISI parameter γ for the spike sequence and PWC priors shown in (B). In (B,C) The constant, PWC and GP priors for the intensity function are represented by the colours magenta, red and blue, respectively. (D) Scatterplot of the MLEs for 200 spike sequences, where each dot corresponds to the MLE of a single spike sequence. 63
- 2.10 Illustration of fitting an inhomogeneous Gamma ISI distribution to spike sequences simulated from a model with a time-varying intensity function. (A) Raster plot of 40 of the simulated spike sequences. (B,C) Mean and 95% credible region of the posterior intensity function distribution inferred from a single spike sequence, shown as black ticks in both. The black line represents the intensity function used to simulate spike sequences. In addition red, blue and magenta lines correspond to posterior intensity computed via PWC, GP and simple priors respectively. (D) Comparison of the intensity function used to simulate spike sequence (black) and the posterior mean of 40 intensity function distributions where each is fitted to a single simulated spike sequences with a PWC prior. 66
- 3.1 Probability density functions corresponding to the MLE of: an Exponential (dark green), a Gamma (orange), inverse Gaussian (pink), log-normal (light green) or Weibull distribution (blue) fit to Ca^{2+} ISIs — shown as black ticks on the x-axis. 73

-
- 3.2 Comparing the intensity function — the red line in both plots — to histograms of 1000 simulated spike sequences. In (A) the light and dark boxes correspond to spikes generated from the model with Gamma ISI distributions with parameters $\text{Gamma}(1.8,1.8)$ and $\text{Gamma}(3.6,1.8)$ respectively. In (B) the spike sequences are simulated from a $\text{Gamma}(1.8,1.8)$ and an inverse Gaussian $(1.8,1)$ resulting in the histograms in blue and red respectively, whose overlap is purple. 79
- 3.3 Raster plots comparing spike sequences drawn from two Gamma ISI distributions with the same intensity function — shown in (A) — and differing ISI parameter. The sequences in (B) and (C) correspond to γ values of 3 and 20 respectively. 81
- 3.4 Posterior mean — black line — and 95% credible interval — red shaded area — for the intensity function, from the model with an inhomogeneous Gamma ISI distribution. The inference was calculated using 80,000 iterations with a burn-in of 50,000. The covariance parameters are $\sigma_f^2 = 1000$, $\sigma_n^2 = 1e^{-5}$ and $l = 1.50$ — l is obtained using the PWC prior, see Section 3.3 for more detail. The intensity function is split into $N = 2000$ steps and the under-relaxed parameter $\omega = 0.001$. The ISI parameter had an exponential prior with rate 0.01. The blue line shows a candidate intensity function created by flattening the posterior mean in $[0s, 1.5s]$ and $[19s, 20s]$. The black ticks represent the Ca^{2+} spike times used to fit the model. 84
- 3.5 Trace plots of the intensity function at time 0s over 300,000 iterations, for the GP prior utilising an under-relaxed method. Each plot shows the results obtained for the same spike sequence; the only difference is the random seed used. The inference parameters are the same as Figure 3.4, using the same spike data. The coloured dots in (C) correspond to the coloured intensity functions shown in Figure 3.6. 86

-
- 3.6 Sampled intensity functions, corresponding to the coloured dots in Figure 3.5. Intensity functions go from purple through green to yellow, as the intensity at 0s increases. The black ticks represent the Ca^{2+} spike times. 87
- 3.7 Illustration of how to partition \mathbf{x} into two regions A and B . The partition value is drawn uniformly from the grey region. An example draw t_M is shown by the dotted red line. The thick black line — left of t_M — is the region of \mathbf{x} we will propose new values for, whereas the thin black line — right of t_M — will remain unchanged. The black ticks on the x-axis represent spike times. 89
- 3.8 Illustration of the proposal mechanism when $f = \log \mathbf{x}_A$. (A) shows ten candidate functions — grey lines — where the current intensity is shown by the red line. (B) The accepted candidate functions from 100 iterations of the proposal mechanism where the original \mathbf{x}_A is shown as the red line. The iterations progress from black to blue. The black ticks show the spike times. The proposal parameters in (A) and (B) are $K = 10$, $l = 1.59$, $\sigma_n^2 = 1e^{-5}$ and $\sigma^2 = 1$. In (A) the width is fixed with $M = 337$, whereas in (B) the width varies in each iteration and $w = 100$ 93
- 3.9 (A,B) and (C,D) are illustrations of the proposal mechanism for $f = f_{\min}$ and $f = f_{\text{mean}}$, respectively. (A,C) shows ten candidate functions — grey lines — where the current intensity is shown by the red line, and the mean of the proposal is shown in blue. (B,D) The accepted candidate functions from 100 iterations of the proposal mechanism where the original \mathbf{x}_A is shown as the red line. The iterations progress from black to blue. The black ticks show the spike times. The proposal parameters are the same as in Figure 3.8. 95

-
- 3.10 The logarithm of proposed candidate functions from using the data-driven method at the start, where red and brown lines show candidate functions using f_{\min} and blue lines show candidate functions using f_{cur} . The logarithm of the current intensity function is shown in black, and spike times as black ticks. 96
- 3.11 (A) gives the posterior mean and 95% credible region using the new proposal mechanism. The inference was calculated using 80,000 iterations with a burn-in of 50,000 with parameters $\sigma_f^2 = 1000$, $\sigma_n^2 = 1e^{-5}$, $l = 1.50$, $N = 2000$ and $\omega = 0.001$. (B) shows ten trace plots of the intensity at time 0s using the new proposal mechanism. 97
- 3.12 The mode of the posterior intensity function distribution fitted to a single spike sequence — shown by the black ticks. The model used was an inhomogeneous Gamma ISI distribution with a GP prior, where the length scale of the GP varies from 169 to 6766 — as shown in the legend. The inference was calculated over 20,000 iterations after a burn-in of 20,000, the parameters used were: $N = 2000$, $\sigma_h^2 = 0.5$, $\sigma_f^2 = 1000$, $\sigma_n^2 = 10^{-5}$, $w = 0.01$. The prior for the ISI parameter was an exponential with rate 0.01. 100
- 3.13 Two intensity functions which give rise to the same likelihood with the spike sequence shown by the black ticks. 102
- 3.14 (A) The intensity function used to create surrogate spike sequences, drawn from a GP with mean 0 and square exponential covariance with parameters $l = 3$, $\sigma_f^2 = 1$ and $\sigma_n^2 = 10^{-5}$. (B) Histogram of posterior samples of l inferred from the intensity function given in (A). (C) 20 simulated spike sequences generated from an inhomogeneous Gamma ISI distribution with $\gamma = 7$ and intensity function given by (A). . . . 103

-
- 3.15 Visualisation of the posterior ISI distribution when using 1, 2, 5, 10 and 20 spike sequences to infer an inhomogeneous Gamma ISI distribution with a GP prior. (A-B) Trace plots of the ISI parameter and length scale, respectively. (C) The mean of the posterior intensity function distribution. The black curves in (A) and (C) coincide with the ISI parameter and intensity function used to generate the spike sequences. 105
- 3.16 Examples of the mean of the posterior intensity function where we fit an inhomogeneous Gamma ISI distribution with a GP prior to the spike sequence — shown by the black ticks — whilst varying the length scale. The spike sequence is taken from Figure 3.14. We calculate 20000 iterations after an initial burn-in of 20000. The inference parameters used were: $\omega = 0.001$, $\sigma_f^2 = 1000$, $\sigma_n^2 = 10^{-5}$ and $\sigma_h^2 = 0.5$. The length scale used for each model is shown in the legend. 108
- 3.17 Illustration of obtaining l using the PWC prior. Begin by fitting an inhomogeneous Gamma ISI distribution with a PWC prior to the spike times shown as the black ticks. The parameters for the PWC prior were: $k_{\max} = 25$, $\kappa = \kappa_0 = 1$ and $\mu = 0.5$. The posterior mean of the intensity function \mathbf{x}_{pwc} is shown by the purple line. We smoothed \mathbf{x}_{pwc} by using local polynomial regression with automatic smoothing parameter selection — shown by the blue line. We estimate l by finding the MLE of l using a subset of the smoothed \mathbf{x}_{pwc} — shown by the yellow dots. The intensity function used to generate the spike sequence is shown by the black line. 110

-
- 3.18 The posterior mean and 95% credible interval for the intensity function fitted using an inhomogeneous Gamma ISI distribution with a GP prior. We use blue to represent the fixed length scale $l = 2.74$ and purple is used when we draw l from a $\text{Gamma}(2.74 \times 8, 8)$ in each iteration. The inference parameters are identical to those used in Figure 3.16. The spikes used to fit the model are shown by the black ticks. The yellow line represents the intensity function used to generate the spike times. 111
- 3.19 Illustration of MPA, showing how a smaller time index is used for proposing the new intensity function ($A \rightarrow B$) and how projection is used to create the intensity function on the desired time index ($A \rightarrow C$ and $B \rightarrow D$). In this example we use $Q = 20$ and $N = 200$ 118
- 3.20 A known intensity function compared to its MPA approximation using a varying values of Q . The intensity function — shown in black — is defined by $x(t) = (1.5 - t/30) \cos(t^2/15) + 2$. The MPA is calculated for Q equal to 10, 20, 50 and 200, using the parameters $\sigma_f^2 = 1000$, $\sigma_n^2 = 10^{-5}$ and $l = 1$. The projected intensity function for each Q is shown by a coloured line described in the legend. 119
- 3.21 The average time taken to simulate an intensity function over 100 samples using the Cooley-Tukey FFT algorithm applied to the spectral representation of a GP against N_{FFT} . The GP has mean zero and square exponential hyper-parameters $\sigma_f^2 = 1000$, $\sigma_n^2 = 0$ and $l = 1.3$. The points are coloured depending on the number of factors N_{FFT} has, shown by the legend. 123
- 3.22 The average time taken over 100 samples to draw an intensity function with varying length scale. We sampled functions in $[0\text{s}, 20\text{s}]$ with a step size of 0.01s and $\sigma_f^2 = 1000$. The length scale is given in percentage of the experiment length, e.g $l = 4$ is given by a length scale of 20%. . . 125

-
- 4.1 Illustration of the ISI distribution with varying values of T_{\min} . The purple, blue and yellow lines correspond to a Gamma ISI distribution with $x = 1$, $\gamma = 10$ where $T_{\min} = 0\text{s}$, 0.5s and 1s , respectively. 130
- 4.2 Illustration of the affect of including the refractory period as a model parameter on the intensity function. We simulated 1000 spike sequences from two ISI distributions. Namely, an inhomogeneous Inverse Gaussian ISI distribution with the refractory period where the ISI parameter $\lambda = 2$ and intensity function $x(t) = \sin(t/2.5) + 1.1$ — shown by the black line — and $T_{\min} = 0\text{s}$ or 0.2s . The lighter blue histogram is obtained by binning the spike sequences when $T_{\min} = 0$ and the darker blue when $T_{\min} = 0.2$. The red line corresponds to the intensity function sifted by T_{\min} as described in equation (4.4). 131
- 4.3 Illustration of the posterior inhomogeneous Exponential ISI distribution with and without the refractory period. (A,B) show samples of the posterior ISI distribution when fitted to one and ten simulated spike sequences, respectively. The black line represents the ISI distribution the simulated spike sequences were drawn from. Red and blue lines correspond to samples from the model with and without the refractory period, where the overlap is shown in purple. In (A) the ISIs used to fit the models are shown by black ticks on the x axis. 137
- 4.4 Illustration of the posterior inhomogeneous inverse Gaussian ISI distribution with and without the refractory period. (A,B) show samples of the posterior ISI distribution when fitted to one and ten simulated spike sequences, respectively. The black line represents the ISI distribution the simulated spike sequences were drawn from. Red and blue lines correspond to samples from the model with and without the refractory period, where the overlap is shown in purple. In (A) the ISIs used to fit the models are shown by black ticks on the x axis. 139

-
- 4.5 Samples of the posterior intensity function and ISI parameter from an inhomogeneous inverse Gaussian ISI distribution with and without the refractory period, shown in blue and yellow, respectively. The posterior distribution coincides with the ISI distribution shown in Figure 4.4(A), where the intensity function is constant. Above and to the right of the plot we provide probability density functions of the intensity function and ISI parameter, respectively. 140
- 5.1 Illustration of violin plots compared to box and whisker plots, for 3 datasets E , F and G . E consists of 200 draws from a uniform distribution on $[0.3, 0.99]$, $F = 1/E$ and G consists of the smallest 100 draws from E and the largest 100 from F . (A,B) show the difference from using the slope compared to the angle of the slope in radians. . . 146
- 5.2 Posterior intensity distribution and Q-Q plot using an inhomogeneous Gamma ISI distribution with a constant and GP prior — shown in black and red, respectively. (A) The 95% confidence interval and posterior mean of the intensity function fitted to a spike sequence shown by the black ticks. (B) Q-Q plot for the models shown in (A), with the slope obtained by least square linear regression. The dashed line represents the line with slope one. 147
- 5.3 Comparison of rescaled ISI in the order they occurred (A,B) and ten simulated spike sequences (C,D) for the ISI distributions with a constant (A,C) and GP prior (C,D). In (A,B) the rescaled ISIs are obtained by rescaling the ISIs of the spike sequence shown in Figure 5.2(A) using the posterior Gamma ISI distribution fitted to the same spike sequence with a constant and GP prior, respectively. In (C,D) each black row corresponds to a simulated spike sequence and the red row corresponds to the spike sequence used to fit the models. 149

-
- 5.4 Raster of Ca^{2+} spike sequences obtained by stimulating astrocytes with $10\mu\text{M}$ and $100\mu\text{M}$ glutamate in (A,B), respectively. Each Ca^{2+} spike sequence corresponds to the response from a unique astrocyte. We split the datasets into the Ca^{2+} spike sequences with at least 15 spikes and those with fewer spikes, shown by the red and black points, respectively. The solid black line represents the length of the experiments. 151
- 5.5 Comparison of the posterior of the inverse Gaussian ISI distribution with a PWC prior fitted to datasets of Ca^{2+} spike sequences when the applied stimulus was $10\mu\text{M}$ (black) and $100\mu\text{M}$ (red) glutamate. For each we plot the mean and 95% credible region of the posterior intensity function distribution and a box plot summary of the posterior ISI parameter distribution. 153
- 5.6 (A) 100 Samples taken from the posterior inverse Gaussian ISI distribution with a PWC prior fitted from the dataset of astrocytes challenged with $10\mu\text{M}$ glutamate — shown in Figure 5.4(A). (B) Comparison of posterior intensity functions fitted either from individual Ca^{2+} spike sequences (black) or the dataset of all Ca^{2+} spike sequences (red). For each ISI distribution fit from an individual Ca^{2+} spike sequence we show the mean of the posterior intensity function distribution and the average over these intensity functions is shown by the yellow line. For the ISI distribution fit from the dataset of all Ca^{2+} spike sequences we present the mean and 95% credible region of the posterior intensity function distribution. 156
- 5.7 Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models using the data of 24 astrocytes challenged with $10\mu\text{M}$ glutamate. 159

-
- 5.8 Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models using the data of 29 astrocytes challenged with 100 μ M glutamate. 160
- 5.9 Comparison of the posterior Weibull ISI distribution with a PWC prior fitted to individual Ca²⁺ spike sequences from astrocytes cells challenged with 10 μ M (black) and 100 μ M (red) glutamate. Each Ca²⁺ spike sequence comes from a unique astrocyte. For each fit we plot the mean of the posterior intensity function distribution. We also plot the mean of the posterior ISI parameter distribution, where we produce box plots for each stimulus strength. 161
- 5.10 Comparison of posterior ISI distributions when approximated by a constant intensity over the region [400s, 1100s]. This is shown for 5 posterior Weibull ISI distributions inferred from individual Ca²⁺ spike sequences (black) and the fit from the whole dataset (red) for cells challenged with 100 μ M glutamate. 162
- 5.11 Raster of Ca²⁺ spike sequences obtained by stimulating HEK293 cells with 10 μ M and 30 μ M carbachol in (A,B), respectively. We split the dataset into the spike sequences with at least 15 spikes and those with fewer spikes, shown by the red and black points, respectively. The black line shows the length of the experiments. We provide a yellow box in (A) to indicate a region containing fewer spikes than expected. 163
- 5.12 Comparison of the posterior inverse Gaussian ISI distributions with a PWC prior fitted from the datasets where the applied stimulus was 10 μ M (red) and 30 μ M (black) carbachol, respectively. We present the mean and 95% credible region of the posterior intensity function distributions and box plots of the posterior ISI parameter distributions. 165

-
- 5.13 Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models using the data of 56 HEK293 cells challenged with 30 μ M carbachol. 167
- 5.14 Overview of the posterior inverse Gaussian ISI distribution with PWC prior fitted to individual Ca²⁺ spike sequences. The Ca²⁺ spikes come from HEK293 cells challenged with 10 μ M and 30 μ M carbachol in (A) and (B), respectively. For each fit we plot the mean of the posterior intensity function distribution. We also plot the mean of the posterior ISI parameter distribution, where we produce box plots for each stimulus strength. The ISI parameters are coloured from purple to yellow depending on their value. The matching intensity function is shown in the same colour. The posterior mean intensity function and box plot of the ISI parameter fitted from the datasets of Ca²⁺ spike sequences are shown in black — this corresponds to the posterior mean and boxplots shown in Figure 5.12. 168
- 5.15 (A) Fura-2 fluorescence intensity traces of a HEK293 cell challenged initially with 20 μ M carbachol and then with 100 μ M carbachol, as shown in the bar. The corresponding spike times are shown as red ticks on the x-axis. (B) Raster of Ca²⁺ spike times obtained from HEK293 cells in a step-change experiment where the stimulus strength 20 μ M \rightarrow 100 μ M. The black line represents the length of the experiments across the dataset. 172
- 5.16 Comparison of the posterior of the inhomogeneous inverse Gaussian ISI distribution with a PWC prior fitted from datasets of HEK293 cells under a step-change experiment. We show the mean of the posterior intensity function distribution together with the 95% credible region. We also present box plots of the mean of the ISI parameter over all spike sequences in the datasets. 174

-
- 5.17 Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models using the data of 78 HEK293 cells subject to a step-change experiment with varying stimulus strengths. 175
- 5.18 Comparison of the posterior intensity function for the inhomogeneous Weibull ISI distribution with a PWC prior fitted to individual Ca^{2+} spike sequences obtained from HEK293 cells under a step-change experiment. (A,B,C,D) correspond to the mean of the posterior distributions fitted to spikes sequences in the datasets where the stimulus strengths were $10\mu\text{M} \rightarrow 50\mu\text{M}$, $20\mu\text{M} \rightarrow 50\mu\text{M}$, $20\mu\text{M} \rightarrow 100\mu\text{M}$ and $50\mu\text{M} \rightarrow 100\mu\text{M}$, respectively. 176
- 5.19 (A) An example of the wave stimulus applied to a HEK293 cell (I) and the cell's Ca^{2+} response (II). (B) Raster of Ca^{2+} spike times obtained from HEK293 cells challenged carbachol which varies over three waves. The black line indicates the length of the experiment. 178
- 5.20 Comparison of the posterior of the inhomogeneous Gamma ISI distribution with a PWC prior fitted to individual Ca^{2+} spike sequences and the whole dataset. (B) the posterior mean of the intensity function for each fit from an individual Ca^{2+} spike sequence. The mean of these intensity functions (yellow) is shown in (A) compared with the posterior mean and 95% credible region of the intensity function fitted from the entire dataset (red). (C) comparison of the posterior ISI parameter when fitting the Ca^{2+} spike sequences collectively (red) or individually (grey). The ISI parameters are coloured depending on their size and their corresponding mean intensity function is coloured the same. 180
- 5.21 (A) The applied pulse stimulus (I) and three trace plots of HEK293 cells response to the stimulus (II). (B) Raster of Ca^{2+} spike sequences obtained from HEK293 cells challenged with pulses of carbachol. The black line represents the length of the experiment. 182

-
- 5.22 (A) The posterior mean and 95% credible regions for the inhomogeneous Gamma ISI distribution with a Constant, PWC and GP prior, shown in yellow, green and purple, respectively. (B) 10 Simulated spike sequences for each of the posterior ISI distributions shown in (A). 184
- 5.23 Visualisation of simulating spike sequences using ISI distributions fitting to HEK293 cells challenged with 10 μ M carbachol. (A) the intensity function — created by combining individual and collective fits — and ISI parameter pairs used to simulate spike sequence. (B) The Ca²⁺ spike sequences obtained from HEK293 cells, used to fit the ISI distributions in (A). (C) 50 simulated spike sequences from the ISI distributions shown in (A). 187
- 5.24 Illustration of simulating spike sequences using parameter values from previous or similar experiments. (A) visualisation of the intensity functions required for an experiment where HEK293 cells are challenged with two step changes in stimulus. In particular, cells challenged with 10 μ M carbachol at 500s and the stimulus is exchanged at 2500s and 5000s for 25 μ M and 50 μ M, respectively. (B) Raster plot of 50 simulated spike sequences using the intensity functions with the shape shown in (A). 188
- 6.1 An illustration of how we generate intensity functions from experimental data. For a single cell we begin with a Fura-2 fluorescence intensity trace (A) — the changes in the fluorescence of the Ca²⁺ indicator is recored relative to its basal level (ΔF) — which is thresholded to calculate Ca²⁺ spike times (B). The Ca²⁺ spike times are used as an input into our model which returns the posterior distribution of the intensity function (C). The posterior distribution in (C) shows the mean (black line) with 95% credible interval (grey region). In (D) we show the mean of the posterior intensity function distribution obtained by fitting our model to 14 individual cells. 192

6.2	Example of clustering using k-means with 3 clusters. Each point on the graph corresponds to a single intensity function. The points are clustered into 3 groups shown by their colour — blue, black and red. The crosses correspond to the center of each cluster.	195
6.3	Example of clustering methodology applied to three functions. The black and blue lines are grouped together as they have similar shape. Whereas, although the red and black lines have a similar intensity they will not be grouped together as they have different shapes.	196
6.4	Clustering a dataset containing constant functions and functions with a single bump. In (A) we cluster using the method in Tilunaite and (B) Delaigle.	198
6.5	The first 8 Haar bases on the interval $[0s, 20s]$	200
6.6	Approximating a function via the Haar basis. The function (black line) is approximated by using 2 (red), 8 (orange) and 32 (blue) Haar bases.	204
6.7	Functions (black lines) drawn from a GP on the interval $[0s, 20s]$ with step size equal to 0.01s and length scale equal 2 (A), 4 (B) and 10 (C). The red piecewise constant functions show the approximation given by 16 Haar bases. The intervals of the finest 8 Haar bases are visualised by the regions separated by the grey lines.	208

6.8	Visualisation of extracting features for the functions in Figure 6.7. In step 2 we highlight the bases whose coefficient is larger than the threshold — 0.18, 0.13 and 0.08 for functions 1, 2 and 3 respectively. In step 3 the orange box symbolised the base added into the important bases, when checking for a peak. In step 4 we split the bases into consecutive indices, hence splitting the bases for function 1 into two groups, shown in light and dark blue. In step 5 each group from step 4 is partitioned into negative (red) and positive values (yellow). The boxes surrounding the bases shows the different features found by looking at the pattern of sign changes. Function 1 has two features; a bump in bases {9, 10, 11, 12} and another bump in bases {15, 16}. Function 2 also has two features; a bump in bases {9, 10, 11, 12, 13} and an increasing region in bases {14, 15, 16}. Function 3 has one feature; a bump in the bases {9, ..., 16}.	209
6.9	(A) shows the functions to be clustered, where the first 10 functions (black lines) have a single bump in [0, 10], functions 11-20 (red lines) have a single bump in [10, 20], and functions 21-30 (blue lines) have a bump in each region. (B) shows the resultant features of each function and (C) the magnitude of the features.	216
6.10	Clusters of the simulated dataset, where clustering depends on position and magnitudes of features explained in Table 6.4.	219
6.11	Fura-2 fluorescence intensity traces of two HEK293 cells stimulated with 50 μ M carbachol. Changes in the fluorescence of the Ca ²⁺ indicator is recored relative to its basal level (ΔF).	221
6.12	Example of removing the trend from ISIs, where the ISIs with trends are shown by black dots and the ISIs with linear trends removed are red. The solid lines show the linear trends in the respective ISIs. . .	221

6.13	Illustration of the posterior intensity distribution when the model is fitted to an individual Ca^{2+} spike sequence with (black) and without (red) linear trends in the spike times removed. This is shown for two cells in (A) and (B), respectively. The solid lines and shaded regions correspond to the mean of the posterior intensity function distribution and 95% credible region.	222
6.14	The mean of the posterior intensity function distribution obtained from fitting individual Ca^{2+} spike sequences to the inhomogeneous Gamma ISI distribution. The Ca^{2+} spike sequences were obtained from HEK293 cells challenged with constant stimulus.	224
6.15	Matrix containing the extracted features for the constant stimulus experiments.	226
6.16	Matrix containing the magnitudes of the features for the constant stimulus experiments.	227
6.17	Histogram of the magnitudes of all the features found from the intensity functions inferred from the constant stimulus experiments.	228
6.18	Clusters of the intensity functions inferred from a constant stimulus experiment. (A) and (B) corresponds to clusters with and without bumps respectively. Whereas, (C) and (D) corresponds to clustering on functions that do and do not rebound after the initial decrease in intensity respectively.	229
6.19	(A) shows the intensity functions inferred from the step change experiment. (B) shows the extracted features of each function and (C) the feature's magnitude. The functions in (A) are coloured corresponding to the row that describes their features in (B) and (C).	231
6.20	Clusters of the intensity functions obtained from a step change experiment. We cluster into 3 groups; (A) functions with a large bump, (B) functions that have a decrease after the bump in [3000s, 4000s] and (C) those that do not have a decrease afterwards.	233

7.1	Screenshot of the app for viewing/analysing raw data when we threshold at 0.7 .	239
7.2	The concentration and ISI plots shown in the app when we (A) change the threshold to ' $0.7, 0.45, 0.7, 800, 1000$ ' and (B) add two transient regions.	241
7.3	Screenshots of the concentration plot for the second cell using a threshold of 0.85 . The closeness parameter is set to 0 and 10 in (A) and (B), respectively.	243
7.4	Screenshot of the simulating spikes app.	245
7.5	Screenshot of the results panel for the second example where the intensity function was $c(t[t < 20] * 0 + 1, t[t > 20] * 0 + 0.5)$.	246
7.6	Screenshot of the opening panel of the app.	248
7.7	Screenshots of the feature (top) and magnitude (bottom) tables taken from the app.	249
7.8	Screenshot of the <i>Individual tab</i> taken from the app.	250
B.1	Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from an astrocyte challenged with $10\mu\text{M}$ glutamate.	266
B.2	Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from an astrocyte challenged with $10\mu\text{M}$ glutamate. The cell used differs from the one used in Figure B.1.	267
B.3	Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from an astrocyte challenged with $100\mu\text{M}$ glutamate.	268

B.4	Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from an astrocyte challenged with $100\mu\text{M}$ glutamate. The cell used differs from the one used in Figure B.3.	269
B.5	(A) 50 simulated spike sequences from the posterior inverse Gaussian ISI distribution with a PWC prior fitted from the datasets where the applied stimulus $30\mu\text{M}$ carbachol. (B) The dataset of Ca^{2+} spike sequences used to fit the model used to simulate spikes in (A). In both the spike sequences are ordered such that the sequences with the most spikes occur at the top of the plots.	271
B.6	Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with $10\mu\text{M}$ carbachol.	272
B.7	Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from HEK293 cells challenged with a step change of carbachol of strength $10\mu\text{M} \rightarrow 50\mu\text{M}$	274
B.8	Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from HEK293 cells challenged with a step change of carbachol of strength $10\mu\text{M} \rightarrow 50\mu\text{M}$. The cell used was different from that used in Figure B.7.	275
B.9	Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with a step change of carbachol of strength $10\mu\text{M} \rightarrow 50\mu\text{M}$	276

B.10 Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with a step change of carbachol of strength $20\mu\text{M} \rightarrow 50\mu\text{M}$	277
B.11 Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with a step change of carbachol of strength $20\mu\text{M} \rightarrow 100\mu\text{M}$	278
B.12 Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with a step change of carbachol of strength $50\mu\text{M} \rightarrow 100\mu\text{M}$	279
B.13 Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with three waves of carbachol.	280

List of Tables

3.1	The MLE and log likelihood of the MLE of each distribution fit to Ca^{2+} ISIs from a HEK293 cell challenged with $10\mu\text{M}$ carbachol. The colour of the distribution matches the colour of the probability densities shown in Figure 3.1.	73
3.2	Parameterisations of the inhomogeneous ISI distributions which lead to non-identifiability, where $a > 0$ is a constant.	77
3.3	One parameter version of the stationary ISI distributions with mean one.	79
3.4	The average time taken to compute a single MCMC loop using either a GP prior or a PWC prior over 1000 iterations. We fitted an inhomogeneous Gamma ISI distribution to a single spike sequence with 34 spikes. The ISI parameter had an exponential prior with rate 0.001. The inference parameters for the GP prior were: $N = 2000$, $w = 0.001$, $\sigma_h^2 = 1$, $\sigma_f^2 = 1000$ and $\sigma_n^2 = 10^{-5}$. The prior for the length scale l was an exponential with rate 0.01, the initial value was 1.5 and $\sigma_l^2 = 0.25$. For the PWC prior the parameters were: $k_{\max} = 25$, $\lambda = 10$, $\kappa = 1$, $\mu = 2$, $\kappa_0 = 1$ and $\sigma_h^2 = 1$. The column denoted ‘Other’ represents an update of both a height and a change point.	114

3.5	Comparison of the average time taken for a single MCMC iteration using varying approaches when l has a prior distribution.	126
6.1	The coefficients of the Haar bases for function f drawn from a GP shown in Figure 6.6.	203
6.2	The coefficients of bases h_9, \dots, h_{16} for the three functions shown in Figure 6.7.	207
6.3	Example of vector representation for the features and magnitudes of the function shown in Figure 6.7(A).	214
6.4	Six clusters of the example dataset, clustered on feature position and magnitude.	218
6.5	Features of intensity functions generated from the same cell, shown in Figure 6.13. Raw and no trend corresponds to the intensity function when the model was fitted from the original Ca^{2+} spike sequence and the spike sequence with linear trends removed, respectively.	223

CHAPTER 1

Introduction

Calcium (Ca^{2+}) is a highly versatile intracellular messenger that contributes to numerous biological functions such as cell maturation, [7], gene expression [8] and orchestrating fertilisation [9, 10, 11]. Along with these functions excessive Ca^{2+} can damage cells [12, 13] and alterations in the Ca^{2+} response of cells can contribute to the onset of multiple human diseases [14].

Ca^{2+} controls these biological functions by transient rises and falls of Ca^{2+} concentration in the cell — commonly known as Ca^{2+} oscillations or Ca^{2+} spikes. These oscillations encode information that the cell is presented with such as hormones or neurotransmitters.

To visualise the Ca^{2+} encoding of information we show the Ca^{2+} response of human embryonic kidney (HEK293) cells challenged with carbachol. This is shown in Figure 1.1 for three different stimulation protocols. In Figure 1.1(A) we present the Ca^{2+} concentration for two cells that experience the same stimulus — $10\mu\text{M}$ carbachol. We see that the two responses are considerably different with one cell exhibiting 11 Ca^{2+} spikes and the other 32 Ca^{2+} spikes. This illustrates the cell-to-cell variability of the Ca^{2+} response in cells challenged with the same stimulus. Moreover,

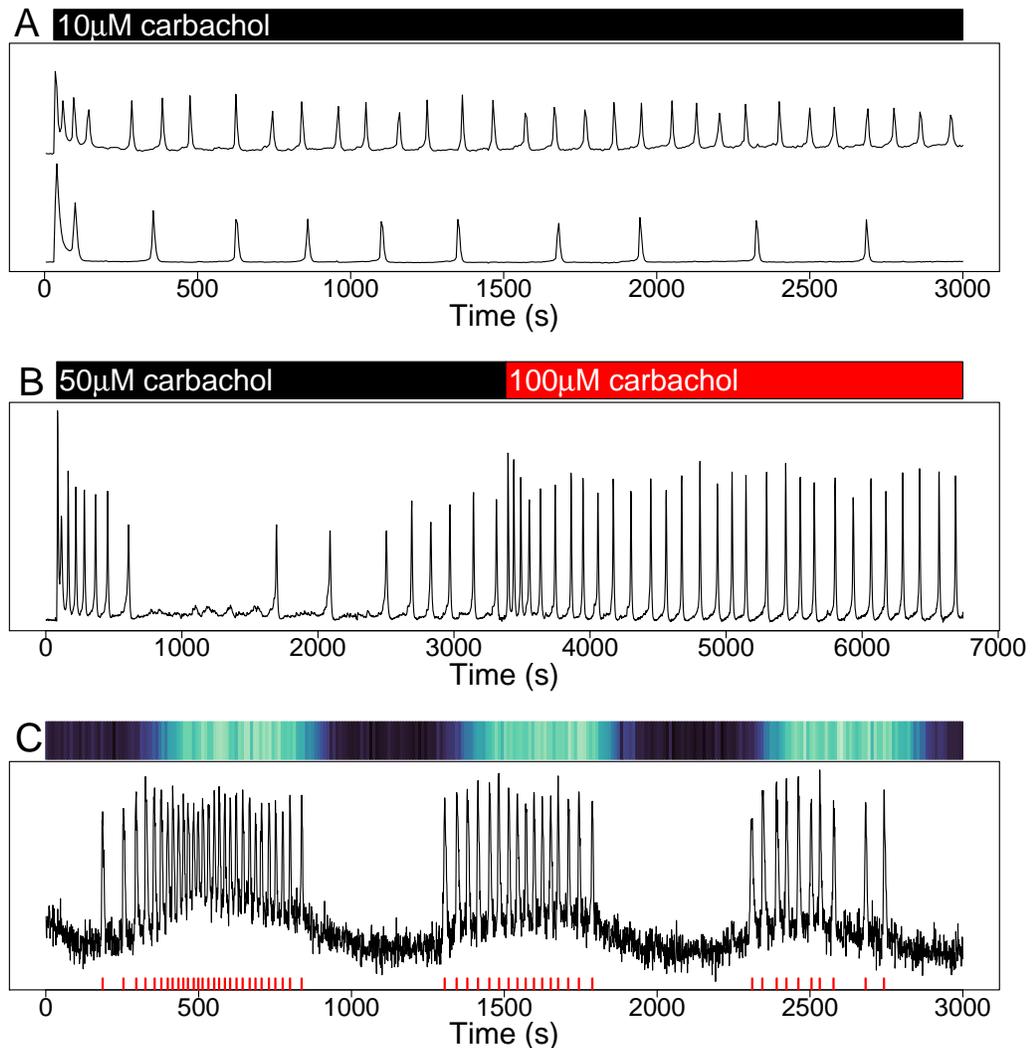


Figure 1.1: Ca²⁺ oscillations of HEK293 cells for a variety of stimulation protocols as shown by the bars. (A) Two cells challenged with 10 μM carbachol. (B) A cell initially challenged with 50 μM carbachol and then with 100 μM carbachol at 3380s. (C) A cell challenged with three waves of carbachol. This is shown in the coloured bar where the lighter colour corresponds to a larger concentration of carbachol. The red ticks on the time axis correspond to the Ca²⁺ spike times.

by presenting three different stimulation protocols in Figure 1.1(A,B,C) we can see that the Ca²⁺ response of the cell depends on the timing and strength of the stimulus

arriving to it. One example of time dependence is that we see more Ca^{2+} spikes closer to the onset or exchange of the stimulus in Figure 1.1(A,B). This data illustrates the versatility of Ca^{2+} oscillations for a single cell type and different types of stimulus. In addition the Ca^{2+} response also varies depending on cell type.

Figure 1.1 illustrates that Ca^{2+} spikes vary in amplitude and frequency. As we discuss in Section 1.1.1.1 both features are thought to encode information about the applied stimulus. For the purpose of the thesis, we will focus on the frequency of Ca^{2+} oscillations and hence base our analysis on the times when Ca^{2+} spikes occur.

We want to understand how Ca^{2+} oscillations differ between cell types and how the response is affected when the signal is time-dependent, as in real physiological conditions. Doing so would improve the understanding of the underlying dynamics driving the Ca^{2+} response, which in turn could enhance our knowledge of the Ca^{2+} signalling toolbox. In particular, considering time-dependent stimulus will provide insight into the interplay between the timing of signals and the biological processes that the signals control.

In this thesis, we take a data-driven approach to capture the behaviour of Ca^{2+} oscillations. This involves fitting models using Ca^{2+} spike sequences, such as the one shown by red ticks in Figure 1.1(C). One advantage of this approach is that it can be used to capture the Ca^{2+} response of any cell type or stimulation protocol, as it only requires the Ca^{2+} spike sequence. This flexibility will allow us to compare models fitted from different cell types and find differences in the Ca^{2+} response, whereas mechanistic models of different cell types may require cell-type specific models to capture the intrinsic difference in the Ca^{2+} response — such as cells expressing different parts of the Ca^{2+} signalling toolbox¹.

A natural starting point is to treat a Ca^{2+} spike sequence as a realisation of a point process. Indeed, Tilunaite et al. [16] developed such a framework for Ca^{2+}

¹Elements of the Ca^{2+} signalling toolbox control Ca^{2+} concentration in the cell such as exchangers or pumps, for more information see [15].

spike sequences, where the point process is defined in terms of an inhomogeneous inter-spike interval (ISI) distribution. To account for the time-dependence of the Ca^{2+} spikes they employed so-called intensity functions. Here, we will develop further the statistical framework used by Tilunaite et al., along three critical ways.

Firstly, we extend the modelling framework to include more ISI distributions. By including more ISI distributions we find new parameterisations that better describe the Ca^{2+} oscillations. This could in turn lead to a greater understanding of what drives the Ca^{2+} response. In the current framework the ISI distributions do not directly contain the refractory period — the minimum time after a Ca^{2+} spike where another spike cannot occur. To understand more of the underlying mechanisms controlling the Ca^{2+} oscillations it may therefore be advantageous to explicitly contain the refractory period in our ISI distributions. Thus, we will also investigate the advantages and disadvantages of including the refractory period as an additional model parameter.

Secondly, the statistical framework developed here uses Bayesian inference to fit the models. In this approach each parameter requires a prior distribution containing our belief of the values the parameter takes before seeing the data. This is then used in combination with the data to obtain a posterior distribution of the parameter containing the beliefs after seeing the data. As a model parameter, the intensity function therefore requires a prior distribution. Tilunaite et al. (2017) utilised Gaussian processes for the prior of the intensity function. The advantage of Gaussian processes comes from the fact that they are non-parametric and as such they avoid imposing strict restrictions of the types of functions allowed. For example constraining the intensity function to be linear. However, this added flexibility comes with a large computational cost. In particular, Therefore, in addition to Gaussian processes it would be beneficial to extend the modelling framework to allow for other prior distributions for the intensity function. In this thesis, we provide such priors in the form of the constant prior and the piecewise constant prior. They are less flexible than the Gaussian process but are substantially cheaper to work with.

Tilunaite et al. approximated the posterior distribution by using Laplace's approximation. Using an approximation may lead to inaccuracies in the posterior

distribution. Therefore, to avoid such issues, in this thesis we develop Markov chain Monte Carlo (MCMC) algorithms to sample from the true posterior distribution.

We then apply our methods to Ca^{2+} spike sequences obtained from HEK293 cells challenged with a variety of stimulation protocols and astrocytes challenged with constant stimulus. We investigate whether the extended modelling framework is able to more accurately capture the dynamic nature of Ca^{2+} oscillations. We also use our methods to explore the differences and similarities of Ca^{2+} oscillations across different cell types (HEK293 cells and astrocytes) and stimulation protocols.

Pivotal to learning about the heterogeneity of Ca^{2+} oscillations in our model is the intensity function, which describes the mean spiking rate over time. Therefore, one method to help understand the cell-to-cell variability in Ca^{2+} oscillations is to look for common features or patterns in the intensity functions, thereby clustering them.

Importantly, cell-to-cell variability is often found in the magnitude of the Ca^{2+} response. For instance, picture two intensity functions which are both linear with the same gradient but whose intercepts are far apart. In this case, the intensity functions share the same shape but different magnitudes. In the context of Ca^{2+} signalling it would be insightful to group together such functions because they have the same shape. For example it would be preferable to group together intensity functions that all have a peak in intensity at the same point in time, as this may provide information of underlying processes controlling the Ca^{2+} oscillations. Common clustering methods struggle to group such responses together, because they cluster on the magnitude rather than the shape. Therefore, it is desirable to develop a clustering approach that finds patterns in the shape of the intensity functions. We develop such a clustering approach and use our method to cluster intensity functions inferred from Ca^{2+} spike sequences, obtained from HEK293 cells.

Finally, we have also created web applications to work in conjunction with the

methods developed. This will give practitioners a simple interface to apply our methods with their own experimental data.

1.1 Ca^{2+} signalling background

Ca^{2+} is needed for the normal function of cells and their survival. The importance of Ca^{2+} in animal cells was first discovered by Sidney Ringer, when he showed that Ca^{2+} was required for the normal contraction of frog hearts [17]. In 1882, he originally reported the opposite result that Ca^{2+} was not necessary for the beating frog heart [18]. However, afterwards he discovered that pipe water rather than distilled water was accidentally used in the original experiment. Repeating the experiment with distilled water his results differed from the original experiment. Thus, he proposed that the effects were caused by inorganic constituents of the pipe water, which was later revealed to be Ca^{2+} .

In the years since that discovery, it is known that in addition to influencing the heart beat, Ca^{2+} has 4 main biological roles [19]:

- 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) [20].

In this thesis we consider Ca^{2+} 's role as an intracellular second messenger, or to be more precise we investigate how changes to the Ca^{2+} concentration inside a cell informs on the applied stimulus. In particular we explore the effect of time-dependent stimulus on the Ca^{2+} concentration of single cells. In this chapter, we begin by introducing intracellular Ca^{2+} signalling. We then briefly review the literature of mathematical models used to analyse Ca^{2+} signals. After which, we explain how experimental Ca^{2+} concentrations are recorded and describe the data used throughout.

1.1.1 Calcium Signalling

In its simplest form cells have a cytosolic baseline Ca^{2+} concentration about 100nM. Through components of the ‘ Ca^{2+} signalling toolbox’ — for example Ca^{2+} pumps that move Ca^{2+} out of the cell or from one cellular compartment to another — the level of Ca^{2+} varies in the cell. These changes in Ca^{2+} concentration are used to regulate many processes due to its versatility, i.e. its speed, amplitude and spatio-temporal patterning [15].

The overarching idea of the Ca^{2+} signalling network can be split into 4 sections. Firstly, signalling is triggered by a stimulus — such as a hormone or neurotransmitter — that generates various Ca^{2+} -mobilising signals. Secondly, these signals activate channels and pumps that feeds Ca^{2+} into the cytoplasm from outside the cell and internal stores. Thirdly, Ca^{2+} functions as a messenger to stimulate numerous Ca^{2+} -sensitive processes — such as muscle contraction or egg fertilisation. Finally, The OFF mechanism activates leading to the removal of Ca^{2+} — via pumps and exchangers — from the cytoplasm. This returns the Ca^{2+} concentration to its basal level.

We illustrate the process by considering the inositol-1,4,5-trisphosphate (IP_3) Ca^{2+} pathway that releases Ca^{2+} from the endoplasmic reticulum (ER) — an internal store of Ca^{2+} — leading to changes (oscillations) in cytosolic Ca^{2+} concentration. This pathway has been shown to be associated with lipolysis — the breakdown of fats by hydrolysis [21]. The pathway’s effect on a cell’s Ca^{2+} concentration is visualised in Figure 1.2(A), where we depict the state of the cell at 5 stages.

For this pathway, we require several components of the ‘ Ca^{2+} signalling toolbox’. Firstly, the plasma membrane contains Ca^{2+} ATPase (PMCA) pumps that transport Ca^{2+} out of the cell and cell-surface receptors that facilitate Ca^{2+} -mobilising signals. Within the cell we have the ER, whose surface contains IP_3 receptors (IP_3Rs) and sarco-endoplasmic reticulum Ca^{2+} ATPase (SERCA) pumps. IP_3Rs transfer Ca^{2+} from the ER into the cytoplasm, whereas SERCA pumps transfer Ca^{2+} from the cytoplasm into the ER. An IP_3R has a tetrameric mushroom-like structure with four subunits [3]. A 3D representation is shown in Figure 1.2(C). The dynamics of the receptor has been modelled by conjecturing that each subunit contains one Ca^{2+}

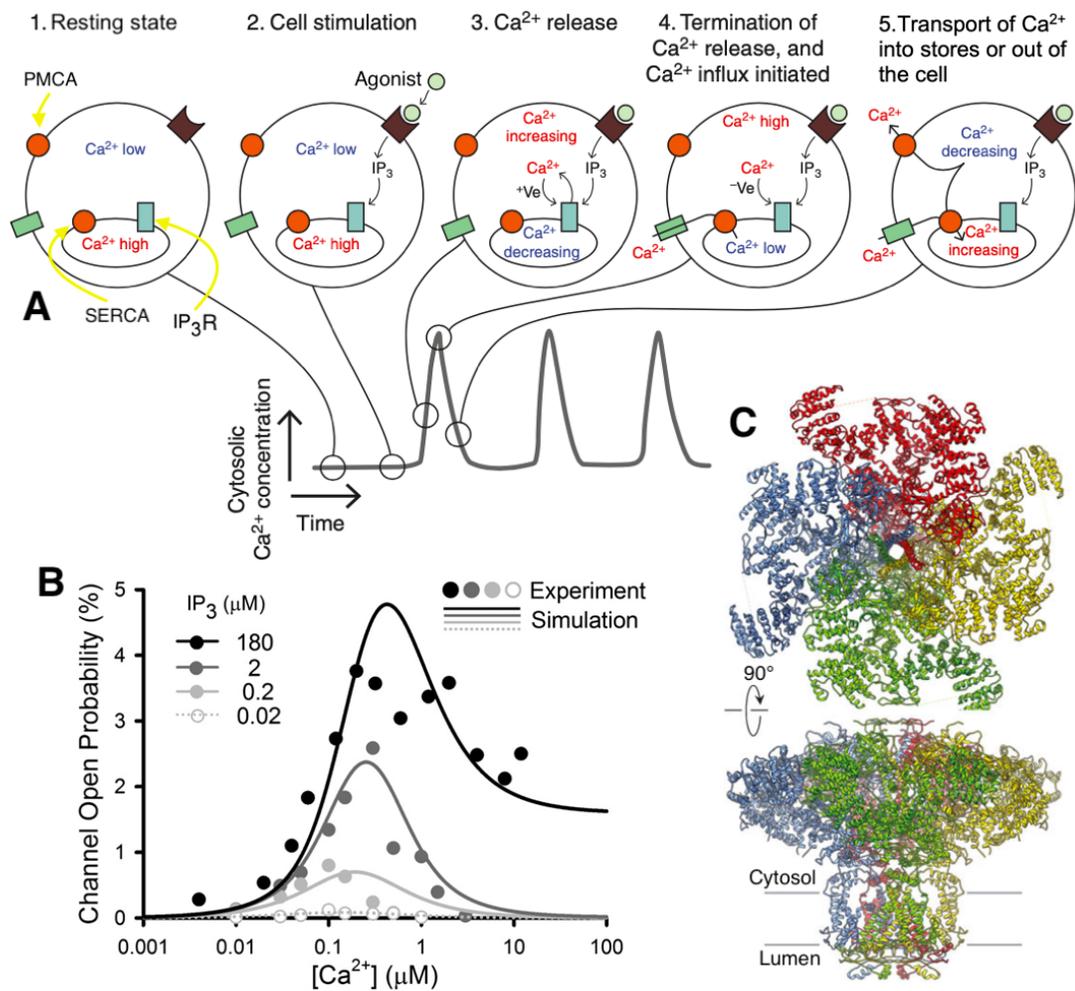


Figure 1.2: (A) Diagram is adapted from [1]. (B) Graph taken from [2]. (C) Image taken from [3]. (A) Illustration of Ca^{2+} oscillations generated by the IP₃ Ca^{2+} pathway. (B) Open probability of IP₃R as a function of Ca^{2+} concentration shown for varying IP₃ concentration. Filled dots show experimentally recorded values and simulation results by the solid lines. (C) 3D IP₃R structure drawn from two perspectives: the cytosol and along the membrane plane.

activation site, one Ca^{2+} deactivation site and one IP₃ binding site [22] and see Falcke and Sneyd for further models [23, 24]. The receptors are distributed on the ER's surface in random clusters, where each cluster contains between 5 and 30 IP₃R per

cluster [25]. On average there are 8 IP₃Rs per cluster [26].

In stage one the cell is at rest, where the cytosolic Ca²⁺ concentration is low and the ER's Ca²⁺ concentration is high.

In stage two an agonist — such as carbachol — arrives to the cell, which leads to the production of IP₃, a Ca²⁺-mobilising signal. Specifically, the agonist binds to the muscarinic M₃ 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 IP₃ (and diacyl-glycerol (DAG)), via the hydrolysis of PLC and phosphatidylinositol 4,5-bisphosphate (PIP₂).

In stage three IP₃ diffuses into the cytoplasm and binds to IP₃Rs situated on the ER. The IP₃R opens in response to IP₃ and Ca²⁺ binding to activation sites. The activated channels release large amount of Ca²⁺ into the cytoplasm from the ER. The influx of Ca²⁺ into the cytoplasm leads to more IP₃R opening — an example of Ca²⁺-induced Ca²⁺ release (CICR). Figure 1.2(B) illustrates this affect where the probability that an IP₃R is open increases as the Ca²⁺ concentration increases from baseline level.

However, as the cytosolic Ca²⁺ concentration increases further the open probability of IP₃Rs decreases as Ca²⁺ binds to inactivation sites on the IP₃Rs. This is represented by stage four. Note that this corresponds to the peak Ca²⁺ concentration in the cytoplasm and the ER has been depleted. At this stage Ca²⁺ flux through the SERCA pumps begins to refill the ER to prepare the cell for another oscillation in Ca²⁺ [1].

Once an IP₃R channel has closed it cannot reopen for some time — the channel is in a refractory state [27, 28]. We reach stage five and Ca²⁺ is transported out of the cytoplasm into the ER and out of the cell by SERCA and PMCA pumps, respectively [29]. This reduces the cytosolic Ca²⁺ concentration back to basel level and refills the ER.

Although not shown cytosolic Ca²⁺ is also controlled by buffers — compounds that bind free Ca²⁺ such as parvalbumins and calretinins [30]. At rest, the majority of buffers are Ca²⁺ free. This means that they are ready to bind with Ca²⁺ ions whenever the cytosolic Ca²⁺ concentration increases. Hence Ca²⁺ buffers can affect

the timing, amplitude and recovery time of Ca^{2+} transients [31, 30]. The intracellular Ca^{2+} buffering capacity depends on cell type [32, 33]. For example motoneurons have a buffering capacity approximately 50 times smaller than Purkinje neurons [31].

Repeated increases and decreases in Ca^{2+} caused by the opening and closing of IP_3Rs along with Ca^{2+} pump activity creates Ca^{2+} oscillations. Depending on the cell type the period of Ca^{2+} oscillations varies from seconds to hours. For example in toadfish swimbladder muscle cells Ca^{2+} oscillations have a period of 10ms compared to a period of approximately 1 day for circadian rhythms [34, 35]. Ca^{2+} oscillations also range from regular spiking to bursting [36].

Changes in cytosolic Ca^{2+} concentration generally does not occur uniformly across the cell. Rather local Ca^{2+} changes occur. These Ca^{2+} events are often categorised into 3 groups: blips, puffs and waves — as illustrated in Figure 1.3. A blip occurs when Ca^{2+} is released from a single receptor, whereas a puff arises when a cluster of receptors are activated. Ca^{2+} can diffuse from one puff site to surrounding clusters activating the receptors within, which leads to wave propagation. In astrocytes it has been shown that at least three puff sites need to be activated synchronously to cause a wave [37]. Intracellular Ca^{2+} waves spread throughout the cell and can transfer information from one part of the cell to another.

1.1.1.1 Encoding and decoding Ca^{2+} Patterns

Cells use spatio-temporal patterns — such as Ca^{2+} oscillations — to transmit information (both within the same cell and to surrounding cells) [38, 39, 40] and to initiate an appropriate physiological response [41, 42]. For example for some cells the frequency of Ca^{2+} oscillations are proportional to the amount of stimulus applied to the cell [43]. However, it is not yet fully understood how encoding and decoding mechanisms convey information. To date, it has been shown that Ca^{2+} responses can vary amongst different cell types [44], and even in the same cell type [45]. Along with cell type, the agonist type [45, 46, 47] and its concentration [44] also affects the Ca^{2+} response of cells.

Currently there are two main schools of thought for stimulus encoding, namely

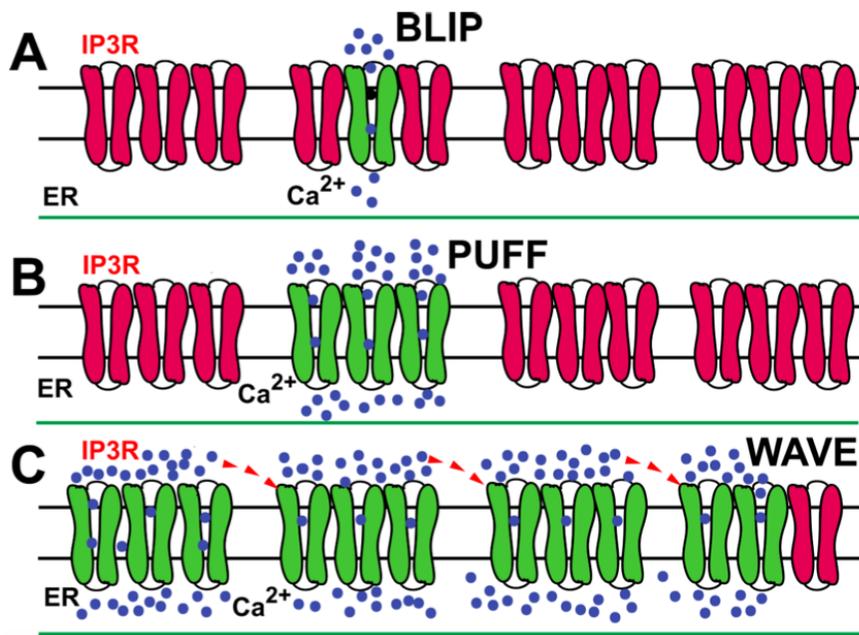


Figure 1.3: Figure taken from [4]. Illustration of Ca^{2+} events, where IP₃R channels that are active and release Ca^{2+} are shown in green. (A) represents a blip, where Ca^{2+} is released from a single receptor. (B) represents a puff, where all receptors from a single cluster are activated. (C) represents a wave, where neighbouring clusters are activated by CICR.

amplitude modulation (AM) and frequency modulation (FM) or some combination of both [48, 47]. AM proposes that the concentration of agonist increases the amplitude of Ca^{2+} signals, whereas FM claims that the frequency of the Ca^{2+} signals increase with the stimulus strength. Recent studies have shown that apoptosis can be modelled using amplitude modulation [49]. The frequency of Ca^{2+} oscillations has been shown to activate genes, affect the dilation of arteries and other events [42, 39, 50, 51, 52].

Although both are valid encoding mechanisms, under repeated constant stimulation experiments cells can exhibit a mixed response [53, 54]. For example HEK293 cells challenged with the same stimulus strength of carbachol have been shown to have a mixed response. Some cells only contain a single Ca^{2+} oscillation at the inset of stimulus, whereas others oscillate regularly but with varying frequency and

amplitude. This is an example of cell-to-cell variability.

The underlying causes of cell-to-cell variability is unknown [55], although it has been postulated that the variability could arise from differences in the cellular composition — for example it has been shown that genetically identical cells arrange components of the Ca^{2+} signalling toolbox in different spatial patterns [56]. Or the variability could come from the intrinsic stochasticity of Ca^{2+} oscillations [37].

Moreover, cell-to-cell variability could improve a systems ability to transmit information. For example it has been postulated that variability in intercellular response may increase the capacity of a system to transmit information [57].

1.1.2 Modelling Ca^{2+} oscillations

Ca^{2+} signalling is often modelled using either deterministic or stochastic models. In this section we provide a brief overview of these models. For a more detailed view, we refer readers to [27, 58, 59, 60, 24]. Often deterministic and stochastic models of Ca^{2+} oscillations follow a modelling approach known as bottom-up. This is when the model contains mechanistic details of the underlying cell such as Ca^{2+} pumps or channels. These models regularly contain a large number of equations and parameters which can come with a heavy computational cost. Therefore recently, studies have explored so-called top-down approaches. These begin with the big picture and model cellular Ca^{2+} spikes directly [61, 16, 62].

1.1.2.1 Deterministic models

As shown in Section 1.1.1, three components that control the cytosolic Ca^{2+} concentration $c(t)$ are IP_3Rs , SERCA pumps and PMCA pumps. IP_3Rs control Ca^{2+} release from the ER into the cytoplasm ($J_{\text{IP}_3\text{R}}$), whereas SERCA and PMCA pumps transport Ca^{2+} from the cytoplasm back into the ER (J_{SERCA}) and outside the cell (J_{PMCA}), respectively. Combining the fluxes, we can mathematically describe the dynamics of the cytosolic Ca^{2+} concentration by the following ordinary differential

equation

$$\frac{dc}{dt} = J_{IP_3R} - J_{SERCA} - J_{PMCA}. \quad (1.1)$$

Here, c describes the spatially averaged Ca^{2+} concentration across the cell. Note that this is not exhaustive and we could include more fluxes to describe other parts of the Ca^{2+} signalling toolbox, such as Ryanodine receptors and ORAI channels or flows into different compartments of the cell. For a review of the Ca^{2+} signalling toolbox we refer the reader to [15].

One of the first realistic models for Ca^{2+} oscillations was produced by De Young and Keizer (DYK) in 1992 [22]. The model became an archetype for later deterministic [63, 64, 65, 66] and stochastic models [67, 68].

The DYK model considers the Ca^{2+} flow into the cytosol via the IP_3R channels and a leak from the internal store, and the Ca^{2+} flow out of the cytosol via an ATP-dependent pump (i.e. the SERCA pump). The SERCA pump is described by a Hill function of $c(t)$

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

where parameters v and k represent maximum Ca^{2+} uptake and pump activation constant, respectively [22].

The IP_3R channel was modelled by a simplified version of the biological mechanics described in Section 1.1.1, where an IP_3R was modelled by three identical and independent subunits. Each subunit had one IP_3 binding site, one Ca^{2+} activation site and one Ca^{2+} inhibition site. A subunit was classed as activated when the IP_3 and Ca^{2+} activation sites were bound and the Ca^{2+} inhibition site unbound. The IP_3R channel is open when all three subunits are activated. Therefore, each subunit was modelled using eight ODEs with a total of 10 parameters. In particular the probability that one of the subunits is activated is given by x_{110} , where the index indicates that the IP_3 site is bound (110), the Ca^{2+} activation site is bound (110) and Ca^{2+} inhibition site is unbound (110). From this the flux into the cytoplasm can be computed by

$$J_{in} = c_1 (v_1 x_{110}^3 + v_2) (c_{er} - c), \quad (1.3)$$

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

The DYK model showed that activation and inactivation of the IP_3R channels is sufficient to produce cytosolic Ca^{2+} oscillations. The DYK model only accounts for changes in cytosolic Ca^{2+} concentration over time and not in space. To generalise we could incorporate a space variable x and include a diffusion term in equation (1.1)

$$\frac{\partial c}{\partial t} = D_c \frac{\partial^2 c}{\partial x^2} + J_{\text{IP}_3\text{R}} - J_{\text{SERCA}} - J_{\text{PMCA}}, \quad (1.4)$$

where D_c is a diffusion constant.

1.1.2.2 Stochastic Models

Stochastic models are considered more realistic than deterministic models due to the random nature of IP_3R channels caused by the small number of channels per cluster and variations in channel opening and closing times [67, 69, 70, 71, 72, 73, 74]. By investigating the relationship between the mean interspike interval (ISI) and standard deviation of the ISI the stochastic nature of four cell types was confirmed [61, 75, 76].

One method to account for the random dynamics of the IP_3Rs is to describe them through a Markov chain. This is then coupled to the differential equations to describe the cellular Ca^{2+} response. We refer the reader to [58, 77, 78] for further discussions.

1.1.2.3 Top-down approach

All the models described previously are classed as bottom-up approaches, where the models are built using the mechanistic understanding of processes controlling the Ca^{2+} concentration — such as IP_3R activity. Deterministic and stochastic models have played a vital part in increasing the understanding of Ca^{2+} dynamics. This includes aiding the understanding which parts of the cell influence Ca^{2+} responses and what can be done to adjust them. However even simple models often end up with a vast number of equations which can lead to computational challenges [58, 24, 72]. Moreover, complex models often have a large number of parameters that can be

difficult to estimate. For example Maurya et al. [79] suggested a model for the Ca^{2+} dynamics of RAW 264.7 cells stimulated with C5a ligand with 65 parameters. 19 of these parameters depended on the cell type.

In contrast the so-called top-down approach constructs models that directly describe key properties of Ca^{2+} oscillations such as their frequency, amplitude or duration without incorporating mechanistic detail. Although this approach cannot immediately inform on specific molecular processes it does offer distinct advantages. Namely, the computational demand is substantially smaller than bottom-up approaches. Therefore, with a top-down approach we can generate large quantities of realistic Ca^{2+} spike sequences which can be used as an input to signalling cascades that decode Ca^{2+} spikes. Moreover, top-down approaches provide a strong framework for fitting data and testing hypotheses on Ca^{2+} spike generation. In particular, information gained from top-down models can be used to improve bottom-up approaches.

Indeed, Skupin et al. [61, 80, 75] used such an approach to model Ca^{2+} oscillations for cells challenged with constant stimulus. Their model uses a time-dependent conditional rate for the Ca^{2+} oscillations containing two parameters, a time scale and an amplitude. They demonstrated that the time scale was cell type specific and different cells of the same type could be distinguished by their amplitude. However, their model can only be used for cells that experience a constant stimulation, but time-dependent stimulus is a vital ingredient to improve understanding of Ca^{2+} dynamics of cells *in vivo*.

To capture the heterogeneity of Ca^{2+} oscillations and investigate time-dependent stimulation Tilunaite et al. [16, 62] define a statistical framework which qualitatively describes the heterogeneous timing of Ca^{2+} spikes. This is done by employing so-called intensity functions to capture the time-dependence of the Ca^{2+} spiking rate. The model is defined by two parameters the ISI distribution and the Ca^{2+} spike rate. Bayesian inference is used to determine the most likely model parameters. In their work, they found that an inhomogeneous Gamma ISI distribution best describes Ca^{2+} oscillations.

1.1.3 Ca^{2+} imaging

The most common and widely used method to capture the Ca^{2+} response of cells is by fluorescent Ca^{2+} indicators — first introduced by Tsien et al., in 1980 [81]. The indicators are special molecules that emit changes in fluorescence when bound with Ca^{2+} . Thus, changes in Ca^{2+} concentration are mirrored by changes in fluorescent intensity. To make use of Ca^{2+} indicators they need to be loaded into the cell or tissue of interest. The changes in fluorescence are captured by using a scientific camera setup with a fluorescence microscope.

There are a large variety of Ca^{2+} indicators available such as Fura-2, Fluo-8 and GCaMP [82, 83]. The Ca^{2+} indicator used often depends on the experimental details, such as length, type of cell, or if the experiment is *in vivo*. The indicators can be split into two groups: chemical indicators and genetically encoded calcium indicators (GECI). An advantage of GECI is that the cellular location of the indicators is easier to control compared to chemical indicators. However, all experimental data we consider uses chemical indicators, whose advantages include the ease at which these dyes can be introduced into the experiment and the large range of Ca^{2+} affinities that are commercially available [82]. Chemical indicators can be grouped into single wavelength indicators or dual-wavelength indicators such as Fluo-8 and Fura-2 respectively [84]. The difference lies in the fact that dual-wavelength indicators have two excitation wavelengths whereas single wavelength indicators only have one. Dual-wavelength indicators are often used for quantitative measurements of Ca^{2+} concentration. However, the difference in intensities of Ca^{2+} -free and Ca^{2+} -bound indicators are often smaller than single wavelength indicators, making it harder to detect smaller Ca^{2+} changes.

In Figure 1.4 we provide an example of fluorescence microscopy, where each image shows the fluorescence at a single point in time. The white arrows indicate cells whose fluorescence varies over time. By combining these images over all time we can create a video of fluorescence, which shows how the Ca^{2+} concentration varies in the region recorded. Moreover, we can calculate time-series information for regions of interest by calculating the fluorescence ratio in the region. Regions of interest

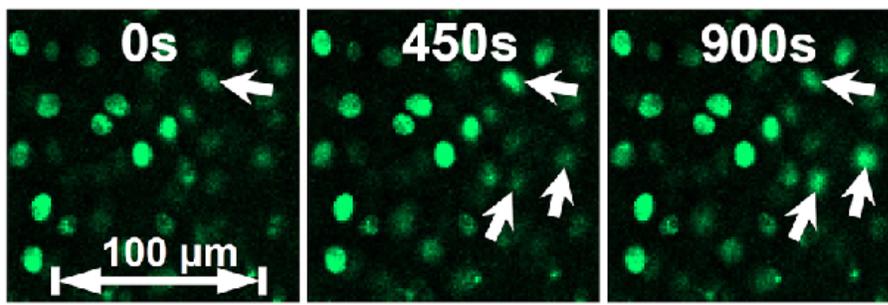


Figure 1.4: Figure taken from [5]. Illustration of Ca^{2+} imaging. Images of chondrocytes in a cartilage explant that were dyed with the Ca^{2+} indicator Fluo-8 AM. Bright green regions correspond to a larger Ca^{2+} concentration. Responsive cells — cells whose Ca^{2+} concentration changes — are marked with a white arrow.

often include whole cells or compartments of cells. Calculation of the fluorescence ratio depends on the type of chemical indicator used. If a single wavelength indicator is used the fluorescence ratio is given by the current fluorescence divided by the fluorescence recorded at the start of the experiment. If a dual-wavelength indicator is used the fluorescence ratio is calculated using the fluorescence at each wavelength. For example Fura-2 data is often shown by a 340/380 ratio. Depending on the Ca^{2+} indicator used the fluorescent ratio can be converted into the Ca^{2+} concentration, for example see [85].

It is important to note that Ca^{2+} indicators are also Ca^{2+} buffers — molecules that bind to free- Ca^{2+} . As such the indicators can impact the levels and kinetics of Ca^{2+} signalling in the cell. Other issues include cell damage, photobleaching and indicator leakage. For more information on Ca^{2+} indicators we refer the reader to [82, 83, 84].

1.1.3.1 Experimental data used in this thesis

Throughout our analysis we use two cell types, human embryonic kidney (HEK293) cells and astrocytes.

HEK293 cells were originally derived from aborted embryo kidney cells in 1973 by Alex van der Eb's laboratory [86]. Since the 1980s HEK293 cells have been used to derive new cell line such as HEK293T and HEK293S [87]. HEK293 cells are often used due to the fact they are easy to grow, maintain, transfect and they have a fast division rate [88].

Astrocytes belong to a group of cells known as glial cells. They are non-excitabile cells which communicate with each other and neurons through Ca^{2+} signalling initiated by external stimuli, such as neurotransmitters or hormones. Astrocytes, were named after their star shape by Michael von Lenhossek [89]. Recent research has found that the size and shape of astrocytes depend on the age, location and species of astrocyte [90, 91]. Moreover, astrocytes have been found to contribute to functions of the central nervous system (CNS) such as the blood-brain barrier and local CNS blood flow regulation [92]. Astrocytes are also involved in various diseases including glaucoma, Parkinson's, Huntington's and other neurodegenerative diseases [90, 93, 94].

In this thesis we analyse Ca^{2+} concentration data obtained from fluorescence microscopy of HEK293 cells and astrocytes. In particular, we analyse both cell types under a constant stimulation protocol with varying stimulus strengths. For HEK293 cells we also analyse time-dependent stimulation protocols. This includes step-changes, pulses and waves of stimulus. To challenge HEK293 cells with a time-varying stimulation protocol a custom built perfusion system was used, for more details see [95]. The Ca^{2+} data has been kindly provided from four sources, namely Dr Falcke, Dr Taylor, Dr Skupin and Dr Bellamy. For experimental protocols we refer the reader to [6, 61, 16] .

1.2 Point Process Background

In this section, we give a brief introduction to point processes and Poisson point processes defined on the positive real line $[0, \infty)$. For an in depth review of the theory of point processes we refer the reader to [96, 97].

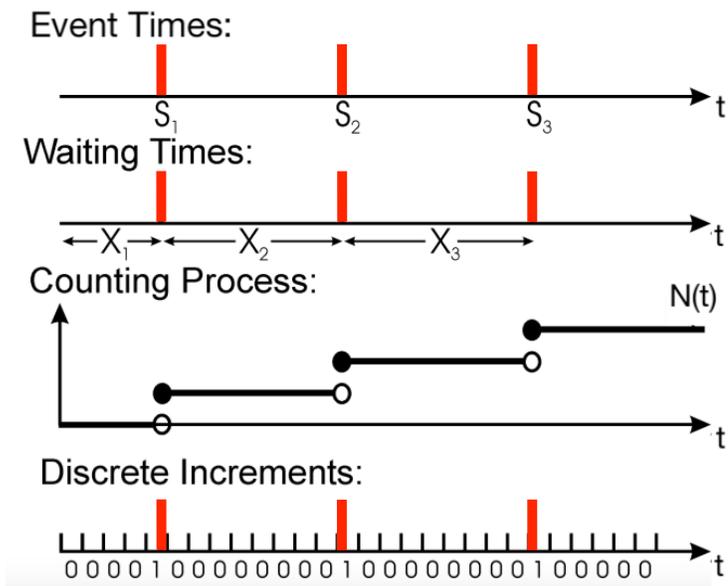


Figure 1.5: Visualisation of the different ways to define a point process. The red lines correspond to event times.

A point process is a stochastic process composed of a time-series of binary events that occur in continuous time [96]. This is illustrated in Figure 1.5, where we have three events. A point process can be defined in four equivalent ways — shown in Figure 1.5 — by the events times, waiting times, a counting process or discrete increments. In particular, define S_1, S_2, \dots to be random variables describing the times that the events of the point process occur. Then a realisation of a point process is $S_1 = s_1, S_2 = s_2, \dots$ where $0 < s_1 < s_2 < \dots$. We can also define the point process by the set of waiting times X_1, X_2, \dots where X_i is the random variable describing the time between the $(i - 1)$ th and i th event time. By definition we have $X_i = S_i - S_{i-1}$ and $S_i = \sum_{k=1}^i X_k$. A third way to define the point process is the counting process $N(t)$ which describes the number of events to have occurred up to and including time t . The final method to define a point process is by discrete increments where we discretise time $\mathbf{t} = \{t_i\}_{i=1}^M$ and we record the number of events to occur in each interval $\Delta N_{(t_i, t_{i+1})} = N(t_{i+1}) - N(t_i)$.

In this thesis, we consider Ca^{2+} spike times as point processes. Therefore, we refer to the event times as spike times and waiting times as inter-spike intervals (ISIs). We shall use this notation for the rest of this section. Furthermore, throughout this thesis we will define the point process models of Ca^{2+} spikes by their ISI distribution.

1.2.1 Poisson Process

We now define two commonly used point processes the Poisson process and the inhomogeneous Poisson process. Due to the various ways to define a point process there are equivalent ways to define the Poisson process, we provide two.

Definition 1. *A Poisson process of rate λ is a point process satisfying:*

a) *For any interval $[t, t + \Delta t]$ the number of spikes in the interval is distributed as a Poisson random variable with rate $\lambda \Delta t$,*

b) *For any non-overlapping intervals the number of spikes in each interval is independent.*

Definition 2. *A Poisson process of rate λ is a point process with independent and identically distributed exponential ISI times with rate λ .*

Therefore, given the last spike occurred at time t_1 the probability density function of the next spike occurring at time t_2 is

$$p(t_1, t_2) = \lambda \exp\{-\lambda(t_2 - t_1)\}. \quad (1.5)$$

The first definition illustrates why the process is named the Poisson process, as the number of spikes in any interval is Poisson distributed. By using the ISI distribution we can easily sample from a Poisson process — by repeated sampling of the exponential ISI distribution.

The inhomogeneous Poisson process is a generalisation of the Poisson process that allows the spiking rate to vary over time. The constant rate parameter λ of the Poisson process is replaced with the rate function $\lambda(t)$ which defines the instantaneous

probability of a spike at each point in time

$$\lambda(t) = \lim_{\Delta t \rightarrow 0} \frac{\text{Pr}(\Delta N_{(t,t+\Delta t)} = 1)}{\Delta t}. \quad (1.6)$$

Definition 3. *The inhomogeneous Poisson process with rate $\lambda(t)$ is a point process satisfying:*

- a) *For any interval $[\tau, \tau + \Delta\tau]$ the number of spikes in the interval is distributed as a Poisson random variable with rate $\int_{\tau}^{\tau+\Delta\tau} \lambda(t)dt$,*
- b) *For any non-overlapping intervals the number of spikes in each interval is independent.*

For the inhomogeneous Poisson process the distribution of the ISIs now depends on the time of the previous spike time. Therefore, given the last spike occurred at time t_1 the probability density function of the next spike occurring at time t_2 is given by

$$p(t_1, t_2) = \lambda(t_2) \exp \left\{ - \int_{t_1}^{t_2} \lambda(s)ds \right\} \quad (1.7)$$

The simple Poisson process is a special case of the inhomogeneous Poisson process where $\lambda(t)$ is constant.

1.2.2 Point processes and time-dependent renewal theory in neuroscience

Although a good starting point, Poisson processes cannot account for the refractoriness of Ca^{2+} spike sequences - a minimum amount of time before another Ca^{2+} spike can occur. One method to account for such properties is to look at renewal processes, which have been influential when modelling spiking neurons, see [98, 99, 100].

In particular, a common model for neurons is the Spike Response Model (SRM) which describes the membrane potential (voltage) of a neuron in relation to the last spike time, and the neuronal response to an input current. A new spike is recorded when the membrane potential hits a threshold voltage. Notably, the discrete-time version of voltage equation in the SRM can be expressed as a scalar product

$$u_t = \mathbf{k} \cdot \mathbf{x}_t$$

where u_t is the membrane potential at time t , \mathbf{k} is the parameter vector — which needs estimating — and \mathbf{x}_t is the “input vector” [98]. Gerstner and Kistler [101] added noise into the SRM by replacing the fixed threshold by a stochastic process. With this change, the instantaneous firing rate $\rho(t)$ is a function of the momentary difference between the membrane potential $u(t)$ and the threshold $\theta(t)$

$$\rho = f(u - \theta) = f(\mathbf{k} \cdot \mathbf{x}_t - \theta).$$

Models of this kind are often referred to as Generalised Linear models (GLM) as the membrane potential can be written as a linear function of the parameters. An example of the functional forms of f is the exponential function [102].

The advantage of the model is that the interspike interval distributions can be expressed analytically for time-dependent input. In particular, given a spike occurs at time s , the next spike will occur at time t with probability

$$P(t|s) = \rho(t) \exp \left[\int_s^t -\rho(u) du \right].$$

In point process theory, the above equation is the interval distribution in terms of the hazard for a time-dependent renewal process [98]. To infer model parameters of GLMs, efficient maximum likelihood estimation (MLE) or maximum a posteriori (MAP) solutions are used, see [98]. These models have been found to accurately describe and predict spiking neurons.

In this thesis, we will endeavour to use time-dependent renewal processes to model the Ca^{2+} spike sequences. However, rather than consider physiological conditions we go straight to point processes. One justification for our approach is that Ca^{2+} spikes occur less often than spikes from neurons, therefore there is less information to infer the underlying mechanics.

1.3 Structure of the Thesis

This thesis is divided into 6 chapters. Chapter 2 outlines our modelling framework and details how to simulate spike sequences from the ISI model and how to sample from

the posterior distribution using MCMC algorithms. Throughout the chapter we use the inhomogeneous Gamma ISI distribution to illustrate the methods contained within it. In Chapter 3, we explain how to create inhomogeneous ISI distributions from any distribution whose support is $(0, \infty)$. We also, describe and mitigate difficulties in sampling from the posterior distribution. In particular, issues arising from: inferring the length scale of the GP, the speed of sampling from the posterior distribution and challenges with sampling from the conditional distribution of the intensity function for the GP prior. In Chapter 4, we extend the model to include the refractory period as a model parameter and discuss the advantages and disadvantages of doing so. In Chapter 5, we apply our modelling framework to HEK293 cells and astrocytes challenged under constant stimulation and HEK293 cells challenged with various time-dependent stimulation protocols. In Chapter 7, we investigate the best method to cluster intensity functions and apply clustering to the intensity functions inferred from HEK293 cells. Chapter 8, describes web applications developed alongside the modelling framework to: extract Ca^{2+} spikes from raw data, simulate surrogate Ca^{2+} spike sequences and generate the features of intensity functions.

CHAPTER 2

Model and Bayesian Inference

In this chapter we describe how Ca^{2+} spikes can be modelled as point processes, and how to create time-heterogeneous models generalising stationary inter-spike interval (ISI) dynamics, by utilising so-called intensity functions. We then explain how to simulate spike sequences from our ISI model. Next we explain how to use Bayesian inference to fit the ISI model to spike sequences, by employing Markov chain Monte Carlo (MCMC). In particular, we give three different prior models for the intensity function and discuss which MCMC algorithms to use for each prior. Throughout this chapter, we provide explicit guidance on how to simulate and fit model parameters for the inhomogeneous Gamma ISI distribution. We conclude with an example of fitting our ISI model using simulated data.

2.1 The model

Before developing a model for Ca^{2+} spiking in cells, we first need to understand the raw Ca^{2+} concentration data. The raw data consists of time-series data for the concentration of Ca^{2+} in individual cells. This is shown in Figure 3.5 where we can

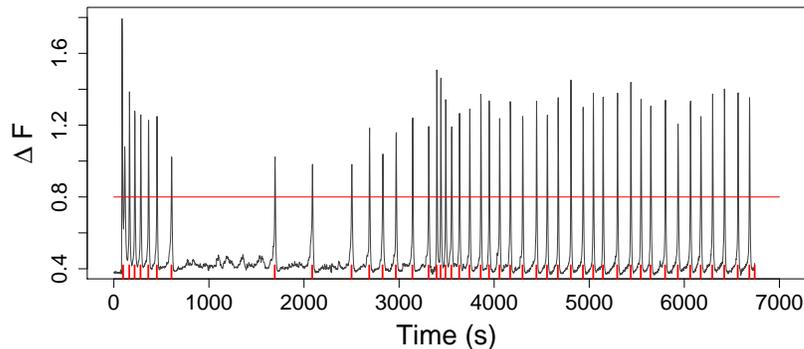


Figure 2.1: Fura-2 fluorescence intensity trace of a HEK293 cell stimulated initially by $20\mu\text{M}$ carbachol and then with $50\mu\text{M}$ carbachol at 3444s. The spike times (red ticks) are obtained by thresholding (red line) the change in fluorescence at 0.8. The data was kindly provided by Dr. Falcke, for experiment details see [6].

see the transient rises and falls in the Ca^{2+} concentration. We extract Ca^{2+} spikes (red ticks on the x-axis) by thresholding the Ca^{2+} concentration (red line in Figure 3.5). This is shown for one cell, however we have Ca^{2+} concentration data for HEK293 cells and astrocytes challenged with a variety of stimulation protocols. Hence our aim is to create a statistical model to mimic the behaviour of Ca^{2+} spikes in cells. Moreover, once we have fitted models to Ca^{2+} spike sequences we will investigate which parameter values are associated with certain cell types and stimulus profiles.

A natural starting point to model 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 distribution. Namely, the probability density $p(s, t)$ of a spike occurring at time t given a spike at time s and 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 and 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 α and β are the shape and rate parameters respectively and Γ denotes the Gamma function. The mean and variance of the Gamma distribution is α/β and α/β^2 , respectively. It is known that directly after a spike in Ca^{2+} concentration the probability of a spike occurring shortly afterwards is low [103]. The downtime after a spike when no spikes occur is known as the refractory period t_{refract} . The Gamma distribution is a suitable choice since increasing both α and β simultaneously, whilst retaining the ratio α/β , 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 by adding the parameter t_{refract} , such that if $t < t_{\text{refract}}$ then $p(s, t) = 0$. Indeed, Skupin et al. [61] included the refractory period when modelling Ca^{2+} spikes with an Exponential distribution. The advantage of this approach is that since we assume ISIs are independent and identically distributed the probability of a Ca^{2+} spike sequence factorises into individual ISIs. Suppose 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 spike 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 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 distribution.

These models are called time-dependent renewal processes [98]. If additionally, $p(y_i, y_{i+1}) = p(y_{i-1} - y_i)$, these models are often referred to as time-homogeneous. If $p_1()$ is chosen as the forward recurrence time it is a stationary renewal process [104].

Note that for a time-homogeneous renewal processes the ISI probability depends only on the time since the last Ca^{2+} spike and not on the absolute experiment time.

For example, suppose the first two Ca^{2+} spikes are separated by 100s, the probability of this occurring is the same whether the first spike occurs 50s or 1000s into the experiment. In Figure 3.5, notice that after the stimulus is exchanged — at 3444s — the rate of Ca^{2+} spikes increases. This demonstrates that the spiking rate depends on the absolute experiment time.

Therefore we need to consider a time-dependent renewal process as a model for Ca^{2+} spike sequences. We choose to follow the practical approach first used by Barbieri et al. [105] for modelling neural spiking activity and later adapted by Tilunaitte et al. [16] for Ca^{2+} oscillations. Their method relies upon combining a known probability model that defines the stochastic nature of the Ca^{2+} spikes sequences with a one-to-one transformation that relates the Ca^{2+} spike times with its history and applied stimulus.

Assume that Ca^{2+} spikes occur at times $\{y_i\}_{i=1}^N$ where $y_i < y_{i+1}$ for all i . Let $f_z(z)$ denote the renewal process probability density function for a general renewal process on $z \in (0, \infty)$. In other words $p(z)dz$ is the probability that a spike occurs in $[z, z + dz]$, and subsequent spikes are independent. Let y represent a time variable on the interval (y_*, ∞) and X be a one-to-one mapping $X(y) = z$ from (y_*, ∞) to $(0, \infty)$ for $y_* \geq 0$. By transformation of variables the probability density of y can be calculated from $f_z(z)$ and X by

$$p(y) = \left| \frac{dX}{dy} \right| f_z(X(y)), \quad (2.3)$$

(see [105]). In other words, the probability density for a Ca^{2+} spike at y_i can be calculated from the renewal probability density function f_z if we know the mapping X . A practical form of X is

$$X(y) = \int_{y_*}^y x(v)dv. \quad (2.4)$$

This form of X satisfies the conditions above and we call $x(t)$ the intensity function, which is a parameter that we will need to estimate. Equation (2.4) can be viewed as rescaling the original time y , such that the time between Ca^{2+} spikes become independent and identically distributed (i.i.d) in the new time [106]. For proof that

the transformation gives rise to i.i.d ISIs we refer the reader to [107]. Therefore, given Ca^{2+} spike times y_i and y_{i-1} in the original time, the ISI in the new time is

$$X(y_{i-1}, y_i) = \int_{y_{i-1}}^{y_i} x(v) dv. \quad (2.5)$$

Note that only through the addition of the intensity function $x(t)$ do the Ca^{2+} ISIs become Markov. Therefore, we include the intensity function explicitly in the ISI probability density function via $p(y_{i-1}, y_i | x)$, where we also include the time of the previous spike y_{i-1} directly replacing y_* .

To illustrate the above construction, we create the inhomogeneous Gamma ISI distribution. We begin with the one-parameter Gamma probability distribution with parameter γ defined by

$$f_z(z) = \frac{\gamma^\gamma}{\Gamma(\gamma)} z^{\gamma-1} e^{-\gamma z}, \quad (2.6)$$

where $z > 0$, $\gamma > 0$ and $\Gamma(\gamma)$ is the gamma function. The mean and variance of this distribution is 1 and $1/\gamma$, respectively. By applying the transformation in equation (2.3) to equation (2.6) and setting $y_* = y_{i-1}$ and $y = y_i$ we get the inhomogeneous Gamma ISI distribution

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

where

$$X(s, t) = \int_s^t x(v) dv, \quad \text{for any } s, t \in [0, T]. \quad (2.8)$$

At this stage, it may appear that the intensity $x(t)$ is devoid of biological meaning, and only a mathematical trick. However, for the Ca^{2+} spiking models contained within the thesis, the intensity function corresponds to the probability of Ca^{2+} spiking independent of the history of the Ca^{2+} spike sequence. See Section 3.1 for further detail. Put differently, if there are N identical Ca^{2+} spiking cells, $Nx(t)$ is the expected number of Ca^{2+} spikes at time t . This is demonstrated for the inhomogeneous Gamma ISI distribution in Section 2.2 — in Figure 2.2(B).

In summary, we model Ca^{2+} spike sequences as point processes that are fully described by their ISI distribution. To construct an inhomogeneous ISI distribution we use a known probability distribution and a time-transformation as described above,

after which the model is defined by the distribution type (Gamma, Weibull, etc) and their parameter θ and the intensity function $x(t)$. In this chapter we have shown the case for the inhomogeneous Gamma distribution, however the concept can be used for any probability distribution on $(0, \infty)$ used to describe the ISI dynamics, see Section 3.1 for more detail.

Notice that the more complex time-heterogeneous model contains the simpler stationary model within it. 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 (2.7) we retrieve the 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 θ and intensity function $x(t)$, our goal is to create spike sequences from the ISI distribution. There are numerous ways to simulate spike sequences, such as time rescaling, an approach using a Bernoulli process and the inverse transform method [107, 99, 108]. For all ISI distributions bar the Exponential distribution we use the inverse transform method. Since the Exponential ISI distribution leads to an inhomogeneous Poisson process, we use a thinning method [96]. 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 from the uniform distribution on $[0, 1]$, $a \sim U[0, 1]$. Then if the cumulative density function (CDF) F of the target distribution is known, the sampled value is $F^{-1}(a)$, the inverse CDF evaluated at a .

To sample spike sequence in $[0, T]$ we iterate this approach for each ISI interval. Namely, for $i > 0$ given the last spike occurred at time y_i the next spike time y_{i+1} is simulated by using the inverse CDF of our ISI distribution. If T occurs before the next spike then y_i is the last spike in the spike sequence. However, the CDF of

our ISI distribution does not have a closed form. Therefore, we discretise time and numerically calculate the CDF. Specifically, we choose K steps to evenly discretise $[0, T]$, obtaining the discretisation $\mathbf{t} = \{t_i\}_{i=0}^K$ where $t_i = iT/K$.

Thus to simulate spike sequences we require the following inputs: length of experiment T , the ISI distribution, ISI parameter θ , intensity function $x(t)$ defined on $[0, T]$ and the number of steps for the discretisation K . Pseudocode for the method is described in Algorithm 1, where C stores the value of the CDF whilst we iterate through times \mathbf{t} and y_{cur} is 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 2.2(A), we have simulated 50 spike sequences from the inhomogeneous Gamma ISI distribution with ISI parameter $\gamma = 10$ and intensity function $x(t) = 2 \cos \frac{t}{2} + \cos \frac{t}{4} + 2.8$. We see that the simulated spike sequences mirror the intensity function, with more spikes at the start of the experiment and a wave of spikes in

Algorithm 1: Simulating spike sequences.

Input: T , K , ISI distribution, θ 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} ;

[10s, 15s]. To visualise the intensity function's biological representation we simulate 1000 spike sequences and bin the spikes into small time regions, creating a peri-stimulus time histogram. Due to the large number of spike sequences used this is comparable to the expected number of spikes, in each time region, irrespective of the spike history. In Figure 2.2(B) we find good agreement between the intensity function (red) and the peri-stimulus time histogram (blue), demonstrating that the intensity function corresponds to the mean spiking rate independent of spike history.

2.3 Bayesian Inference

We have shown that given parameters of an inhomogeneous Gamma ISI distribution we can simulate spike sequences. Now consider the reverse, given Ca^{2+} spike data can we infer the model parameters from which the spikes occurred. In other words, which

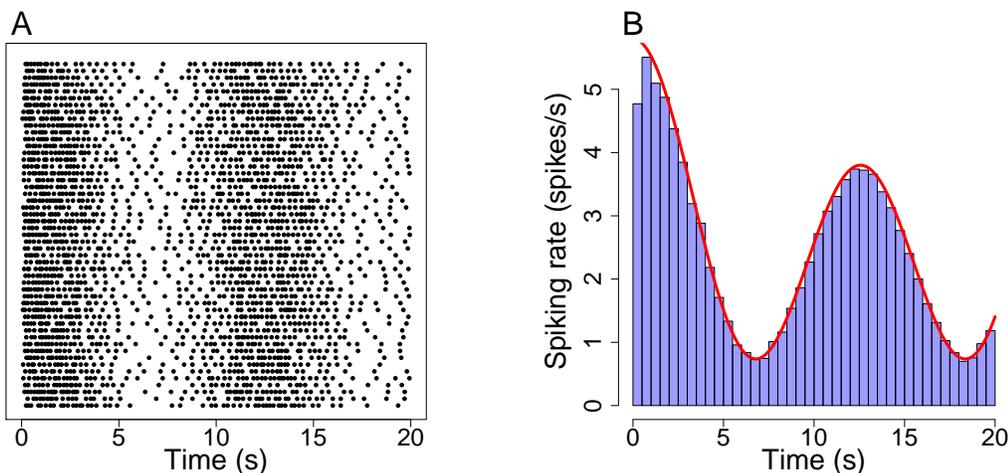


Figure 2.2: (A) Raster plot of simulated spike sequences in $[0s, 20s]$ from an inhomogeneous Gamma ISI distribution with ISI parameter $\gamma = 10$ and intensity function $x(t) = 2 \cos \frac{t}{2} + \cos \frac{t}{4} + 2.8$. (B) Comparison of the intensity function (red) and the histogram (blue) obtained from 1000 spike sequences simulated from the ISI distribution used in (A).

values of model parameters describe the properties of Ca^{2+} spike sequences. Consider a single cell and observe its Ca^{2+} concentration over the time period $[0, T]$. From this we determine the times $0 < y_1 < \dots < y_N < T$ of Ca^{2+} spikes. Let $\mathbf{y} = \{y_i\}_{i=1}^N$ denote the set of all spike times. 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 only get a point estimate for our parameters but a joint distribution of x and θ .

To illustrate the advantage of distributions over point estimates consider the distribution of the ISI parameter θ . If the distribution is strongly centred around some value θ^* then we can be confident that θ^* is a good estimate for θ . However, if the distribution has multiple maxima or large variance, this would imply less confidence in the estimate.

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 the ISI parameter and intensity function.

To calculate the posterior distribution we apply Bayes' theorem

$$\pi(x, \theta | \mathbf{y}) = \frac{\pi(\mathbf{y} | x, \theta) \pi(x, \theta)}{\pi(\mathbf{y})}. \quad (2.9)$$

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 Ca^{2+} spike sequence given the parameters of the model, i.e. the probability of spike sequence \mathbf{y} given $x(t)$ and θ . This is exactly how we defined the model — see (??) and (??). Therefore, the joint probability density of a spike sequence factorises into individual ISIs. Thus, the likelihood is given by

$$\pi(\mathbf{y} | x, \theta) = p_1(y_1 | x) p_T(T, y_N | x) \prod_{i=2}^N p(y_{i-1}, y_i | x, \theta). \quad (2.10)$$

Here, $p_1(y_1 | x)$ represents the conditional probability of finding the first Ca^{2+} spike at time y_1 . This probability describes the forward recurrence time. In other words, the time measured forward to the first Ca^{2+} spike. Furthermore, $p_T(T, y_N | x)$

represents the conditional probability of finding no Ca^{2+} spike in the interval $(y_N, T]$.

We assume p_1 and p_T come from an inhomogeneous Poisson process, i.e.

$$p_1(y_1|x) = x(y_1)e^{-X(0,y_1)}, \quad p_T(T, y_N|x) = e^{-X(y_N,T)}, \quad (2.11)$$

where $X(\cdot, \cdot)$ is defined in (2.8).

For the inhomogeneous Gamma ISI distribution putting together equations (2.7), (2.10) and (2.11) gives the likelihood to be

$$\begin{aligned} \pi(\mathbf{y}|x, \gamma) &= x(y_1)e^{-X(y_0,y_1)}e^{-X(y_N,T)} \\ &\times \prod_{i=2}^N \frac{\gamma x(y_i)}{\Gamma(\gamma)} [\gamma X(y_{i-1}, y_i)]^{\gamma-1} \exp(-\gamma X(y_{i-1}, y_i)). \end{aligned} \quad (2.12)$$

Γ denotes the Gamma function which is defined for all real numbers z except 0 and negative integers by

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt. \quad (2.13)$$

The *prior* contains our beliefs about the parameters before seeing the data. For example, if we strongly believed θ to be close to the value θ^* then we would choose a prior distribution that is centred around θ^* , such as $\pi(\theta) = \Gamma(\kappa\theta^*, \kappa)$ for some value of κ . On the other hand, if we are unsure of the value of θ we would choose a wider prior. Moreover, if we believe there is a positive correlation between say θ_1 and θ_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 Σ has positive correlation.

It is easy to think of priors for the ISI parameter θ as this is a point variable, however it is harder to imagine a prior for the intensity function $x(t)$. 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 would be to assume it is constant, $x(t) = x$. This simplifies the problem greatly as x becomes a point variable. Therefore we only require a distribution on $(0, \infty)$ such as a Gamma or inverse Gaussian distribution. 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, the shape of $x(t)$ is still limited. 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 flexibility for the shape of $x(t)$. We use two such methods, namely applying a Gaussian Processes (GP) — as suggested by Cunningham et al. [109] — and a piecewise constant (PWC) prior. Details of these priors are described in Sections 2.4.3 and 2.4.2, 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 θ . 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, Tilunaite et al. [16] 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. To avoid calculating it we can utilise MCMC algorithms to sample from the posterior distribution instead. When sampling the posterior distribution we only need to know the posterior distribution up to a multiplicative constant. Therefore we can sample from $\pi(\mathbf{y}|x, \theta)\pi(x, \theta)$ and avoid the normalising constant. 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

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

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 [110, 111]), 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.

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

We utilise a Gibbs sampling method over $x(t)$ and ISI parameter θ . 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 θ . 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 θ_0 then iterating the process

Step 1: Update $x(t)$ applying either a constant prior (Section 2.4.1), a piecewise constant

prior (Section 2.4.2) or Gaussian Process prior (Section 2.4.2);

Step 2: Update θ using a Metropolis-Hastings algorithm (Section 2.4.4).

Thus the algorithm returns samples for x and θ , 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_{burn} iterations are removed.

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 against 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 θ . We use two different methods to achieve this, Maximum likelihood estimation (MLE) and MCMC.

We need to decide prior distributions for x and θ , we choose to use independent Gamma distributions for both x and θ , i.e.

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

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

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

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 an optimisation 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 θ 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 (in our case the likelihood and prior). For RW-Metropolis the proposal distribution is Gaussian centred around the current value with some variance, $z' \sim N(z, \sigma^2)$. 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, θ) . Since we update x and θ separately, when updating x θ is constant and 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 σ_x^2 and σ_θ^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 θ by proposing a new value $\theta_{\text{can}} \sim N(\theta_{\text{cur}}, \sigma_\theta^2)$, and it is accepted probability $\min\left(1, \frac{\pi(\theta_{\text{can}}|y, x_{\text{cur}})}{\pi(\theta_{\text{cur}}|y, x_{\text{cur}})}\right)$. Often the calculations include large powers of variables, for example the Gamma ISI distribution (2.12) contains γ^γ , where γ 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, θ_i) the i th sample. Often, the initial M_{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 γ , we have the likelihood given by (2.17). Expanding the priors for x and θ 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 .

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

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

The equation above only holds if $p_{\text{acc}}^x < 1$.

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 could draw from the distribution $\Gamma(\alpha_x + (N-1)\gamma, \beta_x + y_1 + (t - y_N) + \gamma(y_N - y_1))$. However, this is not true for other ISI distributions such as the inverse Gaussian or log-normal.

Similarly the acceptance probability for γ 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 where we fit the Gamma ISI distribution to a simulated Ca^{2+} spike sequence. The simulated Ca^{2+} spike sequence was generated from the Gamma ISI distribution in the region $[0, 20]$ with parameters $x(t) = 3$ and $\gamma = 9$. We apply the MCMC algorithm beginning with $x_0 = 1$ and $\gamma_0 = 1$ and both variances $\sigma_x^2, \sigma_\gamma^2$ also set to 1. We ran the algorithm for 40,000 iterations and

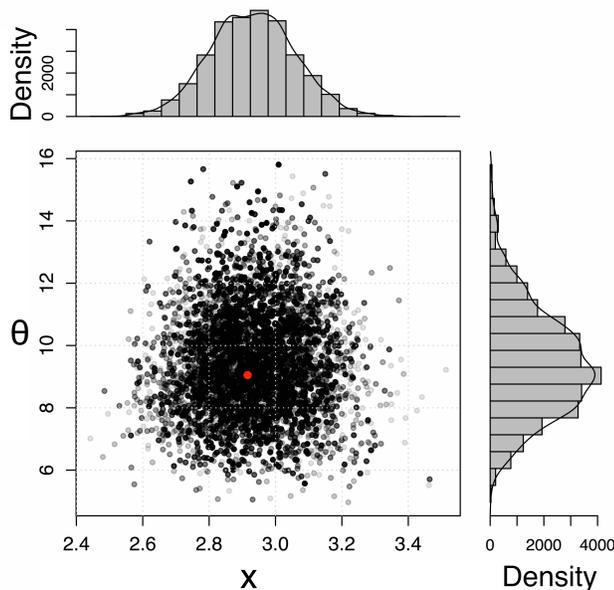


Figure 2.3: Posterior Gamma ISI distribution of the simple model fitted to a single simulated Ca^{2+} spike sequence. Each grey point represents one sample from the posterior distribution — consisting of intensity function x and ISI parameter θ — where overlapping values become darker. The red dot is the value obtained using MLE. The histograms, top and right, represent the individual posterior distributions of x and θ respectively.

removed the first 10,000. 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. 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) algorithm [112], 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 following section.

The advantage of this method compared to the Gaussian Process prior 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 λ 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 λ 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) [112] to probabilistically penalise short intervals and improve the spacing of the change points.

For the heights we have two options for the prior. Firstly, h_0, h_1, \dots, h_k have independent Gamma(κ, μ) distributions. Secondly, the heights can follow a martingale structure (Following Arjas and Gasbarra (1994) [113]) where we assume that $h_0 \sim \text{Gamma}(\kappa, \mu)$ and that given h_0, \dots, h_{i-1} we have $h_i \sim \text{Gamma}(\kappa, \mu_i)$ 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

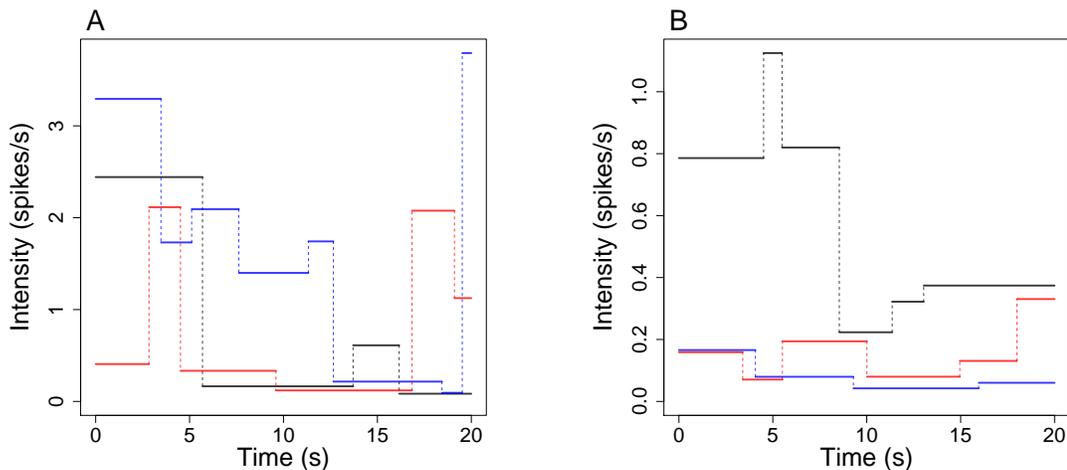


Figure 2.4: Realisations from the PWC prior, where $T = 20$, $k_{\max} = 10$, $\lambda = 5$, $\kappa = 1$ and $\mu = 1$. In (A) the heights are independent and in (B) the heights follow the martingale structure.

then those assumed to be independent. Indeed, we see in Figure 2.4 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 outline 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 2.4.4. 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_{burn} iterations; the parameters of the PWC prior λ , k_{\max} , κ and μ ; the variance σ_{θ}^2 used in the RW-Metropolis algorithm and the prior type for heights (either independent or martingale structure). The algorithm outputs M samples

of the intensity function, where each sample is composed of the change points and corresponding heights. The number of change points and heights can vary in each sample. In addition to M samples of the ISI parameter θ .

We will begin by considering the heights to be independent a priori, and discuss how the martingale structure alters the formulation 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 λ conditioned with maximum value k_{\max} and c is chosen such that $\max_k(b_k + d_k) = 0.9$ [112]. This is done to make sure within model changes occur at a frequent rate, such that each iteration is not always a change in parameter space.

For all of the updates we need to calculate the likelihood ratio. Suppose we initially have intensity function $x(t)$ and the candidate intensity function is $x'(t)$ then the likelihood ratio is

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

Note that θ takes the same value in both numerator and denominator, thus any terms purely in θ cancel. In particular for the inhomogeneous Gamma ISI distribution the likelihood function $p(\mathbf{y}|x, \theta)$ is given in (2.12), where the ISI parameter is γ . 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.

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 θ .

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**

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 $\mathbf{h}^{(i+1)}$;

Use RW-Metropolis algorithm to update θ (Section 2.4.4);

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

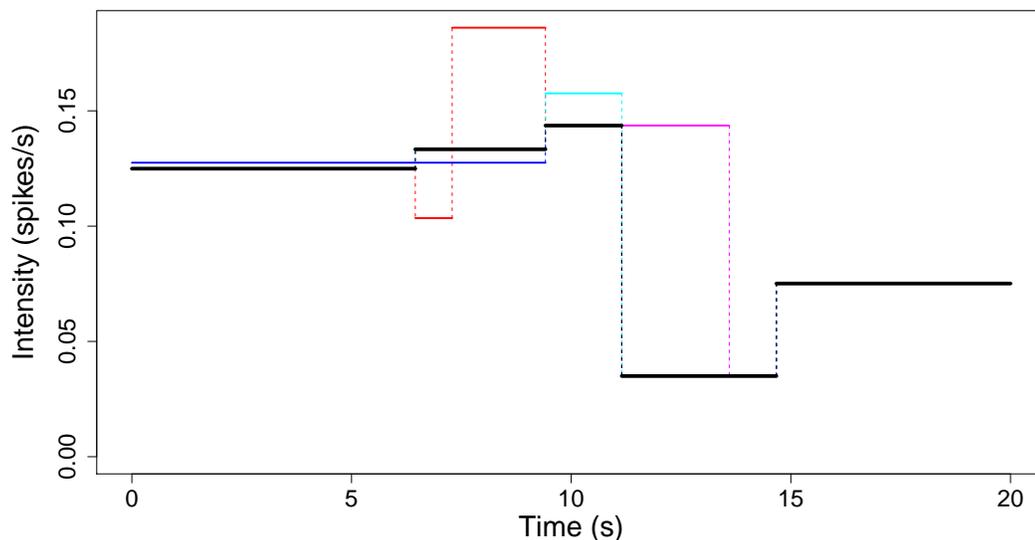


Figure 2.5: Illustration of the possible moves at each iteration of the RJMCMC. The black line corresponds to the original step function. The blue, red, cyan and magenta lines correspond to candidate intensity functions when: a change point is removed, a new change point is added, a height is changed and a change point moves, respectively.

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 $\text{Gamma}(\kappa, \mu)$ 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

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

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

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

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 change point s^* . Thus giving

$$\begin{aligned} (\text{prior ratio for change points}) &= \frac{\left(p(k+1) \frac{(2k+3)!}{T^{2k+3}} (s^* - s_j)(s_{j+1} - s^*) \right)}{\left(p(k) \frac{(2k+1)!}{T^{2k+1}} (s_{j+1} - s_j) \right)}, \quad (2.42) \\ &= \frac{p(k+1)}{p(k)} \frac{(2k+2)(2k+3)}{T^2} \frac{(s^* - s_j)(s_{j+1} - s^*)}{(s_{j+1} - s_j)}. \end{aligned} \quad (2.43)$$

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 $q(u)$ probability density function for the random variable u . 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.

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

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

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

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

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

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

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

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

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

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

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

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

2.4.2.6 Martingale structure for the heights

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 $h_i \sim \text{Gamma}(\kappa, \frac{\kappa}{h_{i-1}})$, for $i > 0$ and $h_0 \sim \text{Gamma}(\kappa, \mu)$. 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 lets consider the simpler case when $j = k$, explicitly writing the prior ratio gives the following

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

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 $h_0 \sim \text{Gamma}(\kappa, \mu)$.

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

Simplifying

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

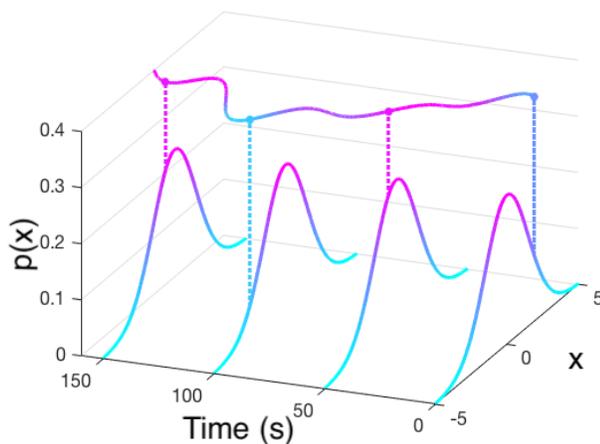


Figure 2.6: Realisation of a Gaussian Process (top line), with mean 0 and SE covariance. The bell shape curves show the probability distributions of the values the realisation can take at times 0, 50, 100 and 150, where the probability is coloured such that light blue indicates low probability and magenta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

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 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 [114]. 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 a GP with SE covariance are smooth when $\sigma_v^2 = 0$. This is a property we desire for the intensity function, to mimic the varying stimulus that cells experience in vitro.

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 a GP with mean 0 and SE covariance function, each curve underneath corresponds to the normal distribution that the point comes from, where magenta corresponds to a higher probability of the value occurring and blue a lower probability.

With the SE covariance, three hyper parameters define the shape of the intensity function. This is illustrated in Figure 2.7. σ_f^2 determines the function's average distance from the mean (Figure 2.7(A,C)); l controls the 'smoothness' of the intensity, where smaller l leads to an intensity that varies more (Figure 2.7(A,B)) and σ_v^2 is additional Gaussian noise (Figure 2.7(A,D)).

To sample from a GP we discretise time by partitioning it into $\mathbf{t} = \{t_i\}_{i=1}^R$, where $t_i = i\Delta$ such that Δ is the time step used. From these points we can retrieve the covariance matrix $\Sigma = \{\Sigma(t_i, t_j)\}_{t_i, t_j \in \mathbf{t}}$, where Σ is defined in (2.68). Assuming the GP has mean $\boldsymbol{\mu}$ — a vector of length R — we can approximate a realisation from the GP by sampling from the multivariate normal with mean $\boldsymbol{\mu}$ and covariance matrix Σ .

Furthermore, we can calculate the probability of any function $f(t)$ defined on $[0, T]$ by discretising to $\mathbf{f} = \{f_i\}_{i=1}^R$ with $f_i = f(t_i)$ and evaluating the probability

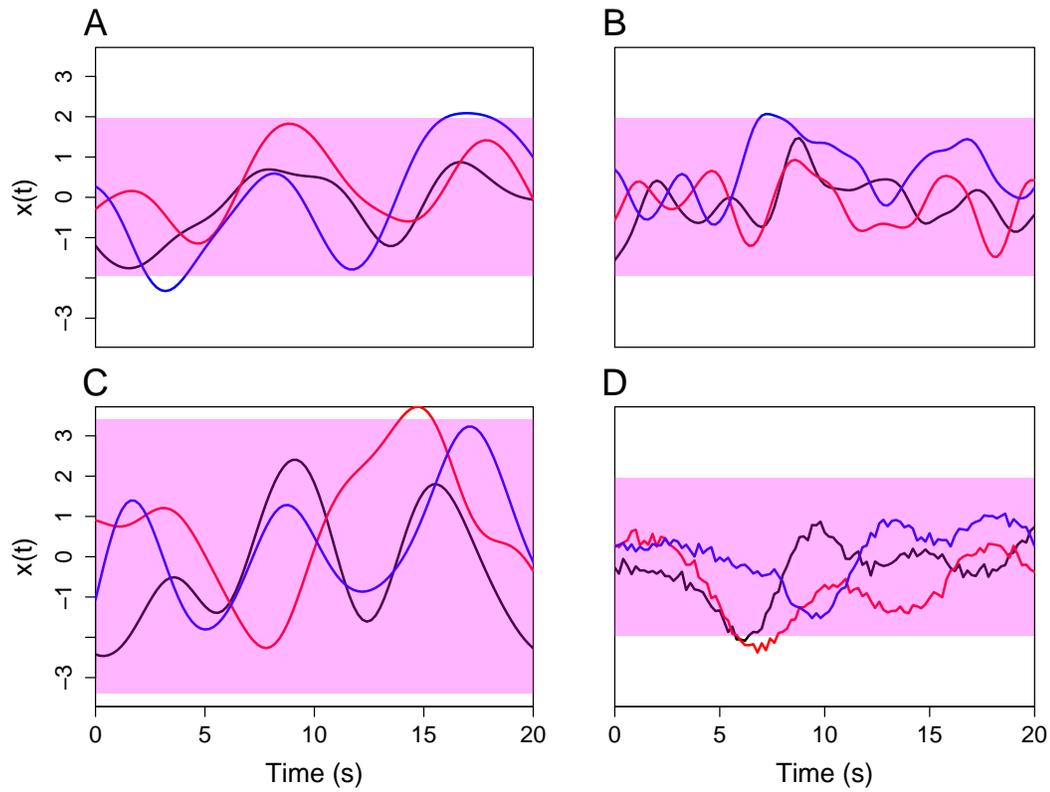


Figure 2.7: Illustration of how varying hyper parameters changes realisations of a Gaussian Process with mean 0. We draw 3 samples from the following GP priors: (A) $\sigma_f^2 = 1, l = 2, \sigma_v^2 = 0$ (B) $\sigma_f^2 = 1, l = 1, \sigma_v^2 = 0$ (C) $\sigma_f^2 = 2, l = 2, \sigma_v^2 = 0$ (D) $\sigma_f^2 = 1, l = 2, \sigma_v^2 = 0.01$. 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 σ_f^2 increases the range of values taken (variance). Finally, (D) shows noise is controlled by σ_v^2 .

density function for the multivariate normal distribution, i.e $\mathcal{N}(\mathbf{f}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

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 $\mathbf{x} = \{x_i\}_{i=1}^R$, where $x_i = x(t_i)$. Often the time step Δ 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 time 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 $\mathbf{t} = \{t_i\}_{i=1}^R$, we extend to $\tilde{\mathbf{t}} = \mathbf{t} \cup \mathbf{y}$, where $\mathbf{y} = \{y_i\}_{i=1}^N$ are the spike times. We define the time indices of the spike times by $\boldsymbol{\iota} = \{\iota_i\}_{i=1}^N$, where ι_i corresponds to the time index of the i th spike time.

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

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

where $\hat{X}_{i,j} = \Delta \sum_{k=\iota_i}^{\iota_j} x_k$ with $\iota_0 = 0$, and $\iota_{N+1} = T/\Delta$. Since we are updating x we can remove terms purely in γ , to get

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

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

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

where $\boldsymbol{\Sigma}$ is defined in (2.68).

It is not necessary to take the logarithm, however if we sampled the posterior distribution of $x(t)$ without taking the logarithm 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 $\{x, \theta, \sigma_f^2, l, \sigma_n^2\}$.

We choose to fix the signal variance σ_f^2 to improve the mixing of the Markov Chain. For methods to fit the signal variance we refer the reader to [114]. It is vital to

choose a suitable value since if σ_f^2 is too small the intensity function will not be able to capture the underlying spiking rate. However, a signal variance that is too large will result in a Markov chain that mixes poorly. To visualise the effect of the signal variance we consider the one-dimensional normal prior distribution. If the variance parameter for the normal prior is small this corresponds to a narrow prior distribution (centered about the mean of the normal distribution). Conversely, choosing a very large variance leads to a vague prior distribution which can place too much weight on values that are outside a plausible range. In both this one-dimensional example and the GP prior we need to choose the variance to be suitably vague. We elect to fix $\sigma_f^2 = 1000$ which we found to be suitably vague without affecting the mixing of the Markov chain.

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 [115]. 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 estimate.

Therefore, with the GP prior our model consists of the intensity function $x(t)$, ISI parameter θ and length scale l . To sample from the posterior distribution we utilise a Gibbs sampler, where we sample each variable one at a time whilst keeping the remaining model parameters fixed. For the length scale and ISI parameter we use a RW-Metropolis algorithm. For the intensity function we use an under-relaxed algorithm. The intensity function requires a more complex MCMC algorithm as in each iteration we need to create a new candidate function.

In Algorithm 3 we outline the method used, where the required inputs are: the Ca^{2+} spikes \mathbf{y} ; the ISI distribution used (e.g. Gamma, Weibull, etc); the experiment length T ; the number of samples required M ; the initial number of iterations to

Algorithm 3: Inference for GP prior.

Input: \mathbf{y} , ISI distribution, T , M , M_{burn} , \mathbf{t} , ω , σ_θ^2 , σ_l^2

Output: M samples of intensity \mathbf{x} , ISI parameter θ 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}_{can} and calculate acceptance probability using under-relaxed algorithm, update $\mathbf{x}^{(i)}$;

Step 2. Propose l_{can} and calculate acceptance probability using a RW-Metropolis algorithm, update $l^{(i)}$;

Step 3. Update θ_{can} and calculate acceptance probability using a RW-Metropolis algorithm, update $\theta^{(i)}$ (Section 2.4.4);

Return: \mathbf{x} , θ and l with initial M_{burn} iterations removed;

remove M_{burn} ; the discretisation \mathbf{t} used for the GP; step-size parameter ω used in the under-relaxed method and the variances σ_θ^2 , σ_l^2 used in the RW-Metropolis algorithms for θ 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}

In this section we describe the algorithm used to propose candidate intensity functions and probability of accepting the new intensity function. We use an under-relaxed algorithm, based on a Metropolis-Hastings method [116]. 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 we set $\mathbf{x}_0 = \mathbf{0.58}$, the vector of length R with all values equal 0.58. Consider an iteration, with current intensity function \mathbf{x}_{cur} , we propose a new intensity function \mathbf{x}_{can} by $\mathbf{x}_{\text{can}} = e^{\mathbf{x}_{\text{can}}^*}$ where

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

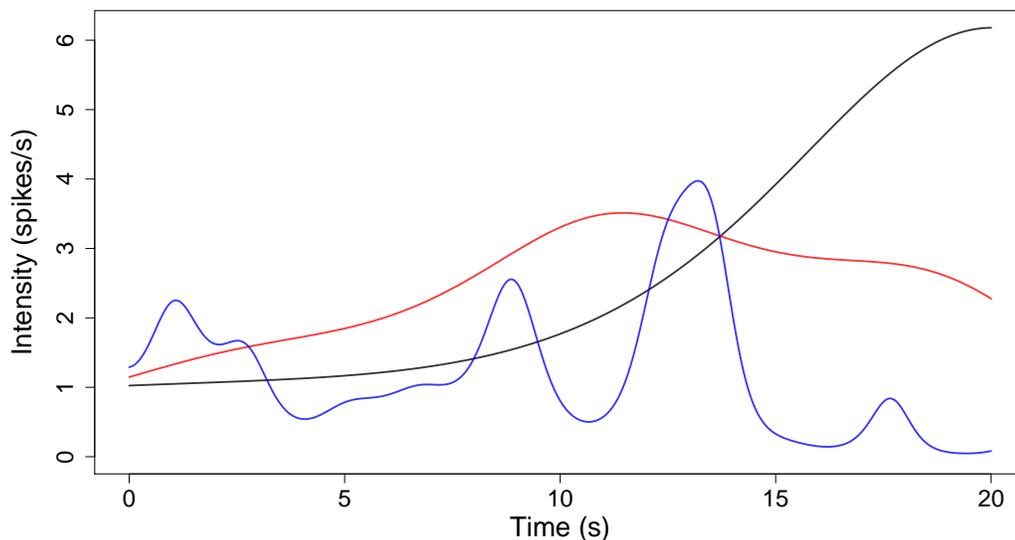


Figure 2.8: Samples from the intensity function prior distribution, where its logarithm has a GP prior. To illustrate the affect of the length scale on the prior distribution the black, red and blue lines correspond to length scales 10, 5 and 1, respectively.

$\omega \in [-1, 1]$ is a step-size parameter. The nearer ω is to zero the closer \mathbf{x}_{can} resembles \mathbf{x}_{cur} whereas, if ω is close to one \mathbf{x}_{can} 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 — candidate intensity functions 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 function s to intensity function r . Due to the proposal regime this reduces to the likelihood ratio because 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. $q(x_{\text{can}} | x_{\text{cur}})$ is equal in distribution to $N(\sqrt{1 - \omega^2} \log(x_{\text{cur}}), \omega^2 \sigma^2)$. In this case the prior on $\log(x)$ is a gaussian centred at 0 with variance σ^2 . Substituting this into (2.73) gives

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

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}_{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 the candidate intensity function if

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

where we consider the discretised intensity functions. For clarity in the equation we use x' and \hat{X}' to correspond to the candidate intensity function and integrated

candidate intensity function and x and \hat{X} the current intensity function and integrated current intensity function.

The base MCMC algorithm for the intensity function can incur mixing issues. These are described and resolved in Section 3.2.

2.4.3.2 Step 2. Estimating the length scale

In this section we describe the method to sample the length scale from the posterior distribution. We begin by applying Bayes' theorem where we consider only the intensity function and length scale of the GP prior

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

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

We do not include the ISI parameter because it is fixed when updating the length scale and does not affect the conditional distribution of the length scale. Equation (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 Gamma prior, $l \sim \text{Gamma}(1, 0.01)$, unless specified otherwise.

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

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

where $\mathbf{x}_* = \log \mathbf{x}$. The term $\frac{1}{\prod_{i=1}^R x_i}$ is the Jacobian factor of the transformation. Recall that we discretised the intensity function into R steps. Hence, combining with

the prior we have

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

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 σ_l^2 , i.e. $l_{can} \sim N(l_{cur}, \sigma_l^2)$. Recall for this algorithm the transition probabilities cancel out — see (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 $u \sim U[0, 1]$.

Inferring the length scale of the GP prior is not always viable due to the limited information contained within a single Ca^{2+} spike sequence. This is investigated further in Section 3.3.

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 θ to be fixed. Now we consider how to sample the ISI parameter θ , where we fix the value of $x(t)$ whilst updating θ . Note that the length scale of the GP only affects the prior of the intensity function we do not consider it when updating the ISI parameter. 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 θ are independent and we only require the distribution up to a multiplicative constant giving

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

We require a prior distribution on θ . 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 θ 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 inhomogeneous Gamma ISI distribution where we have $\theta = \gamma$, and we use the generalised prior $\gamma \sim \text{Gamma}(\alpha, \beta)$

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

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

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

For the Gamma ISI distribution the terms in red were removed, as they do not depend on γ . For computational ease we will work on the logarithmic scale, as there are several exponents in (2.88). 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 θ we use a RW-Metropolis algorithm. We begin with initial value θ_0 , which depends on the ISI distribution. For the inhomogeneous Gamma ISI distribution the initial value is taken to be 1. Suppose the current value is θ_{cur} , in the next iteration we propose a candidate value θ_{can} drawn from $N(\theta_{\text{cur}}, \sigma_\theta^2)$, Where σ_θ^2 is chosen ad hoc. Since we follow a RW-Metropolis algorithm, the transition probabilities cancel and the acceptance probability reduces to

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

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

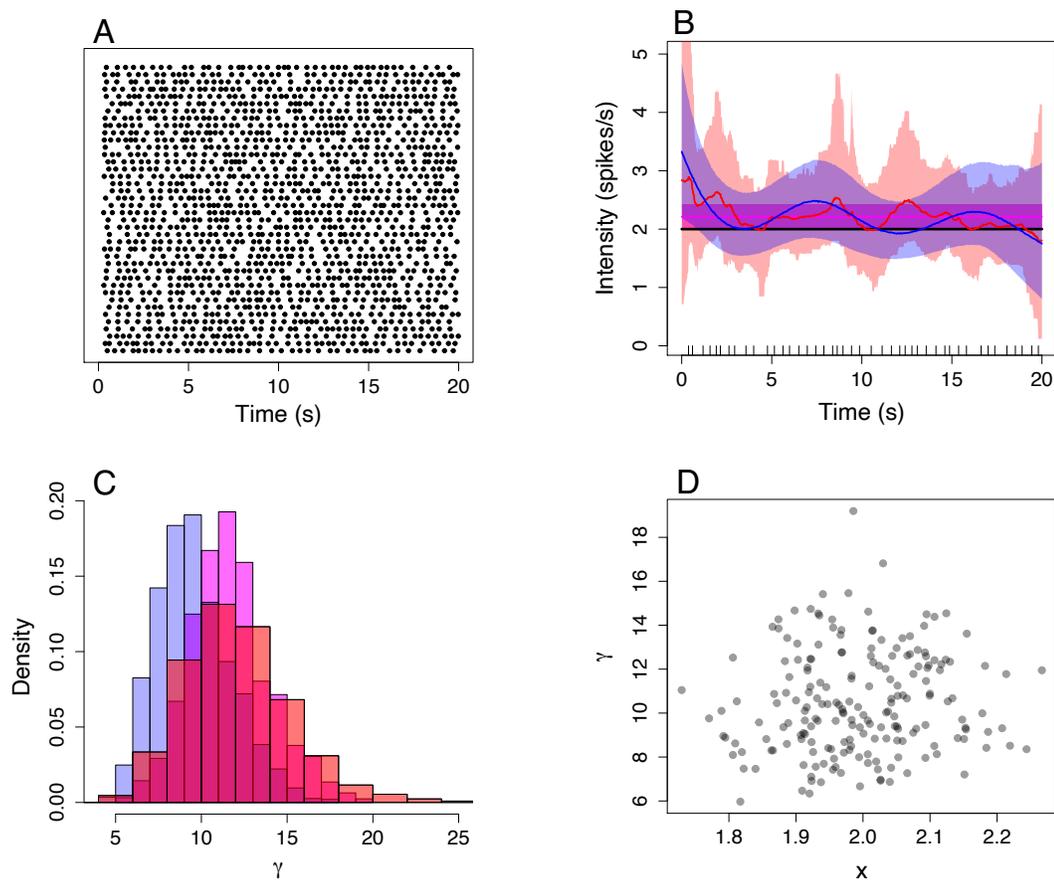


Figure 2.9: Illustration of fitting an inhomogeneous Gamma ISI distribution to spike sequences simulated from a model with a constant intensity function. (A) Raster plot of 40 simulated spike sequences. (B) Mean and 95% credible region of the posterior intensity function distribution inferred from a single spike sequence. The black line represents the intensity function used to simulate spike sequences. (C) Histogram of the ISI parameter γ for the spike sequence and PWC priors shown in (B). In (B,C) The constant, PWC and GP priors for the intensity function are represented by the colours magenta, red and blue, respectively. (D) Scatterplot of the MLEs for 200 spike sequences, where each dot corresponds to the MLE of a single spike sequence.

2.5 Example with simulated data

In this section we demonstrate how to fit the inhomogeneous Gamma ISI distribution to simulated Ca^{2+} data. To simulate Ca^{2+} spike sequences we use the approach outlined in Section 2.2. We choose to generate spikes sequences over 20s from an inhomogeneous Gamma ISI distribution with ISI parameter $\gamma = 10$ and constant intensity function $x(t) = 2$, where $K = 8000$ steps were used. The expected number of Ca^{2+} spikes in one sequence is 40. The simulated Ca^{2+} spike sequences are shown in Figure 2.9(A).

We fit our inhomogeneous Gamma ISI distribution to individual simulated spike sequences. We do this by sampling from the posterior distribution using MCMC algorithms, as outlined in Section 2.4. Recall that we have three choices for the prior of the intensity function: constant, PWC and GP. We infer model parameters for each choice of prior. In all instances the ISI parameter γ has a $\text{Gamma}(1, 0.01)$ prior distribution and RW-Metropolis variance $\sigma_\theta^2 = 1$. We compute 20,000 iterations after an initial burn-in period of 10,000 iterations. When the intensity function is constant a priori we give it a $\text{Gamma}(1, 0.01)$ prior distribution, where the RW-Metropolis variance is $\sigma_x^2 = 1$. For the PWC prior the parameters are: maximum number of change points $k_{\max} = 30$, mean number of change points $\lambda = 20$, height parameters $\kappa = 1$ and $\mu = 2$ where the heights have a martingale prior. For the GP prior the parameters are: the discretisation $\mathbf{t} = \{0, 0.05, 0.1, \dots, 20\}$, step-size parameter $\omega = 0.01$ and length scale variance $\sigma_l^2 = 0.5$.

In Figure 2.9(B,C) we illustrate the posterior ISI distribution fitted to a single simulated Ca^{2+} spike sequence shown by the black ticks in (B). In both the constant, PWC and GP priors are represented by the colours magenta, red and blue, respectively. The 95% credible region and mean of the posterior intensity function distribution is shown in Figure 2.9(B), where the intensity function used to simulate the spike sequence is shown in black. In Figure 2.9(C) we present histograms of the posterior ISI parameter γ . As expected, we find that the posterior distributions of x and θ are close to the model parameters the spike sequence was simulated

from — $x = 2$ and $\theta = 10$. Note that the mean of the intensity function for each prior is higher (~ 2.2) than the intensity function the spikes were generated from — $x(t) = 2$ shown by the black line). This is due to the simulated spike sequence having 44 spikes. The MLE of the simple model fitted to the spike sequence was $x = 2.214$ and $\theta = 11.76$. Even though the PWC and GP prior allows for large changes in the intensity function, we see that both remain approximately at 2.2, with slight fluctuations due to variation in the ISIs. In Figure 2.9(C) we find that the histograms of the posterior γ distribution all are centered around the true value of $\gamma = 10$. Therefore, we have shown that our posterior ISI distribution closely follows the model used to simulate Ca^{2+} spike sequences. To consolidate that we recover the model parameters we calculate the MLE for 200 spike sequences. In Figure 2.9(D) we plot the MLEs for each individual spike sequence. We observe that the dots are scattered around the true model parameters. The variance of the dots around the model parameters we simulated from is due to the random nature of the simulated Ca^{2+} spike sequences.

In the above example we used a constant intensity function. However, we have seen that Ca^{2+} spike times are dependent on the absolute time. Therefore consider a time-dependent intensity function, we choose $x(t) = 2 \cos(t/2) + 1.1$. We simulate 40 spike sequences using the new intensity function — where we keep $\gamma = 10$. This is shown in Figure 2.10(A). We see that the simulated spike sequences are time dependent, where more spikes occur in the regions [0s, 4s] and [9s, 17s].

To each simulated spike sequence we fit the inhomogeneous Gamma ISI distribution using a constant, PWC and GP prior. The parameters used to fit each model are identical to the example using a constant intensity function, described previously.

In Figure 2.10(B,C) we illustrate the posterior intensity function distribution using two different spike sequences, shown by black ticks in both. The intensity function used to simulate spike sequences is shown by the black line. The constant, PWC and GP priors correspond to magenta, red and blue, respectively. We see that the posterior mean intensity function with a PWC and GP prior both closely

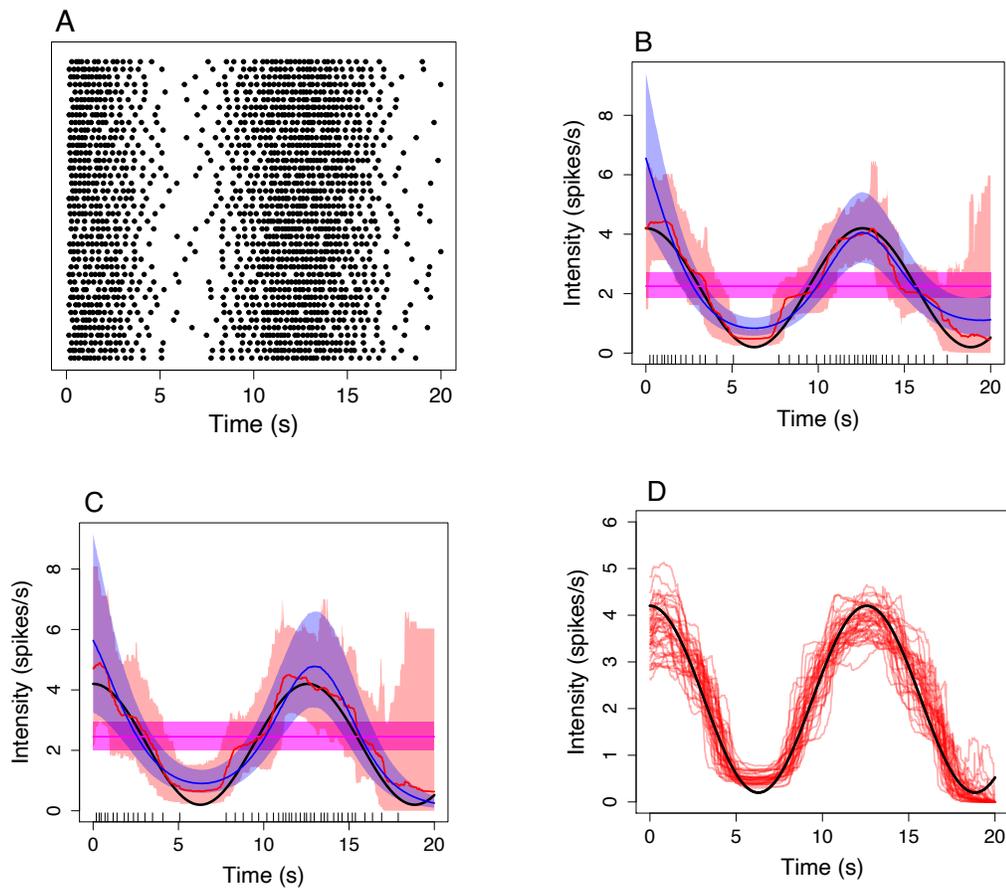


Figure 2.10: Illustration of fitting an inhomogeneous Gamma ISI distribution to spike sequences simulated from a model with a time-varying intensity function. (A) Raster plot of 40 of the simulated spike sequences. (B,C) Mean and 95% credible region of the posterior intensity function distribution inferred from a single spike sequence, shown as black ticks in both. The black line represents the intensity function used to simulate spike sequences. In addition red, blue and magenta lines correspond to posterior intensity computed via PWC, GP and simple priors respectively. (D) Comparison of the intensity function used to simulate spike sequence (black) and the posterior mean of 40 intensity function distributions where each is fitted to a single simulated spike sequences with a PWC prior.

resemble the underlying intensity function. This shows that the flexibility in these intensity priors can capture the time-dependence of the spike sequences. However, the intensity function with a constant prior — equivalent to a stationary ISI model — cannot capture the underlying intensity function. This illustrates the strength of our inhomogeneous ISI model, as it can express changes in the spiking rate over time. This provides a method to capture underlying properties in Ca^{2+} oscillations.

To check that we can recover the intensity function used to simulate Ca^{2+} spike sequences we plot the mean of the posterior intensity function distribution using the PWC prior fitted to 40 individual spike sequences. This is shown in Figure 2.10(D), where the black line is the intensity function used to simulate spike sequences and the red lines the mean of the posterior intensity function distributions. We see that all posterior intensity functions closely follow the underlying rate. This shows that we can accurately recover the time-dependent intensity function used to simulate spike sequences.

2.6 Summary

In summary, in this chapter we have illustrated how to statistically model Ca^{2+} spike sequences as point processes. Our model allows for time-heterogeneity in the Ca^{2+} spikes by employing so-called intensity functions which captures the average Ca^{2+} spiking rate irrespective of the spiking history. We have also shown how to simulate spike sequences given model parameters.

We used a Bayesian framework to fit model parameters to Ca^{2+} spike sequences. Under the Bayesian framework we required prior distributions for the model parameters. To maximise flexibility in the intensity function we used a non-parametric approach, whereby we use PWC and GP priors for the intensity function. To calculate the posterior distribution we utilise MCMC algorithms to do exact inference, rather than approximate the posterior distribution as Tilunaite et al. [16] did. We applied a Gibbs sampling algorithm to obtain samples from the posterior distribution. This relies on sampling each model parameter individually, whilst keeping the remaining

parameters fixed. Therefore, for each model parameter — and intensity function prior — we need to define an MCMC algorithm. For each univariate model parameter — intensity function with a constant prior, length scale of the GP and ISI parameter — we used a RW-Metropolis algorithm. For the intensity function with a GP prior we employed an under-relaxed algorithm and with a PWC prior we used a RJMCMC algorithm.

Finally, we provided two examples of fitting model parameters to simulated Ca^{2+} spike sequences. In the first example, we fitted the ISI model to stationary spike sequences. We found that the posterior intensity function distribution for all three priors were similar in distribution. Moreover, we calculated the MLEs of the ISI model for 200 individual spike sequences. We found that the MLEs are centered about the model parameters used to simulate the spike sequences. Thus, illustrating that we recover the underlying model parameters. In the second example, we simulate from an ISI distribution when the intensity function depends on time. We found that with the PWC and GP priors for the intensity function the posterior distribution was able to capture the time-varying nature of the underlying intensity function, whereas the constant prior cannot. This illustrates the advantage of the PWC and GP priors for the intensity function over the constant prior.

CHAPTER 3

Non-trivial Model Properties

In Chapter 2 we gave an insight into our ISI model and how to compute the posterior distribution given Ca^{2+} spike sequences for a selection of prior beliefs for the intensity function. In this chapter we take a closer look at the details behind the model and how to sample from the posterior distribution, addressing problems that arise within our methodology. This chapter is split into four distinct sections, each dealing with a single issue. In Section 3.1 we extend the choice of ISI distributions from the inhomogeneous Gamma distribution to include the Exponential, inverse Gaussian, log-normal and Weibull distributions. We also confront challenges with the GP prior. In particular, we discuss mixing issues of the intensity function and difficulties in inferring the length scale in Sections 3.2 and 3.3, respectively. In Section 3.4 we consider the computational demand of our MCMC algorithms and propose methods to reduce the cost.

3.1 ISI Distribution

In this section we discuss the importance of the choice of the inter-spike interval (ISI) distribution. In Tilunaite et al.'s original work [16], they considered three ISI distributions, derived from the Exponential, Gamma and inverse Gaussian probability distributions. We explain, in general, how to transition from stationary to non-stationary ISI distributions, and extend the ISIs considered to include the log-normal and Weibull distributions. Furthermore, our generalisation of the inverse Gaussian distribution differs from the one proposed by Tilunaite et al. This is to maintain the meaning on the intensity function across all ISI distributions.

In Chapter 2 we gave an example whereby we created a time-dependent ISI distribution by extending the one dimensional $\text{Gamma}(\gamma, \gamma)$ distribution. The extension is performed by applying a time-rescaling method via the intensity function $x(t)$ which accounts for the time-dependence of Ca^{2+} spiking. This approach can be used for a wide variety of probability distributions, such as an inverse Gaussian or Weibull probability distributions. We will demonstrate that the choice of the original distribution is crucial to how well the model fits the Ca^{2+} data. Furthermore, we will explain how to create a time-dependent ISI distribution from any probability distribution on $(0, \infty)$. Care is required in constructing these inhomogeneous ISI distributions to avoid non-identifiability issues. We will show how to circumvent this problem whilst also providing biological meaning to the intensity function.

To begin, let us first consider the case of stationary Ca^{2+} spikes — spikes that depend only on the time since the last spike and not the time of the experiment. The ISI distribution can be modelled as a function of the time t since the last spike s . We can choose to model the ISI distribution $\psi(t, s)$ as a continuous probability distribution on $(0, \infty)$ such as: the Exponential distribution

$$\psi(t, s|\alpha) = \alpha e^{-\alpha(t-s)} \quad \alpha > 0,$$

which has mean $1/\alpha$ and variance $1/\alpha^2$; the Gamma distribution

$$\psi(t, s|\alpha, \beta) = \frac{\beta^\alpha (t-s)^{\alpha-1} e^{-\beta(t-s)}}{\Gamma(\alpha)} \quad \alpha, \beta > 0,$$

which has mean α/β and variance α/β^2 ; the inverse Gaussian distribution

$$\psi(t, s|\lambda, \mu) = \sqrt{\frac{\lambda}{2\pi(t-s)^3}} \exp\left\{-\frac{\lambda((t-s) - \mu)^2}{2\mu^2(t-s)}\right\} \quad \lambda, \mu > 0,$$

which has mean μ and variance μ^3/λ ; the log-normal normal

$$\psi(t, s|\mu, \sigma) = \frac{1}{(t-s)\sigma\sqrt{2\pi}} \exp\left\{-\frac{(\log(t-s) - \mu)^2}{2\sigma^2}\right\} \quad \mu \in \mathbb{R}, \sigma > 0,$$

which has mean $\exp(\mu + \sigma^2/2)$ and variance $[\exp(\sigma^2) - 1] \exp(2\mu + \sigma^2)$; or the Weibull distribution

$$\psi(t, s|\lambda, k) = \frac{k}{\lambda} \left(\frac{t-s}{\lambda}\right)^{k-1} e^{-((t-s)/\lambda)^k} \quad \lambda, k > 0,$$

which has mean $\lambda\Gamma(1 + 1/k)$ and variance $\lambda^2 [\Gamma(1 + 2/k) - (\Gamma(1 + 1/k))^2]$, where Γ denotes the Gamma function [117]. The ISI parameters for each distribution are α , (α, β) , (λ, μ) , (μ, σ) and (λ, k) , respectively. For a general distribution we denote the ISI parameters by θ . Note that since we are considering stationary spikes, the distributions do not depend explicitly on the timing of spikes but the time since the last spike. This means the times t and s only appear as $t - s$ and we could rewrite $\psi(t, s|\theta) = \psi(t - s|\theta) = \psi(\tau|\theta)$, where τ denotes the time since the last spike.

We consider the Exponential distribution because it is the simplest distribution with support in $(0, \infty)$. Moreover, it describes a process which is memoryless, i.e. a process that does not depend on its history. For example, the probability of the next spike occurring after 1s is identical to the probability of the next spike occurring after 2s given we know there is not a spike in $[0s, 1s]$. The Gamma distribution offers a natural extension to the Exponential distribution, such that the Exponential distribution is a special case of the Gamma — where $\alpha = 1$. The Gamma distribution is an appealing candidate for the ISI distribution because it is a probability distribution for a combination of events occurring for the first time. Thus each event could be interpreted as a Ca^{2+} puff, and each spike occurs the first time that α puffs occur.

The Weibull distribution can also be viewed as an extension to the Exponential distribution — the special case $k = 1$ gives rise to the Exponential distribution. The Weibull distribution is often used to describe the “time to failure” of some component where the failure rate is proportional to a power of time. Thus, this distribution is justifiable if we believe Ca^{2+} spikes depend on a power of time, i.e. if $k > 1$ a spike is more likely to occur as time goes on. We use the inverse Gaussian distribution because it has been used to model first passage events which contains positive drift and random diffusion only. Thus, since conceptually Ca^{2+} spikes have been described as first passage events, the inverse Gaussian is a suitable distribution to test. The log-normal distribution is often used to model the firing rate across a population of neurons. Although firing neurons differ from Ca^{2+} oscillations both describe a spiking process. Therefore, the last ISI distribution model we consider is the log-normal distribution.

To find which of the models best describes a Ca^{2+} spike sequence we first convert the spike sequence into a sequence of K ISI times $\{T_i\}_{i=1}^K$. Then to fit the model $\psi(\tau|\theta)$ with ISI parameter(s) θ , we calculate the maximum likelihood estimate (MLE) $\hat{\theta}$ by

$$\hat{\theta} = \arg \max_{\theta \in \Theta} \left\{ \prod_{i=1}^K \psi(T_i|\theta) \right\},$$

where Θ is the space of all possible parameter values.

To visualise the difference between the five stationary ISI distribution models we fit each model to the same spike sequence generated by a HEK293 cell challenged with $10\mu\text{M}$ carbachol. The cell exhibited 66 Ca^{2+} spikes and we convert the spikes into a sequence of ISI times — shown as the black ticks on the x-axis of Figure 3.1. Subsequently, we calculate the MLE $\hat{\theta}$ for each of the distributions. The results are shown in the second column of Table 3.1. We compare the models by examining the log likelihood at the MLE, shown as the third column in Table 3.1. We find that the Exponential model performs badly having a log likelihood substantially lower than the other models. The inverse Gaussian model has the largest log likelihood, implying it fits the spike sequence the best. However, the log likelihood of the Gamma is similar, thus either model could be used to describe the spike sequence.

Distribution	MLE of ISI parameter/s	Log Likelihood
Exponential	$(\alpha) = 0.0221$	-312.75
Gamma	$(\alpha, \beta) = (0.229, 10.35)$	-261.86
Inverse Gaussian	$(\lambda, \mu) = (442.13, 45.22)$	-261.06
Log-normal	$(\mu, \sigma) = (3.762, 0.312)$	-268.78
Weibull	$(k, \lambda) = (3.226, 50.41)$	-266.48

Table 3.1: The MLE and log likelihood of the MLE of each distribution fit to Ca^{2+} ISIs from a HEK293 cell challenged with $10\mu\text{M}$ carbachol. The colour of the distribution matches the colour of the probability densities shown in Figure 3.1.

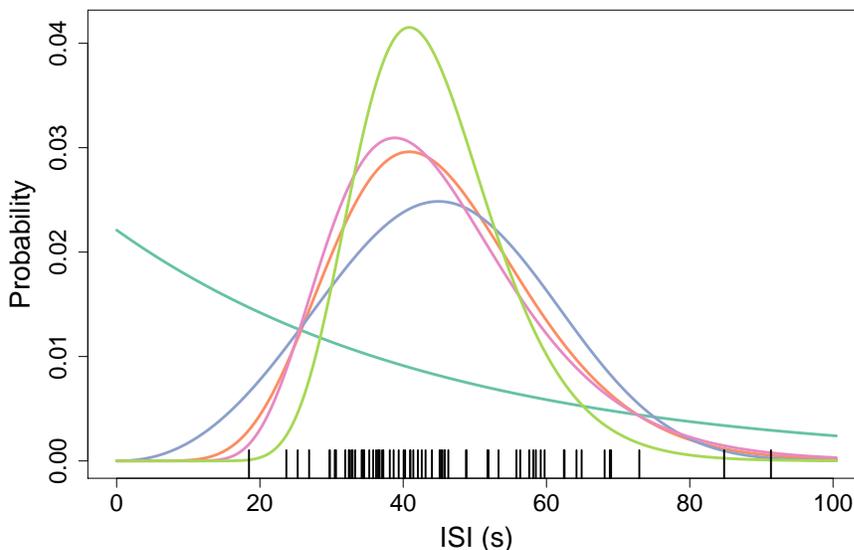


Figure 3.1: Probability density functions corresponding to the MLE of: an Exponential (dark green), a Gamma (orange), inverse Gaussian (pink), log-normal (light green) or Weibull distribution (blue) fit to Ca^{2+} ISIs — shown as black ticks on the x-axis.

To explain this difference in performance we plot the probability density function of the MLE for each model. This is shown in Figure 3.1 where dark green, orange, pink, light green and blue correspond to the Exponential, Gamma, inverse Gaussian, log-normal and Weibull distributions respectively. We see that the largest proportion of spike times are centered around 40s, which is mirrored by the probability densities of all the distributions bar the Exponential. The Exponential model does not capture this due to the limitations of its shape, i.e. it must take the shape of an exponential function. We see that the Gamma and inverse Gaussian have similar probability density functions and hence similar likelihoods, whereas the log-normal and Weibull have similar centers as the Gamma and inverse Gaussian but with narrower and wider tails respectively. Zooming in on the region between 0s and 20s we see a single ISI just before 20s. In this region we observe that the Weibull's probability density is too high — we would expect more ISIs smaller than 20s — and the log-normal's probability density is too low — the density at the lowest ISI time is almost zero. This illustrates why the log-normal and Weibull do not fit the ISIs as well as the Gamma and inverse Gaussian models.

The above example demonstrates that the choice of ISI distribution is crucial, since we found that the distribution affects how accurately the model describes the Ca^{2+} ISIs. This is due to the fact that shape of the probability density function varies for each family of probability distributions. Thus, when considering the ISI distribution for non-stationary spike sequences we need to consider several distributions as some may describe the Ca^{2+} spikes better than others.

Returning to our model, we want to create non-stationary ISI distributions by generalising the distributions — the Exponential, Gamma, inverse Gaussian, log-normal and Weibull — described in the stationary case.

We begin with the Exponential distribution. This differs from the other distributions because it only has a single parameter α . A point process with identical Exponential(α) ISI distributions is known as a Poisson process with rate α because the number of events in the interval $[s, t]$ is a Poisson random variable with mean $\alpha(t - s)$. The non-stationary generalisation of the Poisson process is called the

inhomogeneous Poisson process, where the parameter α is replaced with the rate function $\alpha(t)$. Thus, we shall call the inhomogeneous Exponential ISI distribution

$$p(t, s|\alpha) = \alpha(t) \exp \left\{ - \int_s^t \alpha(u) du \right\}.$$

Consider now a general stationary ISI distribution $\psi(t, s|\theta)$ with parameter θ . To generalise to an inhomogeneous distribution we rescale time via the positive intensity function $x(t)$. This maps the original experiment time t to a new time z by

$$z(t) = \int_0^t x(u) du.$$

Thus by the transformation of variables we generalise the stationary ISI distribution $\psi(t, s|\theta)$ into a non-stationary ISI distribution $p(t, s|x, \theta)$ by

$$\begin{aligned} p(t, s|x, \theta) &= \frac{dz}{dt} \psi(z(t), z(s)|\theta), \\ &= \frac{dz}{dt} \psi(z(t) - z(s)|\theta), \\ &= x(t) \psi \left(\int_0^t x(u) du - \int_0^s x(u) du \middle| \theta \right), \\ &= x(t) \psi \left(\int_s^t x(u) du \middle| \theta \right), \\ &= x(t) \psi(X(s, t)|\theta), \end{aligned}$$

where $X(s, t) = \int_s^t x(u) du$.

We apply the transformation to the Gamma, inverse Gaussian, log-normal and Weibull distributions. We obtain the following probability density functions for: the inhomogeneous Gamma ISI distribution

$$p(t, s|x, \alpha, \beta) = \frac{\beta x(t)}{\Gamma(\alpha)} [\beta X(s, t)]^{\alpha-1} \exp \{-\beta X(s, t)\} \quad \alpha, \beta > 0,$$

the inhomogeneous inverse Gaussian distribution

$$p(t, s|x, \lambda, \mu) = x(t) \left(\frac{\lambda}{2\pi X(s, t)^3} \right)^{0.5} \exp \left\{ - \frac{\lambda(X(s, t) - \mu)^2}{2\mu^2 X(s, t)} \right\} \quad \lambda, \mu > 0,$$

the inhomogeneous log-normal distribution

$$p(t, s|x, \mu, \sigma) = \frac{x(t)}{X(s, t) \sigma \sqrt{2\pi}} \exp \left\{ - \frac{(\log X(s, t) - \mu)^2}{2\sigma^2} \right\} \quad \mu \in \mathbb{R}, \sigma > 0,$$

and the inhomogeneous Weibull distribution

$$p(t, s|x, k, \lambda) = \frac{x(t)k}{\lambda} \left(\frac{X(s, t)}{\lambda} \right)^{k-1} \exp \left\{ - \left(\frac{X(s, t)}{\lambda} \right)^k \right\} \quad \lambda, k > 0.$$

Thus, each inhomogeneous ISI distribution has $x(t)$ as an additional parameter. The parameter set of the inhomogeneous Gamma, inverse Gaussian, log-normal and Weibull are (x, α, β) , (x, λ, μ) , (x, μ, σ) and (x, k, λ) , respectively.

We first need to check if each parameter in the ISI distribution is identifiable. This means that we can theoretically learn the true values of the parameters if given an infinite number of observations. Moreover, a model is non-identifiable if two distinct parameterisations lead to identical probability distributions.

Begin by considering the Gamma ISI distribution. Notice that β is always multiplied by either a factor of $x(t)$ or $X(s, t)$. Thus, it appears that we cannot untangle β and the intensity function $x(t)$. Indeed, consider the Gamma distribution with parameters $(ax, \alpha, \beta/a)$ with $a > 0$ a constant, where we have varied the values of β and x but maintained the value of their product. Rearranging we get

$$\begin{aligned} p(t, s|ax, \alpha, \beta/a) &= \frac{(\beta/a)ax(t)}{\Gamma(\alpha)} [(\beta/a) \int_s^t ax(u)du]^{\alpha-1} \exp \left\{ -(\beta/a) \int_s^t ax(u)du \right\}, \\ &= \frac{\beta x(t)}{\Gamma(\alpha)} [(\beta/a)a \int_s^t x(u)du]^{\alpha-1} \exp \left\{ -(\beta/a)a \int_s^t x(u)du \right\}, \\ &= \frac{\beta x(t)}{\Gamma(\alpha)} [\beta X(s, t)]^{\alpha-1} \exp \{ -\beta X(s, t) \}, \\ &= p(t, s|x, \alpha, \beta). \end{aligned}$$

Thus, we find two parameterisations with identical probability distributions and the Gamma ISI distribution is non-identifiable. Furthermore, all the other considered ISI distributions are also non-identifiable. This is shown in Table 3.2, where each distribution is given a pair of unique parameters that give rise to identical distributions — for proof these parameter sets lead to identical distributions see Appendix A.

To resolve the non-identifiability issues we need to constrain the ISI distributions. One method for doing this is restricting the stationary distributions we used to create the inhomogeneous ISI distributions. For example we could restrict the stationary

Distribution	Base parameters	Identical to ..
Gamma	(x, α, β)	$(ax, \alpha, \beta/a)$
Inverse Gaussian	(x, λ, μ)	$(ax, a\lambda, a\mu)$
Log-normal	(x, μ, σ)	$(ax, \mu + \log a, \sigma)$
Weibull	(x, k, λ)	$(ax, k, a\lambda)$

Table 3.2: Parameterisations of the inhomogeneous ISI distributions which lead to non-identifiability, where $a > 0$ is a constant.

Gamma distribution from two parameters (α, β) to one parameter γ by setting $\alpha = \beta = \gamma$ or $\alpha = \gamma$ and $\beta = \gamma^2$. However, there are infinitely many possibilities for restricting the stationary ISI distribution. Thus we want to find a more principled way to restrict these distributions whilst giving meaning to the remaining parameters.

Recall that the intensity function is a mathematical construct that allows time-dependent spiking processes to be factorised into individual ISIs. Thus, it would be advantageous to restrict the parameterisation in such a manner to give some meaning to the intensity function. Furthermore, it would be beneficial if the intensity function had the same meaning for each ISI distribution. In this case, after fitting, say, the Gamma and inverse Gaussian model to Ca^{2+} spikes we can check if the intensity functions of both models are similar, which would show consistency between the models.

Taking this into account, we choose to restrict the ISI distributions by setting the mean to one in the corresponding stationary ISI distributions. With this condition the intensity function coincides with the mean spike rate irrespective of the chosen stationary ISI distribution, i.e. Gamma, inverse Gaussian, etc.

To visualise the impact of setting the mean of the stationary ISI distribution to one, we simulate spike sequences from a variety of inhomogeneous models. The intensity function $x(t)$ is the same for each model — the red line in Figure 3.2(A) and (B) — and is drawn from a GP. The first model has a Gamma ISI distribution with parameters $\alpha = 1.8$ and $\beta = 1.8$, and the second model also has Gamma ISI

distribution but with the parameters $\alpha = 3.6$ and $\beta = 1.8$. Recall that the mean of a Gamma distribution is α/β , hence the mean for the first and second models are one and two respectively. We simulate 1000 spike sequences from both models and bin the spikes to calculate the mean spiking rate — this is shown as the light and dark grey boxes in Figure 3.2(A) for the first and second model respectively. We find that only the spikes from the first model have mean spiking rate equalling the intensity function. In fact, the second model's mean spiking rate is equal to half the intensity function, shown as the red dotted line. Consequently, we find that only the Gamma ISI distribution with mean one has its mean spike rate coincide with the intensity function.

Furthermore, we consider a third model with the same intensity function but whose ISI distribution follows an inverse Gaussian with parameters $\lambda = 1.8$ and $\mu = 1$, which has mean one. We simulate 1000 spike sequences from this model. In Figure 3.2(B), we compare the mean spike rate of this model with the first model — Gamma ISI distribution with mean one. The red and blue boxes correspond to binning the spike sequences from the inverse Gaussian and Gamma models respectively. The overlap of the boxes is coloured purple. We see that the mean spike rate from both models are near identical. This shows that the intensity function of both models correspond to the mean spike rate, thus allowing us to compare intensity functions from models using different underlying probability distributions.

Henceforth, we use inhomogeneous ISI distributions generalised from stationary distributions with mean one (see page 71 for the mean of the ISI distributions), the parameterisations of which are shown in Table 3.3.

Applying the time-rescaling transformation to the new restricted one-parameter version of the stationary distributions leads to the following probability density functions for: the inhomogeneous Gamma ISI distribution

$$p(t, s|x, \gamma) = \frac{\gamma x(t)}{\Gamma(\gamma)} [\gamma X(s, t)]^{\gamma-1} \exp\{-\gamma X(s, t)\} \quad \gamma > 0,$$

the inhomogeneous inverse Gaussian ISI distribution

$$p(t, s|x, \lambda) = x(t) \left(\frac{\lambda}{2\pi X(s, t)^3} \right)^{0.5} \exp \left\{ - \frac{\lambda(X(s, t) - 1)^2}{2X(s, t)} \right\} \quad \lambda > 0,$$

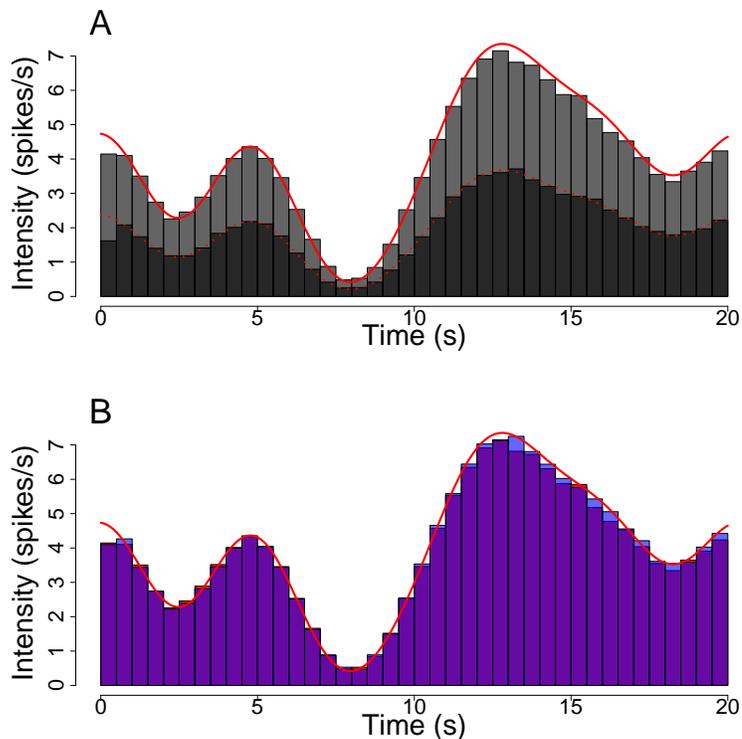


Figure 3.2: Comparing the intensity function — the red line in both plots — to histograms of 1000 simulated spike sequences. In (A) the light and dark boxes correspond to spikes generated from the model with Gamma ISI distributions with parameters $\text{Gamma}(1.8,1.8)$ and $\text{Gamma}(3.6,1.8)$ respectively. In (B) the spike sequences are simulated from a $\text{Gamma}(1.8,1.8)$ and an inverse Gaussian $(1.8,1)$ resulting in the histograms in blue and red respectively, whose overlap is purple.

Distribution	Original parameters	New parameter
Gamma	(α, β)	$(\gamma, \gamma), \gamma > 0$
Inverse Gaussian	(λ, μ)	$(\lambda, 1), \lambda > 0$
Log-normal	(μ, σ)	$(-\mu, \sqrt{2\mu}, \mu > 0)$
Weibull	(k, λ)	$(k, \frac{1}{\Gamma(1+1/k)}), k > 0$

Table 3.3: One parameter version of the stationary ISI distributions with mean one.

the inhomogeneous log-normal ISI distribution

$$p(t, s|x, \mu) = \frac{x(t)}{2X(s, t)\sqrt{\pi\mu}} \exp \left\{ -\frac{(\log X(s, t) + \mu)^2}{4\mu} \right\} \quad \mu > 0$$

and the inhomogeneous Weibull ISI distribution

$$p(t, s|x, k) = \frac{x(t)k}{\Gamma(1+1/k)} \left(\frac{X(s, t)}{\Gamma(1+1/k)} \right)^{k-1} \exp \left\{ -\left(\frac{X(s, t)}{\Gamma(1+1/k)} \right)^k \right\} \quad k > 0.$$

Recall, we also have the Exponential ISI distribution giving rise to an inhomogeneous Poisson process. We can replace $\alpha(t)$ with $x(t)$ since both describe the mean spiking rate of the process. Thus the inhomogeneous Exponential ISI distribution is given by

$$p(t, s|x) = x(t)e^{-X(s, t)}.$$

Thus, we have created five inhomogeneous ISI distributions based on the Exponential, Gamma, inverse Gaussian, log-normal and Weibull probability distributions. The time-dependence of each distribution comes from the intensity function $x(t)$ which for our models corresponds to the time-dependent spike rate. All the ISI distributions bar the Exponential also have a single ISI parameter γ , λ , μ and k for the Gamma, inverse Gaussian, log-normal and Weibull, respectively. This parameter describes the shape and variance of the ISI distribution.

In Figure 3.3 we show how the ISI parameter affects simulated Ca^{2+} spikes for the inhomogeneous Gamma ISI distribution. We simulate spike sequences from two Gamma ISI models with the same intensity function (Panel A), where the first has ISI parameter $\gamma = 3$ (Panel B) and the second has $\gamma = 20$ (Panel C). We see that the larger ISI parameter leads to more regular spiking. This corresponds to an ISI distribution whose mass is centered around the mean with small tails.

Each inhomogeneous ISI distribution contains the stationary distribution as a special case, when the intensity function is constant. Specifically, if $x(t) = a$ for some $a > 0$ then:

- the inhomogeneous Gamma ISI distribution with parameters (x, γ) is equivalent to a Gamma distribution with rate γa and shape γ ,

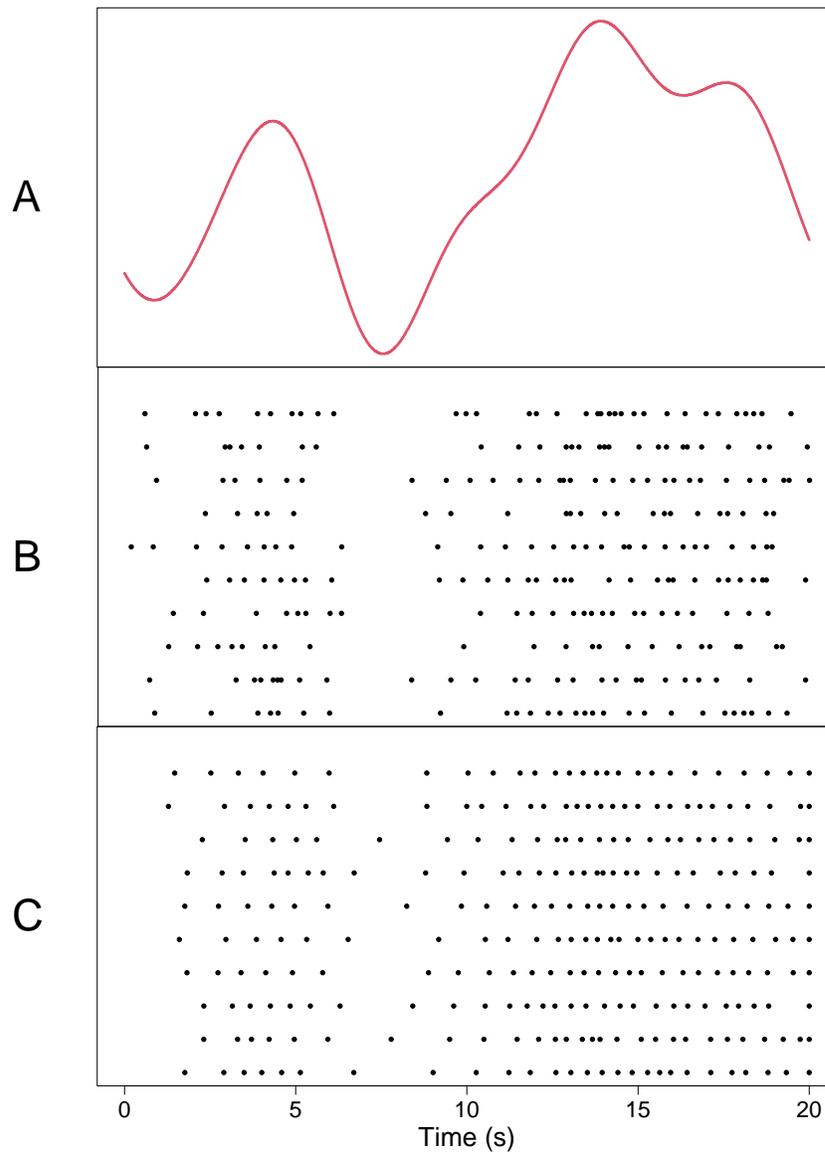


Figure 3.3: Raster plots comparing spike sequences drawn from two Gamma ISI distributions with the same intensity function — shown in (A) — and differing ISI parameter. The sequences in (B) and (C) correspond to γ values of 3 and 20 respectively.

- the inhomogeneous inverse Gaussian ISI distribution with parameters (x, λ) is equivalent to a inverse Gaussian distribution with mean $1/a$ and shape

parameter λ/a ,

- the inhomogeneous log-normal ISI distribution with parameters (x, μ) is equivalent to a log-normal distribution with mean $\log a - \mu$ and standard deviation $\sqrt{2\mu}$,
- the inhomogeneous Weibull ISI distribution with parameters (x, k) is equivalent to a Weibull distribution with shape k and scale $(a\Gamma(1 + 1/k))^{-1}$,

We will use the five inhomogeneous ISI distributions — namely Exponential, Gamma, inverse Gaussian, log-normal and Weibull — in our analysis of Ca^{2+} spike sequences and explore which, if any, of the distributions best describe Ca^{2+} data and what we can learn from the results. This is shown in Chapter 5.

We now compare our inhomogeneous ISI distributions to previously explored distributions from the literature. Tilunaite et al. [16] used three ISI distributions in their work, the so-called inhomogeneous Poisson (IP), inhomogeneous Gamma (IG) and the inhomogeneous inverse Gaussian (IIG). The IP and IG agree exactly with our inhomogeneous Exponential ISI distribution and inhomogeneous Gamma ISI distribution. However, their IIG ISI distribution has a different parameterisation to our inhomogeneous inverse Gaussian ISI distribution. Their model is generalised from the inverse Gaussian with parameters $(1, \alpha)$ whereas our model is based on the parameters $(\lambda, 1)$. We do not use their parameterisation because their parameterisation does not lead to a distribution with mean one, and as such the intensity function $x(t)$ does not align with the mean spiking rate.

Thus the ISI distributions we consider builds upon previous work using the inhomogeneous Exponential and Gamma ISI distributions and explores new distributions including the log-normal and Weibull, which could better describe the Ca^{2+} spike sequences.

3.2 Mixing issues with the GP and their resolutions

In this section, we outline the difficulty in inferring the start and end of the intensity function when using a GP prior. In particular, the standard MCMC algorithm often gets stuck in regions of the posterior distribution with large intensity before the first spike time and after the last spike time. In response, we create a bespoke proposal mechanism to work in tandem with the standard approach to improve the mixing of the MCMC.

When using a GP prior for the logarithm of the intensity function we use the under-relaxed MCMC method. This method was first proposed by Neal in 1995 [118] and applied to GPs by Adams et al [119]. We use this method because it allows the intensity function to be updated as a single block and the acceptance probably simplifies to the ratio of likelihoods. To propose a candidate intensity function \mathbf{x}^* we use

$$\log \mathbf{x}^* = \sqrt{1 - \omega^2} \log(\mathbf{x}) + \omega \boldsymbol{\nu}, \quad \boldsymbol{\nu} \sim \text{GP prior}, \quad (3.1)$$

where $\boldsymbol{\nu}$ is a draw from the GP prior and $\omega \in (0, 1]$. The parameter ω is a tuning parameter that acts similarly to the variance parameter in the RW-Metropolis algorithm. When ω is close to 1 the candidate intensity functions closely resemble draws from the prior distribution, whereas when ω is close to 0 the candidate intensity function is similar to the current intensity function.

We have found that using this method alone can lead to posterior intensity functions that have undesirable properties. Namely, the intensity at the beginning and end of the intensity function can be artificially high. We show this in the following example, where we apply the under-relaxed method to a Ca^{2+} spike sequence obtained from a HEK293 cell challenged with a time-varying stimulus. The Ca^{2+} spike sequence can be seen as the black ticks in Figure 3.4. The spikes occur in 3 bursts at approximately 4s, 10s and 17s. Figure 3.4 shows the posterior mean and 95% credible region for the intensity function as the black line and red shaded region,

respectively. The posterior ISI parameter distribution had mean 3.4 and 95% credible region [1.4, 6.2]. As expected we find peaks in intensity surrounding the regions where the spikes occur. However, we also find the intensity increases in $[0s, 1.5s]$ when approaching 0s and in $[19s, 20s]$ when approaching 20s. Since there are no spikes in $[0s, 1.5s]$ and $[19s, 20s]$ we would expect the intensity to be close to zero. To test

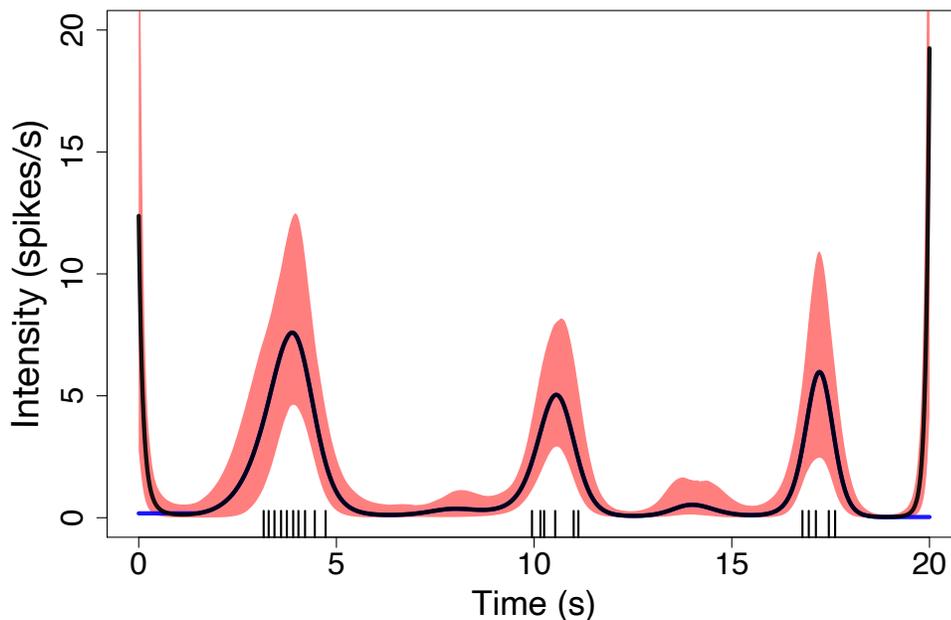


Figure 3.4: Posterior mean — black line — and 95% credible interval — red shaded area — for the intensity function, from the model with an inhomogeneous Gamma ISI distribution. The inference was calculated using 80,000 iterations with a burn-in of 50,000. The covariance parameters are $\sigma_f^2 = 1000$, $\sigma_n^2 = 1e^{-5}$ and $l = 1.50$ — l is obtained using the PWC prior, see Section 3.3 for more detail. The intensity function is split into $N = 2000$ steps and the under-relaxed parameter $\omega = 0.001$. The ISI parameter had an exponential prior with rate 0.01. The blue line shows a candidate intensity function created by flattening the posterior mean in $[0s, 1.5s]$ and $[19s, 20s]$. The black ticks represent the Ca^{2+} spike times used to fit the model.

whether the intensity function should be close to zero in these regions we propose a candidate intensity function equal to the mean of the posterior intensity function except for the regions $[0s, 1s]$ and $[19s, 20s]$ where the intensity is flattened — shown in blue in Figure 3.4. To flatten the intensity function we set the intensity in $[0s, 1s]$ and $[19s, 20s]$ to the value of the posterior mean intensity function at 1s and 19s, respectively. We found that the log likelihood of the posterior mean is -8.90 and that of the newly proposed intensity function is -7.20 . Therefore, it is more likely that the intensity function stays close to zero in the regions $[0s, 1.5s]$ and $[19s, 20s]$.

To investigate why the under-relaxed method gives rise to large intensity at 0s, we generate trace plots of the sampled intensity functions at 0s. This is illustrated in Figure 3.5, for three traces where each is produced using a different random seed. Each plot shows the value of the sampled intensity at 0s against the iteration number, with no burn-in. In Figure 3.5(A) we see that the intensity varies in $[0, 8]$ for the first 150,000 iterations and after this the intensity remains around zero. In Figure 3.5(B) the intensity converges close to zero within the first 25,000 iterations. However, at around the 250,000th iteration we find that the intensity increases and does not converge to zero in the next 50,000 iterations. This shows that the under-relaxed method is liable to get stuck in regions where the intensity at 0s is large. Furthermore, running more iterations or having a larger burn-in will not alleviate this issue because we can move from zero intensity back to regions with large intensity as shown in Figure 3.5(B). To visualise why we can go from low intensity at 0s to high intensity we plot the intensity functions corresponding to the coloured dots in Figure 3.5(C). This is shown in Figure 3.6, where the intensity functions go from purple through green to yellow as the intensity at 0s increases. We see that the intensity function was initially flat in $[0s, 3s]$ and as the iterations progress we find a large peak of intensity at 0s which quickly decreases to zero within the first 2s. We also find that the intensity at the 3 peaks — approximately at 4s, 10s and 17s — vary considerably, which mirrors the uncertainty of the intensity function. As we will show next, we accept intensity functions with high intensity in $[0s, 1s]$ because the likelihood over the rest of the domain is improved.

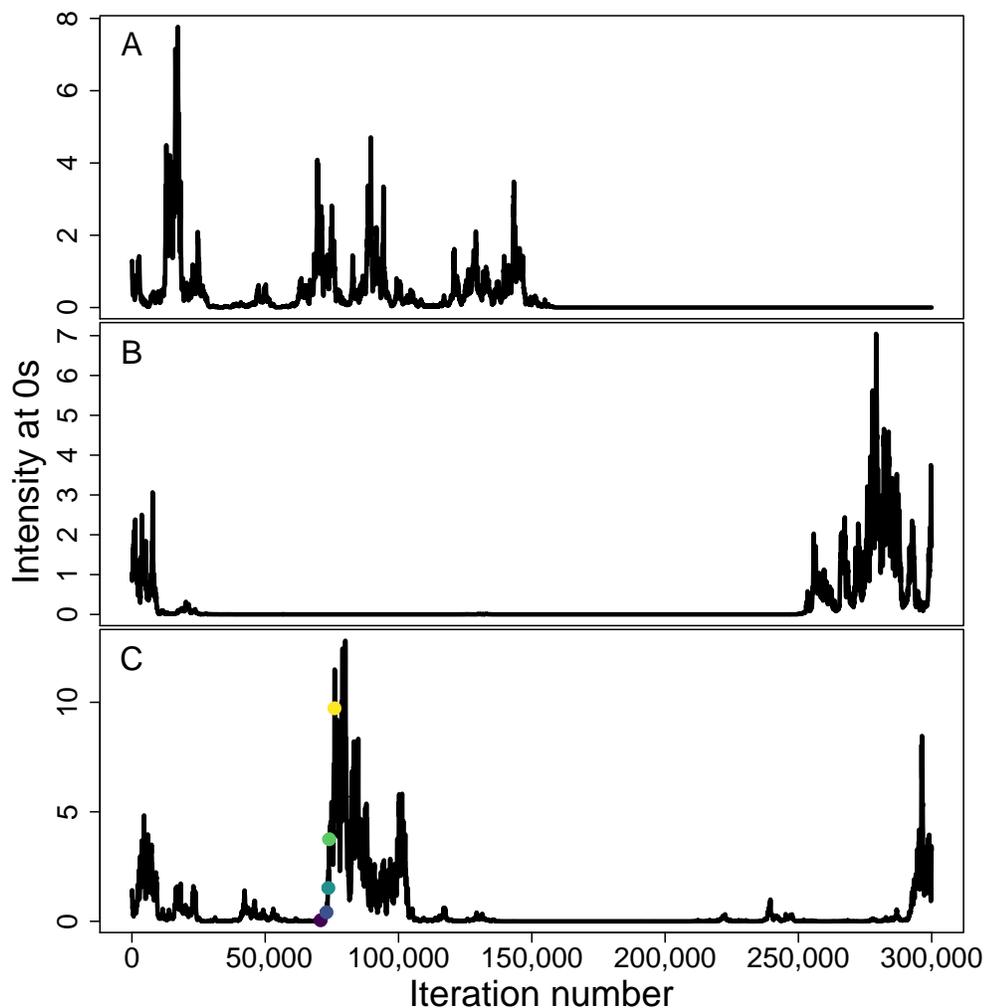


Figure 3.5: Trace plots of the intensity function at time 0s over 300,000 iterations, for the GP prior utilising an under-relaxed method. Each plot shows the results obtained for the same spike sequence; the only difference is the random seed used. The inference parameters are the same as Figure 3.4, using the same spike data. The coloured dots in (C) correspond to the coloured intensity functions shown in Figure 3.6.

We partition time into two regions $[0s, 3.16s]$ and $[3.16s, 20s]$ which corresponds to the time up to the first spike and the time after the first spike. Since our likelihood function factorises into individual ISIs we can split the likelihood into contributions

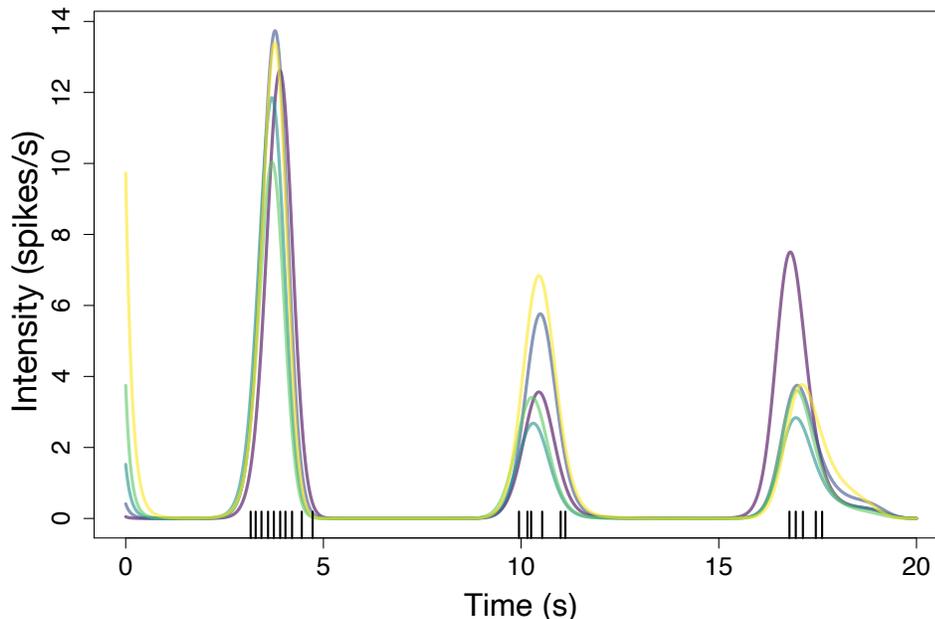


Figure 3.6: Sampled intensity functions, corresponding to the coloured dots in Figure 3.5. Intensity functions go from purple through green to yellow, as the intensity at 0s increases. The black ticks represent the Ca^{2+} spike times.

in $[0\text{s}, 3.16\text{s}]$ and $[3.16\text{s}, 20\text{s}]$. For example the likelihood of the purple and yellow intensity functions in Figure 3.6 is 3.45 and 4.75, respectively. Therefore, the yellow intensity function is more likely. However, the likelihood from $[0\text{s}, 3.16\text{s}]$ is 0.11 and -1.15 for the purple and yellow lines, respectively and the likelihood from $[3.16\text{s}, 20\text{s}]$ is 3.34 and 5.80. We find that the improvement of the likelihood in $[3.16\text{s}, 20\text{s}]$ outweighs the decrease of the likelihood in $[0\text{s}, 3.16\text{s}]$ when comparing the yellow and purple intensity functions. Therefore, we accept intensity functions with an increasing intensity towards 0s, i.e the yellow intensity function. Moreover, the same holds true for the high intensity found at the end of the experiment.

Therefore, we want to create a new proposal mechanism to work in tandem with the under-relaxed method to improve the mixing at the beginning and end of the

intensity function. We will do this by proposing candidate intensity functions that vary only on these regions of interest. In particular, we will call the proposals the *data-driven method at the start* and the *data-driven method at the end*, for the proposals that only change values in the beginning and end of the experiment, respectively.

3.2.1 Method

We first present the steps for updating the intensity function only at the beginning of the intensity function. Assume initially that we have discretised time into $N + 1$ steps by $\mathbf{t} = \{t_0, t_1, \dots, t_N\}$ defined as $t_i = iT/N$ for $i \in \{0, \dots, N\}$. Furthermore, the current value of the discretised intensity function is $\mathbf{x} = \{x_0, \dots, x_N\}$ where $x_i = x(t_i)$ for $i \in \{0, \dots, N\}$. We need to decide how to partition \mathbf{x} into two regions — the part we update and the part that remains unchanged. We choose to sample the width M at which to delineate the two partitions uniformly between time zero and just after the first spike time. Suppose the first spike occurs at time t_ξ , where $\xi \in \{1, \dots, N\}$. Then the partition value M is drawn uniformly from the set $\{1, 2, \dots, \min(\xi + w, N)\}$, where w controls how far past the first spike we can sample. The minimum is used to only allow valid index values. We split time into two groups A and B, one either side of t_M : $A = \{t_0, \dots, t_M\}$ and $B = \{t_{M+1}, \dots, t_N\}$. We similarly split \mathbf{x} into two groups $\mathbf{x}_A = \{x_0, \dots, x_M\}$ and $\mathbf{x}_B = \{x_{M+1}, \dots, x_N\}$. Therefore, the candidate intensity function \mathbf{x}^* consists of two parts: \mathbf{x}_A^* describing the new values we propose in region A and \mathbf{x}_B^* which corresponds to the unchanged values in region B, $\mathbf{x}_B^* = \mathbf{x}_B$.

In Figure 3.7 we give an example of splitting \mathbf{x} — the black line — into two regions. In this example \mathbf{x} is discretised into 2000 steps. The first spike occurs at 3.08s which corresponds to the 309th element of \mathbf{t} . Taking $w = 100$ we sample the partition index M from $\{1, \dots, 408\}$ which corresponds to times $\{0.01, \dots, 4.08\}$ — which is shown as the grey box. A realisation is indicated by the red dotted line where $M = 326$. Thus, the thin black line in region B will remain the same for the candidate function and we propose new values in region A.

We now need to choose a method to propose \mathbf{x}_A^* given the current value \mathbf{x}_A . A natural option is to propose candidate values from a multivariate normal (MVN)

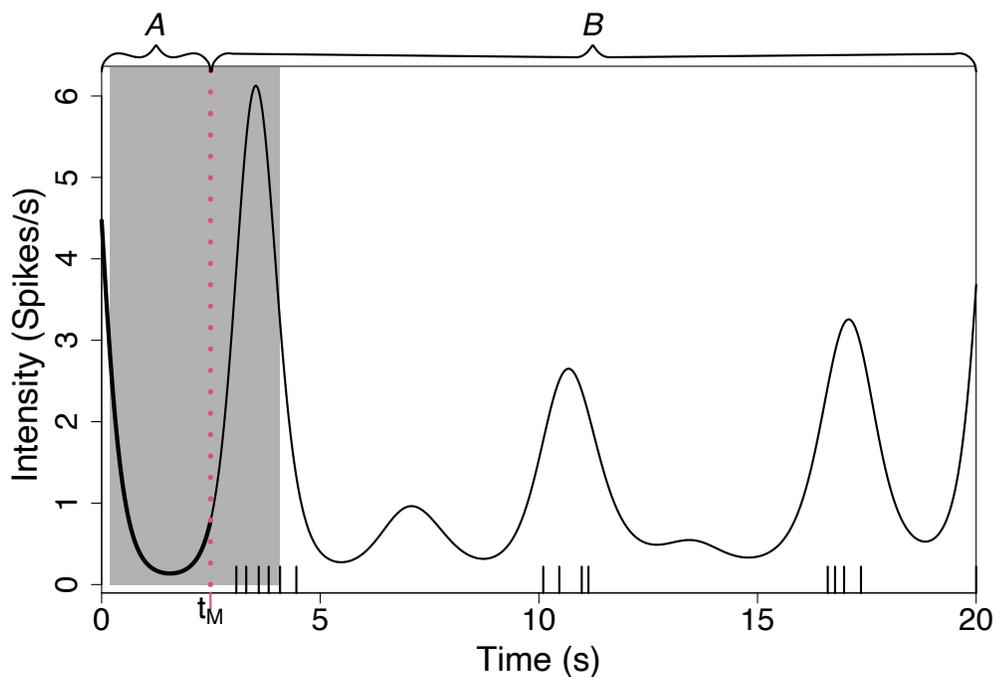


Figure 3.7: Illustration of how to partition \mathbf{x} into two regions A and B . The partition value is drawn uniformly from the grey region. An example draw t_M is shown by the dotted red line. The thick black line — left of t_M — is the region of \mathbf{x} we will propose new values for, whereas the thin black line — right of t_M — will remain unchanged. The black ticks on the x-axis represent spike times.

distribution. Recall that we say the random vector $\mathbf{X} = \{X_0, X_1, \dots, X_M\}$ comes from a $(M + 1)$ -dimensional MVN distribution $\mathbf{X} \sim \mathcal{N}_{M+1}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ if its probability density function $\varphi_{M+1}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ of $\mathbf{x} = \{x_0, \dots, x_M\}$ is given by

$$\varphi_{M+1}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{(M+1)/2} \det(\boldsymbol{\Sigma})^{-1/2} \exp \left[-\frac{1}{2}(\mathbf{X} - \boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}(\mathbf{X} - \boldsymbol{\mu}) \right],$$

where $\boldsymbol{\mu} \in \mathbb{R}^{M+1}$ is the vector of means and $\boldsymbol{\Sigma} \in \mathbb{R}^{(M+1) \times (M+1)}$ is the covariance matrix. In our case we want our proposal to depend on the current value of the intensity function. We do this by making the mean of the MVN depend on \mathbf{x}_A via a function f giving $\boldsymbol{\mu} = f(\mathbf{x}_A)$. The choice of f is discussed in Section 3.2.2. Recall that the prior for the intensity function is a GP with square exponential covariance

with hyperparameters $\{\sigma_f^2, \sigma_n^2, l\}$ representing the signal variance, noise, and length scale, respectively. We would like our proposal to match the ‘shape’ of the GP prior. Therefore the covariance matrix Σ for our proposal inherits the length scale and noise parameter of the GP. Formally, $\Sigma = \{\Sigma_{i,j}\}_{i,j \in \{0, \dots, M\}}$ where

$$\Sigma_{i,j} = \sigma^2 \exp \left[\frac{(t_i - t_j)^2}{2l^2} \right] + \delta(t_i - t_j) \sigma_n^2,$$

where l and σ_n^2 come from the GP prior. The variance σ^2 is often taken small, such that the proposed values are close to $f(\mathbf{x}_A)$.

Recall that we defined the intensity function to be a positive function. Therefore to only allow positive intensity functions we choose to propose intensity functions on the logarithmic scale. Thus, the proposal on region A is given by $\mathbf{x}_A^* = e^{\mathbf{x}_A^\dagger}$ where $\mathbf{x}_A^\dagger \sim \mathcal{N}_{M+1}(f(\mathbf{x}_A), \Sigma)$ and the logarithm is subsumed into f .

However, if we choose to propose new values in this manner there is no guarantee the new intensity function would be continuous at t_M . To mitigate this issue we will propose candidate values in the region A by drawing from the MVN distribution described above conditioned on the first K values of \mathbf{x}_B . The conditioned MVN distribution follows a MVN distribution with mean and covariance matrix outlined below. To construct the conditional MVN we first need a MVN distribution defined on region A and the values we wish to condition on, namely $C = \{t_{M+1}, \dots, t_{M+K}\}$ and $\mathbf{x}_C = \{x_{M+1}, \dots, x_{M+K}\}$. It follows that $\mathbf{x}_{AUC}^\dagger \sim \mathcal{N}_{M+K+1}(f(\mathbf{x}_{AUC}), \Sigma)$ such that

$$\mathbf{x}_{AUC}^\dagger = \begin{bmatrix} \mathbf{x}_A^\dagger \\ \mathbf{x}_C^\dagger \end{bmatrix} \sim \mathcal{N}_{M+K+1} \left(\begin{bmatrix} f(\mathbf{x}_A) \\ f(\mathbf{x}_C) \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \right),$$

where $f(\mathbf{x}_{AUC})$ is partitioned into $f(\mathbf{x}_A)$ and $f(\mathbf{x}_C)$. Σ has been partitioned into four blocks: Σ_{11} of size $(M+1) \times (M+1)$, Σ_{12} of size $(M+1) \times K$, Σ_{21} of size $K \times (M+1)$, and Σ_{22} of size $K \times K$. Then the conditional distribution of $\mathbf{x}_{AUC}^\dagger | (\mathbf{x}_C^\dagger = \log \mathbf{x}_C)$ is a $(M+1)$ -dimensional MVN distribution with mean $\bar{\boldsymbol{\mu}}$ and covariance $\bar{\Sigma}$ given by

$$\bar{\boldsymbol{\mu}} = f(\mathbf{x}_A) + \Sigma_{12} \Sigma_{22}^{-1} (\log \mathbf{x}_C - f(\mathbf{x}_C)) \quad (3.2)$$

and

$$\bar{\Sigma} = \Sigma_{11} - \Sigma_{12} \Sigma_{22}^{-1} \Sigma_{21}.$$

Therefore, we propose candidate values \mathbf{x}_A^* on region A by $\mathbf{x}_A^* = e^{\mathbf{x}_A^\dagger}$ where $\mathbf{x}_A^\dagger \sim \mathcal{N}_{M+1}(\bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\Sigma}})$. Thus the candidate function $\mathbf{x}^* = [\mathbf{x}_A^*, \mathbf{x}_B]$ is the concatenation of vectors \mathbf{x}_A^* and \mathbf{x}_B . Simply put, the candidate function consists of a draw from a MVN distribution in region A conditioned on the first K values in B and is equal to the current intensity on all other points.

The probability that we accept the candidate function is given by

$$p_{\text{acc}} = \frac{\text{Likelihood}(\mathbf{x}^*)\text{Prior}(\mathbf{x}^*)q(\mathbf{x}^*|\mathbf{x})}{\text{Likelihood}(\mathbf{x})\text{Prior}(\mathbf{x})q(\mathbf{x}|\mathbf{x}^*)}.$$

The likelihood is defined in equation (2.10). We work with a Gaussian Process prior for the logarithm of the intensity function — with mean zero and square exponential covariance function with hyperparameters σ_f^2 , σ_n^2 and l . Therefore, in discretised time we have $\text{Prior}(\mathbf{x}) = \varphi_{N+1}(\log \mathbf{x}; \mathbf{0}, \boldsymbol{\Sigma}_{\text{prior}})$, where $\boldsymbol{\Sigma}_{\text{prior}}$ denotes the covariance matrix on \mathbf{t} for the GP. Since the hyper-parameters remain the same for both \mathbf{x}^* and \mathbf{x} the prior ratio simplifies

$$\begin{aligned} \frac{\text{Prior}(\mathbf{x}^*)}{\text{Prior}(\mathbf{x})} &= \frac{(2\pi)^{(N+1)/2} \det(\boldsymbol{\Sigma}_{\text{prior}})^{-1/2} \exp\left[-\frac{1}{2}(\log \mathbf{x}^*)\boldsymbol{\Sigma}_{\text{prior}}^{-1}(\log \mathbf{x}^*)\right]}{(2\pi)^{(N+1)/2} \det(\boldsymbol{\Sigma}_{\text{prior}})^{-1/2} \exp\left[-\frac{1}{2}(\log \mathbf{x})\boldsymbol{\Sigma}_{\text{prior}}^{-1}(\log \mathbf{x})\right]} \\ &= \frac{\exp\left[-\frac{1}{2}(\log \mathbf{x}^*)\boldsymbol{\Sigma}_{\text{prior}}^{-1}(\log \mathbf{x}^*)\right]}{\exp\left[-\frac{1}{2}(\log \mathbf{x})\boldsymbol{\Sigma}_{\text{prior}}^{-1}(\log \mathbf{x})\right]}. \end{aligned}$$

Working on the logarithmic scale gives

$$\log\left(\frac{\text{Prior}(\mathbf{x}^*)}{\text{Prior}(\mathbf{x})}\right) = \frac{1}{2} \left((\log \mathbf{x})\boldsymbol{\Sigma}_{\text{prior}}^{-1}(\log \mathbf{x}) - (\log \mathbf{x}^*)\boldsymbol{\Sigma}_{\text{prior}}^{-1}(\log \mathbf{x}^*) \right).$$

$q(\mathbf{x}^*|\mathbf{x})$ is the proposal distribution — the conditional distribution of proposing function \mathbf{x}^* given \mathbf{x} . For our proposal we have $q(\mathbf{x}^*|\mathbf{x}) = \varphi_{M+1}(\log \mathbf{x}_A^*; \bar{\boldsymbol{\mu}}(\mathbf{x}_{AUC}), \bar{\boldsymbol{\Sigma}})$, where we explicitly state the dependency of $\bar{\boldsymbol{\mu}}$ on \mathbf{x}_{AUC} . Similar to the prior ratio, we can simplify the proposal ratio since $\bar{\boldsymbol{\Sigma}}$ is the same for $q(\mathbf{x}^*|\mathbf{x})$ and $q(\mathbf{x}|\mathbf{x}^*)$. This is true because $\bar{\boldsymbol{\Sigma}}$ depends only on the choice of σ^2 , σ_n^2 and l . On the logarithmic scale this gives

$$\begin{aligned} \log\left(\frac{q(\mathbf{x}^*|\mathbf{x})}{q(\mathbf{x}|\mathbf{x}^*)}\right) &= \frac{1}{2} \left((\log \mathbf{x}_A - \bar{\boldsymbol{\mu}}(\mathbf{x}_{AUC}))\bar{\boldsymbol{\Sigma}}^{-1}(\log \mathbf{x}_A - \bar{\boldsymbol{\mu}}(\mathbf{x}_{AUC})) \right. \\ &\quad \left. - (\log \mathbf{x}_A^* - \bar{\boldsymbol{\mu}}(\mathbf{x}_{AUC}))\bar{\boldsymbol{\Sigma}}^{-1}(\log \mathbf{x}_A^* - \bar{\boldsymbol{\mu}}(\mathbf{x}_{AUC})) \right). \quad (3.3) \end{aligned}$$

To propose intensity functions that vary at the end of the experiment time a symmetric argument can be used where we swap the definitions of regions A and B .

3.2.2 Choice of f

In this section we discuss the choice of the function f used to propose \mathbf{x}_A^* . Recall that the idea behind these proposals is to ‘flatten’ the intensity in region A . A natural starting point is to use the current intensity as the mean of the proposed MVN distribution giving $f(\mathbf{x}_A) = f_{\text{cur}} = \log \mathbf{x}_A$ — where the log arises from proposing on the logarithmic scale. One advantage of this approach is that the acceptance probability simplifies due to the proposal ratio cancelling out, because the mean of the MVN simplifies due to the second term in (3.2) cancelling, leaving $\bar{\boldsymbol{\mu}} = \log \mathbf{x}_A$. Substituting into equation (3.3) we get

$$\begin{aligned} \log \left(\frac{q(\mathbf{x}^*|\mathbf{x})}{q(\mathbf{x}|\mathbf{x}^*)} \right) &= \frac{1}{2} \left((\log \mathbf{x}_A - \log \mathbf{x}_A^*) \bar{\boldsymbol{\Sigma}}^{-1} (\log \mathbf{x}_A - \log \mathbf{x}_A^*) \right. \\ &\quad \left. - (\log \mathbf{x}_A^* - \log \mathbf{x}_A) \bar{\boldsymbol{\Sigma}}^{-1} (\log \mathbf{x}_A^* - \log \mathbf{x}_A) \right) \\ &= 0, \end{aligned}$$

since $\bar{\boldsymbol{\Sigma}}$ does not depend on \mathbf{x}_A . Thus, the acceptance ratio simplifies to

$$p_{\text{acc}} = \frac{\text{Likelihood}(\mathbf{x}^*) \text{Prior}(\mathbf{x}^*)}{\text{Likelihood}(\mathbf{x}) \text{Prior}(\mathbf{x})}.$$

However, this approach may be slow at ‘flattening’ the intensity in region A . This is because the candidate intensity functions are centered on \mathbf{x}_A . This can be seen in Figure 3.8(A) which shows ten candidate intensities — with fixed partition — zoomed in on the region $[0, 3.3]$. The red line shows the current intensity function \mathbf{x}_A and the grey lines show ten candidate intensity functions. We see that four of the candidate functions have larger intensity than \mathbf{x}_A and the majority of proposals are close to \mathbf{x}_A . It is also important to note that the smaller \mathbf{x}_A the more difficult it is to propose smaller values since the calculations are done on the logarithmic scale. For example, suppose on the logarithmic scale the intensity at time t is 0 and the drawn candidate value equals -1 , this translates to an original intensity of 1 and candidate intensity

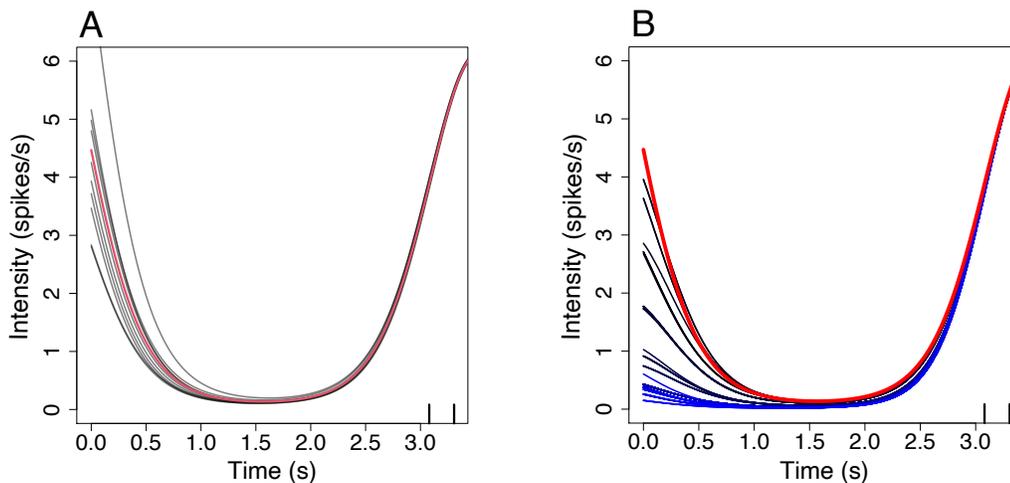


Figure 3.8: Illustration of the proposal mechanism when $f = \log \mathbf{x}_A$. (A) shows ten candidate functions — grey lines — where the current intensity is shown by the red line. (B) The accepted candidate functions from 100 iterations of the proposal mechanism where the original \mathbf{x}_A is shown as the red line. The iterations progress from black to blue. The black ticks show the spike times. The proposal parameters in (A) and (B) are $K = 10$, $l = 1.59$, $\sigma_n^2 = 1e^{-5}$ and $\sigma^2 = 1$. In (A) the width is fixed with $M = 337$, whereas in (B) the width varies in each iteration and $w = 100$.

equal 0.368, hence squashing smaller values closer together. This implies that it will take many iterations to flatten the curve in the region $[0, 1]$. Indeed, this is shown in Figure 3.8(B) where the red line shows \mathbf{x}_A , we then compute 100 iterations, the accepted candidate functions are shown going from the black lines to blue lines as iterations progress. We see that we do indeed steadily flatten the intensity function in the region $[0, 1]$.

So we have found that using $f = f_{\text{cur}}$ does reduce the peak intensity found at time 0s. However, decreasing the intensity often requires a large number of iterations. With this in mind, it may be more intuitive to propose functions whose shape is flatter than the current intensity function. For example proposing intensity functions whose mean is constant. We shall consider two alternative functions for f both of

which are constant. Namely, $f_{\text{mean}} = \{\text{mean}(\log \mathbf{x}_A)\}_{i=0}^M$ which returns a constant vector where each element is the mean of $\log \mathbf{x}_A$ and $f_{\text{min}} = \{\min(\log \mathbf{x}_A)\}_{i=0}^M$ which returns a constant vector where each element is the minimum of $\log \mathbf{x}_A$.

In Figures 3.9(A) and (C) we show ten candidate functions from the proposal distribution using f_{min} and f_{mean} , respectively. The blue lines depict the mean of the proposal distribution. This illustrates how using f_{min} and f_{mean} makes the candidate intensity functions flatter than those using $f = f_{\text{cur}}$ in Figure 3.8. Therefore, if accepted we require fewer iterations than f_{cur} to remove the peak intensity at time 0. We also see a marginal difference between using f_{mean} and f_{min} where the proposed functions using f_{min} are smaller than those using f_{mean} . In (B,D) we plot the accepted candidate functions from 100 iterations of this proposal mechanism, where we begin at the red line. Iterations are shown going from grey to blue lines at the iterations progress. It takes f_{min} three accepted iterations and f_{mean} five accepted iterations for the intensity function to take values close to zero at 0s. This corresponds to the 14th and 15 iteration for f_{min} and f_{mean} , respectively. Hence, we find that it takes f_{min} and f_{mean} fewer iterations to reduce the intensity in $[0\text{s}, 1.5\text{s}]$ compared to f_{cur} .

In the above discussion we have always used the same initial intensity function, and we found that all considered f reduce the high intensity found in the region $[0\text{s}, 1.5\text{s}]$. Therefore when deciding which method to implement we are interested in the speed at which the proposals flatten this region. Therefore, in the particular example shown one would choose to use either f_{min} or f_{mean} as they took a similar amount of iterations. However, in general it may take a large number of iterations to accept a proposal when using f_{min} or f_{mean} . For example in Figure 3.10 we show proposed candidate functions on the logarithmic scale compared to the logarithm of the current intensity function shown in black. The blue lines show candidate functions when $f = f_{\text{cur}}$. We see that these candidate are all close to the current intensity function and have the same shape. Whereas the candidate functions when $f = f_{\text{min}}$ are far from the current intensity function and the shapes vary. In other words some have an uptick at 0s whereas others are decreasing. The large difference in the current intensity function and the proposed candidate functions can lead to a

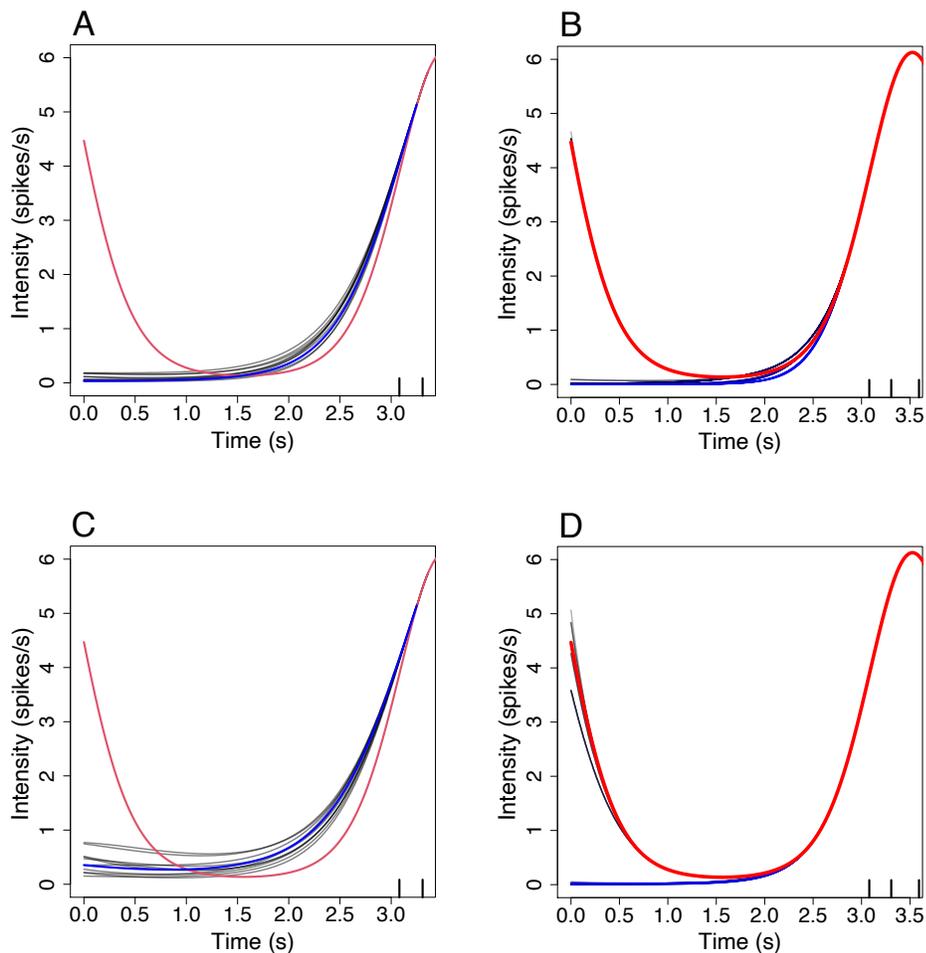


Figure 3.9: (A,B) and (C,D) are illustrations of the proposal mechanism for $f = f_{\min}$ and $f = f_{\text{mean}}$, respectively. (A,C) shows ten candidate functions — grey lines — where the current intensity is shown by the red line, and the mean of the proposal is shown in blue. (B,D) The accepted candidate functions from 100 iterations of the proposal mechanism where the original \mathbf{x}_A is shown as the red line. The iterations progress from black to blue. The black ticks show the spike times. The proposal parameters are the same as in Figure 3.8.

reduced acceptance probability. For example, we find that the likelihood ratio is 1.32 for the candidate intensity function shown by the brown line in Figure 3.10. However,

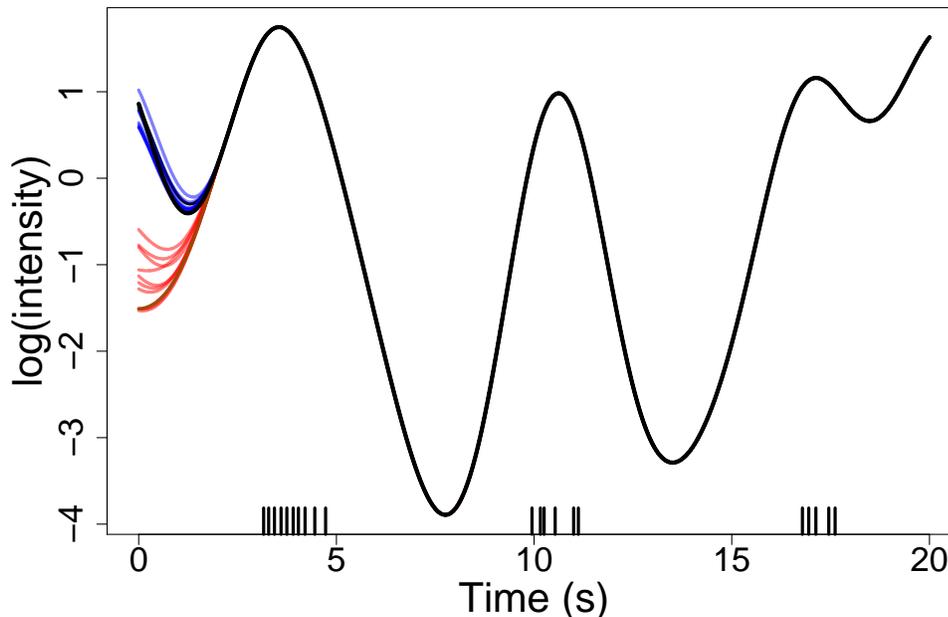


Figure 3.10: The logarithm of proposed candidate functions from using the data-driven method at the start, where red and brown lines show candidate functions using f_{\min} and blue lines show candidate functions using f_{cur} . The logarithm of the current intensity function is shown in black, and spike times as black ticks.

this improvement is outweighed by the proposal ratio which equals -26.13 . We find that only 1 of the candidate intensity functions were accepted for f_{\min} which is lower than the 6 accepted using f_{cur} .

Therefore, we find that if candidate functions are accepted fast enough f_{\min} and f_{mean} will flatten the region quicker. However, there is no guarantee that candidate functions will get accepted. Thus, although f_{cur} reduces the intensity less per iteration it will not get stuck not accepting proposals.

Other functions could be used in this proposal mechanism, and further research could look into whether there is an optimal f for proposing updates to the intensity function on these regions of interest.

3.2.3 Proposal mechanism for GP

We now need to decide how to use this new proposal in conjunction with the under-relaxed method. We choose to use both f_{cur} and f_{min} because f_{min} can quickly flatten the intensity if accepted, and f_{cur} can reduce the intensity if f_{min} fails to. We only want to update at the beginning and end of the intensity function infrequently. This is because we are only using this method to help explore regions that the under-relaxed methods struggles to search. Therefore, we will only use this proposal every 1000th iteration, but we shall use the proposal multiple times. This is to improve the chance of removing undesired features in these regions. We choose to initially do 50 iterations on both the beginning and end of the function using f_{min} . If no candidate functions are accepted under f_{min} then we do a further 50 iterations using f_{cur} . The algorithm for the MCMC is shown in Algorithm 4.

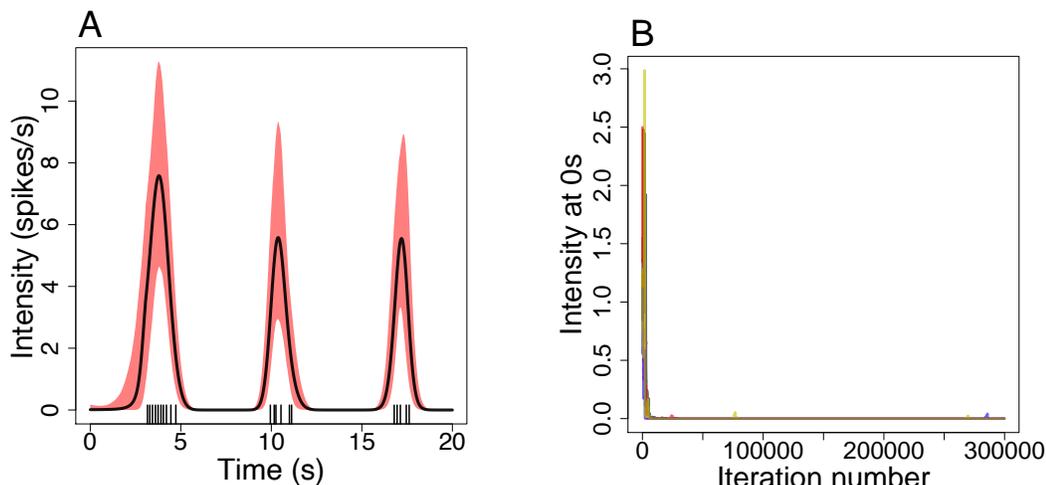


Figure 3.11: (A) gives the posterior mean and 95% credible region using the new proposal mechanism. The inference was calculated using 80,000 iterations with a burn-in of 50,000 with parameters $\sigma_f^2 = 1000$, $\sigma_n^2 = 1e^{-5}$, $l = 1.50$, $N = 2000$ and $\omega = 0.001$. (B) shows ten trace plots of the intensity at time 0s using the new proposal mechanism.

Algorithm 4: Inference for GP intensity function.

Input:Spikes, ISI distribution, T , Number of iterations N_{iter} , burn in N_{burn} , \mathbf{t} ,

GP hyperparameters and ISI parameter

Output: N_{iter} samples of intensity \mathbf{x} Set initial values $\mathbf{x}^{(0)}$;**for** i in 1 to $(N_{\text{iter}} + N_{\text{burn}})$ **do**
 Propose \mathbf{x}_{can} and calculate acceptance probability using under-relaxed
 method, update $\mathbf{x}^{(i)}$;
if $i \% 1000 == 0$ **then**
for j in 1 to 50 **do**

 Use data-driven method at the start with f_{min} , and data-driven
 method at the end with f_{min} to update $\mathbf{x}^{(i)}$;
if *No proposals using f_{min} are accepted* **then**
for j in 1 to 50 **do**

 Use data-driven method at the start with f_{cur} , and data-driven
 method at the end with f_{cur} to update $\mathbf{x}^{(i)}$;
Return: \mathbf{x} ;

We now apply the new proposal mechanism for the GP prior to the Ca^{2+} spike data used in Figures 3.4 and 3.5. In Figure 3.11(A) we show the posterior mean and 95% credible region by the black line and red shaded region respectively. The inference was calculated using 80,000 iterations with an initial burn-in of 50,000. We see that the intensity at 0s and 20s is near zero. To check that this posterior distribution does not correspond to a fortunate region where the intensity is small at time 0s and 20s we look at the trace plots of the intensity at time 0s over 300,000 iterations. This is shown in Figure 3.11(B), where we have ten trace plots for ten different initial random seeds. We find that after the initial transients the intensity quickly converges towards zero intensity — within 10,000 iterations. Occasionally, we find small upticks in the intensity at time 0s, at for example iteration 75,000 or 290,000. These correspond to the under-relaxed method accepting intensity functions

with larger intensity at time 0s and then the data-driven proposal quickly reducing the intensity close to zero. Therefore, our data-driven proposal mechanism aids the under-relaxing method before the first spike time and after the last spike time.

In summary, we found that the under-relaxed method often gets stuck in regions of the posterior distribution with high intensity before the first spike time and after the last spike time, which the likelihood does not support. To help avoid the MCMC from getting stuck in these regions we created a bespoke data-driven method to work in tandem with the under-relaxed method to improve the mixing of our MCMC.

3.3 Estimating the Length Scale of a GP prior

In this section we explain the difficulty of inferring the length scale of the GP prior. In particular, we find that it is only possible to infer the length scale when provided with several spike sequences. To mitigate this issue when fitting the ISI model to individual Ca^{2+} spike sequences we use the intensity function inferred using a PWC prior to provide an appropriate length scale.

Imperative to inferring the intensity function of our model with a GP prior is the length scale we use in the square exponential covariance function. The length scale informs the shape the intensity function takes, where the smaller we choose l the quicker the intensity is allowed to vary over time.

To illustrate the effect that l has on the inferred intensity function we fit an inhomogeneous Gamma ISI distribution with a GP prior to a single spike sequence shown by the black ticks in Figure 3.12. The Ca^{2+} spike sequence comes from a HEK293 cell in a step change experiment over 6766s, where the cell is initially challenged with $20\mu\text{M}$ carbachol before the stimulus is exchanged at 3434s for $50\mu\text{M}$ carbachol. We fit the model for six different length scales varying from 169 to 6766. The mean of the posterior intensity function distribution for each choice of l is shown in Figure 3.12. We see that the smallest l gives rise to the intensity function that varies the quickest over time. Moreover, when $l = 169$ the intensity function often peaks at the locations of individual spikes and decreases between spike times — for

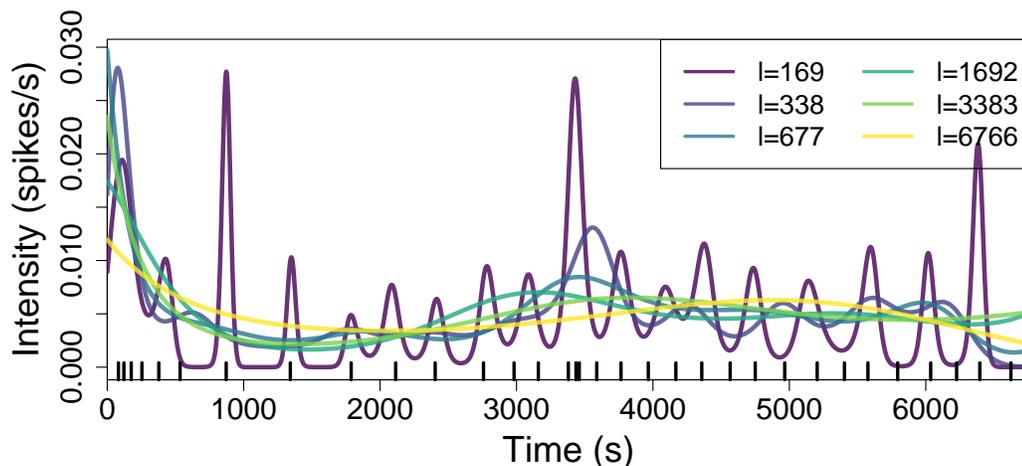


Figure 3.12: The mode of the posterior intensity function distribution fitted to a single spike sequence — shown by the black ticks. The model used was an inhomogeneous Gamma ISI distribution with a GP prior, where the length scale of the GP varies from 169 to 6766 — as shown in the legend. The inference was calculated over 20,000 iterations after a burn-in of 20,000, the parameters used were: $N = 2000$, $\sigma_h^2 = 0.5$, $\sigma_f^2 = 1000$, $\sigma_n^2 = 10^{-5}$, $w = 0.01$. The prior for the ISI parameter was an exponential with rate 0.01.

example in [800s, 3300s]. As l increases the intensity function no longer peaks at spike times. Instead, the intensity function requires more time to fluctuate. For example, when $l = 3383$ the intensity function gradually increases in [800s, 3300s]. Therefore, the choice of l can lead to posterior intensity functions with substantially different features.

We also infer the ISI parameter γ . The mean of the posterior distribution of γ was 13.49, 8.22, 7.96, 6.60, 5.93 and 4.11 for l equal to 169 through to 6766, respectively. We find that the larger the length scale the smaller the inferred γ . Note that a larger γ implies the variance of the ISI distribution is smaller. This suggests that a smaller length scale allows more of the variability of the spike times to be accounted by the intensity function. Indeed, we expect this behaviour because if the intensity function

cannot vary quickly enough to encompass the changing spiking rate, then we require a larger variance in the ISI distribution.

We have seen that the length scale can have a large affect on the inferred ISI distribution, affecting both the intensity function and ISI parameter. Since we aim to use our model to help inform on mechanisms of Ca^{2+} signalling — such as using the intensity function to examine how a cell's Ca^{2+} oscillations vary over time — it is vital to find the optimal length scale.

Theoretically, it is possible to infer the length scale from the spiking data we use to fit our model. One method is to use Bayes' theorem to obtain the marginal for l and a RW-Metropolis algorithm to sample the posterior of the length scale given spiking data, as described in Section 2.4.3.2.

However, in certain instances it is not possible to infer l . For example, suppose our spike sequence consists of a solitary spike. From this single spike it is not possible to determine if we have a small constant intensity throughout the entire experiment, or if the spiking intensity is zero everywhere except for a small region surrounding the spike time, where the intensity would be high. Moreover, it is possible to create two intensity functions that have the same likelihood but whose length scales differ significantly. Note that the likelihood of a spike sequence depends on the intensity function only through its value at spike times and the integral of the intensity function between spike times. For example, recall that the likelihood of a Ca^{2+} spike sequence \mathbf{y} coming from an inhomogeneous Gamma ISI distribution with intensity $x(t)$ and ISI parameter γ is

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

We see that $\pi(\mathbf{y}|x, \gamma)$ only interacts with the intensity function at spike times $x(y_i)$ and the integral between spike times $X(y_{i-1}, y_i)$, for $i = 1, \dots, N$. Therefore, we can create two intensity functions with the same likelihood provided that the intensity function at spike times and the integral of the intensity function between spike times are equal. This is illustrated in Figure 3.13, where we demonstrate two such intensity

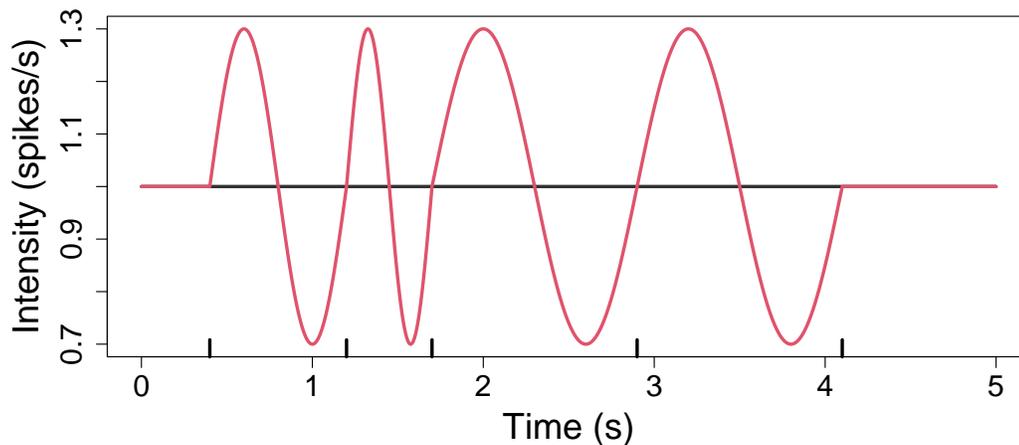


Figure 3.13: Two intensity functions which give rise to the same likelihood with the spike sequence shown by the black ticks.

functions. The length scale of the intensity functions are 42 and 0.195 for the black and red curves, respectively. This shows that with limited spiking data it is not possible to infer a length scale.

In this section we explore, via an example, the number of spikes and spike sequences required to infer the length scale directly. Furthermore, we describe an approach to choose an appropriate length scale when there is not enough information in the spiking data.

3.3.1 Estimating l with multiple spike sequences

The more spike sequences we use to fit our model the more information about the underlying ISI distribution we have. Therefore, using multiple spike sequences leads to additional detail about the shape of the intensity function. To explore if multiple spike sequences can accurately infer the shape of the intensity function we create a surrogate dataset from a known intensity function. We then fit an inhomogeneous Gamma ISI distribution with a GP prior — comprised of the intensity function, ISI

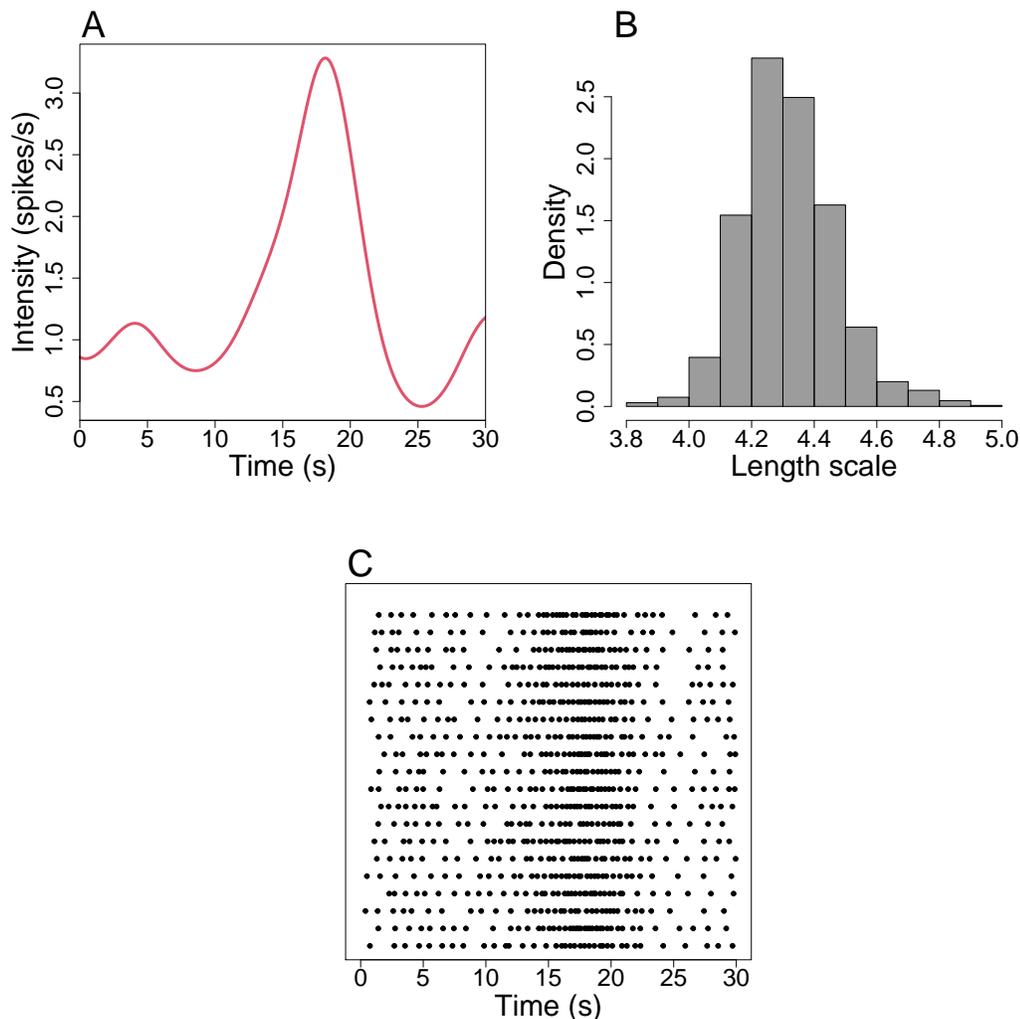


Figure 3.14: (A) The intensity function used to create surrogate spike sequences, drawn from a GP with mean 0 and square exponential covariance with parameters $l = 3$, $\sigma_f^2 = 1$ and $\sigma_n^2 = 10^{-5}$. (B) Histogram of posterior samples of l inferred from the intensity function given in (A). (C) 20 simulated spike sequences generated from an inhomogeneous Gamma ISI distribution with $\gamma = 7$ and intensity function given by (A).

parameter and length scale — using 2, 5, 10 and 20 spike sequences.

To create a surrogate dataset we first draw an intensity function on $[0s, 30s]$

by sampling from a GP where we take the hyperparameters to be $l = 3$, $\sigma_f^2 = 1$ and $\sigma_n^2 = 10^{-5}$. The drawn function is shown in Figure 3.14(A). Due to sampling variability the best length scale to describe the drawn function is not necessarily $l = 3$. Therefore, to find the best length scale we use Bayes' theorem to infer l from our drawn intensity function, where the prior for l is an Exponential distribution with rate 0.01. We use a RW-Metropolis algorithm with a Gaussian proposal distribution with variance of 1. We draw 5000 samples from the posterior distribution after an initial burn-in period of 5000. A histogram of the posterior is shown in Figure 3.14(B). We find that the 95% confidence interval of the length scale for the drawn intensity function is in $[4, 4.8]$.

We create a dataset of 20 spike sequences obtained from the Gamma ISI distribution with $\gamma = 7$ and intensity function given in Figure 3.14(A). We fit an inhomogeneous Gamma ISI distribution with a GP prior to 1, 2, 5, 10 and 20 spike sequences amalgamated from the dataset. The spike sequences are shown in Figure 3.14(C) where if we fit the model using Y spike sequences this consists of the bottom Y sequences shown. For each, the prior for γ and l are Exponential distributions with rate 0.01. We calculate 20,000 iterations after an initial burn-in of 20,000. The inference parameters used were: $\omega = 0.001$, $\sigma_f^2 = 1000$, $\sigma_n^2 = 10^{-5}$, $\sigma_h^2 = 0.5$ and $\sigma_l^2 = 0.5$.

In Figure 3.15 we illustrate the posterior ISI distribution via trace plots for l and γ and the mean of the posterior intensity function distribution. The results for fitting the model with 1, 2, 5, 10 and 20 spike sequences are shown in different colours — ranging from purple to yellow — as shown by the key at the top of the figure.

Consider the traces of γ shown in Figure 3.15(A). We find that with additional spike sequences we have stronger confidence in the estimate of γ . For example the 95% credible interval for γ is $[3.25, 12.62]$ and $[6.58, 7.06]$ when we use 1 and 20 spike sequences, respectively. We find that each of the 95% credible intervals contain $\gamma = 7$, the ISI parameter used to generate the spike sequences. The mean γ is 7.84, 7.71, 7.18, 6.81 and 6.82 when using 1, 2, 5, 10 and 20 spike sequences, respectively. Hence, with fewer spike sequences the mean of the inferred γ is slightly larger.

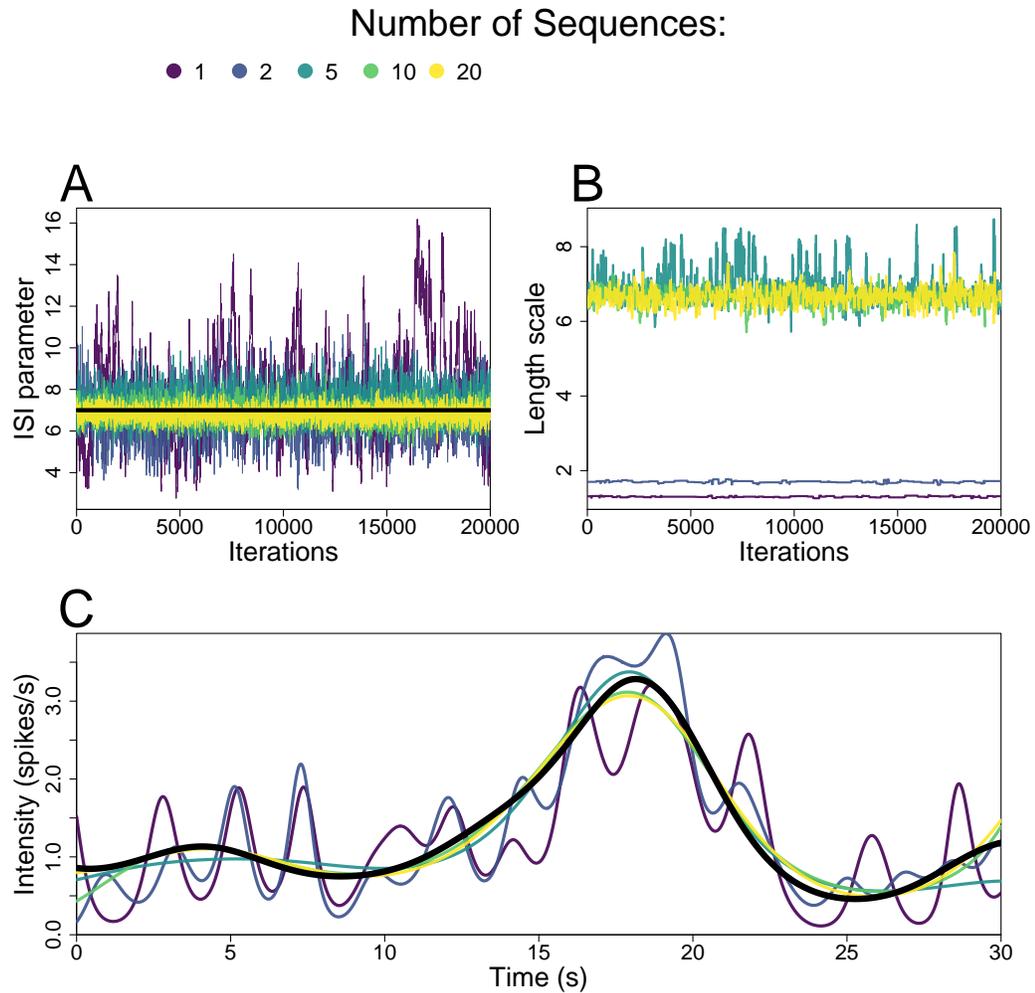


Figure 3.15: Visualisation of the posterior ISI distribution when using 1, 2, 5, 10 and 20 spike sequences to infer an inhomogeneous Gamma ISI distribution with a GP prior. (A-B) Trace plots of the ISI parameter and length scale, respectively. (C) The mean of the posterior intensity function distribution. The black curves in (A) and (C) coincide with the ISI parameter and intensity function used to generate the spike sequences.

In Figure 3.15(B) we show traces of the inferred length scale. We see that none of the traces converge onto the length scale of the intensity function used to create the spike sequences — in the region $[4, 4.8]$. However, we do find that given at least 5 spike sequences the length scale converges towards 6.6. When we used either 1 or 2 spike sequences the inferred length scale is considerably smaller than the models using 5, 10 or 20 spike sequences — the mean l is 1.30 and 1.70 when using 1 and 2 spike sequences, respectively.

The effect of the inferred l on the intensity function can be seen in Figure 3.15(C), which shows the posterior mean of the intensity function. We see that smaller l corresponds to a posterior mean that fluctuates in a shorter time frame. Moreover, the mean posterior intensity function from fitting either one or two spike sequences does not mirror the function used to simulate the spike sequences. Instead, they fluctuate around the true intensity function. Whereas, when we increase the number of sequence to 5, the posterior mean closely mirrors the true intensity, which only improves as the number of sequences increases to 10 and 20. Therefore, although the length scale does not align with the length scale of the true intensity, the mean of the posterior does closely resemble the true intensity.

In this example, to infer the intensity function such that shape is similar to the true intensity we require at least 5 spike sequences. When we inferred from one or two spike sequences the posterior intensity function has a length scale that is too small, and therefore changes of the intensity occurs quicker than the intensity function used to create the spike sequence. Therefore, in this example, there is not enough information from one or two spike sequences to infer the shape of the intensity function accurately.

In general, we find that with limited data it is difficult to infer l directly from the spike sequences. The number of sequences required could vary substantially since the information available in a single spike sequence depends on the number of spikes in the sequence. For example, if the average spike sequence contained only two spikes then we would require far more spike sequences to infer the correct shape of the intensity function compared to a dataset where every spike sequence contained 50

spike sequences. Therefore, it is difficult to determine whether a dataset contains enough information such that we can directly infer the shape of the intensity function from the spike sequences.

In regards to the Ca^{2+} datasets we consider each contains at least 15 spike sequences where the average number of spikes in each sequence exceeds 18 for each dataset. Therefore, if we want to fit the ISI model to an entire dataset, there is enough information to infer the shape of the intensity function. However, we also aim to infer ISI models onto individual spike sequences, since cells respond to stimulus in a heterogeneous manner. Moreover, in vivo each cell experiences a unique stimulus, therefore it is inappropriate to assume the dynamics are the same for each cell. In this case there is often not enough information to directly infer the length scale. Therefore, it is imperative to create a robust method to estimate l when a spike sequence alone does not possess enough information to accurately infer the shape of the intensity function.

3.3.2 Estimating l with limited information

We have found that given limited spike data it is difficult to infer the length scale directly. Therefore, we need to decide which values we expect the length scale to take. For example, returning to Figure 3.12, do we expect an intensity function with peaks centered at spike times — as for $l = 169$ — or an intensity that changes more gradually — for example $l = 677$? Since the cell is subject to a step change experiment we would expect a gradual change in intensity rather than an intensity that varies about each spike time. Therefore $l = 677$ is preferred to $l = 169$.

Broadly speaking, we want a length scale that is small enough to pick out all the details in the intensity function but not too small that the intensity ‘wobbles’ when it should remain flat. One advantage of approximating l is that a small change in the length scale only makes a small difference in the shape of the sampled intensity functions. For example, from Figures 3.14 and 3.15 we found that the true intensity function with l in $[4, 4.8]$ was closely mirrored by the inferred intensity function with l in $[6.58, 7.06]$. Therefore, we do not need to be unduly precise on the length scale

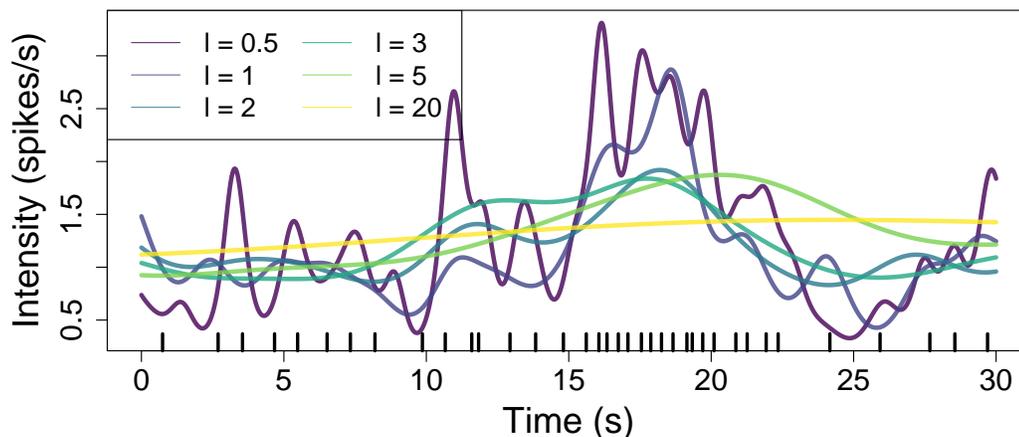


Figure 3.16: Examples of the mean of the posterior intensity function where we fit an inhomogeneous Gamma ISI distribution with a GP prior to the spike sequence — shown by the black ticks — whilst varying the length scale. The spike sequence is taken from Figure 3.14. We calculate 20000 iterations after an initial burn-in of 20000. The inference parameters used were: $\omega = 0.001$, $\sigma_f^2 = 1000$, $\sigma_n^2 = 10^{-5}$ and $\sigma_h^2 = 0.5$. The length scale used for each model is shown in the legend.

we choose. Choosing a length scale that is too small will lead to an intensity function that varies too quickly. This leads to an intensity function that will fluctuate about the ideal intensity function. On the other hand, if we choose a length scale too large then the intensity function cannot capture the features of the spike sequence because it takes too long for the intensity function to vary. We prefer a smaller length scale to a larger one since the intensity will fluctuate around true intensity function whereas a larger intensity cannot capture the shape of the true intensity function.

To give context on the effect of l we fit inhomogeneous Gamma ISI distributions using several values of l to a single spike sequence from the simulated data given in Figure 3.14. In Figure 3.16 we plot the mean of the posterior intensity function distribution. We see that the larger length scales do not pick up features of the spike times. For example when $l = 20$, the intensity slowly increases over the entire domain,

which does not capture the increased spiking rate in [15s, 20s]. Whereas, smaller length scales — such as $l = 0.5$ or $l = 1$ — have a larger intensity in this region. Examining the posterior intensity function for $l = 0.5$ we find that it fluctuates in regions where we would expect the intensity to remain flat, such as in [3s, 8s]. Therefore, Figure 3.16 shows the caveats of both large and small l . We also observe that a smaller length scale is preferable because the inferred intensity varies about the underlying rate.

Given a single spike sequence it is not obvious how to select l . To help inform on an appropriate length scale we fit the model with a PWC prior, and use the posterior mean of the intensity function distribution \mathbf{x}_{pwc} to inform our choice of l . The PWC prior does not mitigate the fact that intensity functions with varying shape can have the same likelihood. However, the structure of the prior contains more information compared to the GP prior. The frequency with which the intensity function can vary is controlled in the prior distribution by the number of change points. The more change points you expect the more the function can vary. Therefore, by limiting the number of change points it is less likely for the intensity function to contain peaks of intensity around spike times. If we infer l directly from \mathbf{x}_{pwc} the length scale would be too small because of the large change of intensity at change points. Moreover, this would rely upon inferring a smooth function to a PWC function. Therefore, to create a good estimate of l we smooth \mathbf{x}_{pwc} and only infer l on a wide time grid. For example, if \mathbf{x}_{pwc} is defined on a time step of 0.01s to infer l we use a coarser grid with a time step of 0.5s.

To estimate l we begin by fitting the model with a PWC prior to obtain \mathbf{x}_{pwc} , defined on the time index with N steps. Next we smooth \mathbf{x}_{pwc} . There exists several methods to smooth a function. Here, we use local polynomial regression with automatic smoothing parameter selection [120], because it does not require the tuning of any smoothing parameters. We restrict the smoothed function to the uniform time grid with $M < N$ steps — where N is divisible by M — to get $\mathbf{x}_{\text{pwc}}^\dagger$. We estimate l by finding the MLE of the marginal of l given $\mathbf{x}_{\text{pwc}}^\dagger$.

An example of estimating l using a PWC prior is shown in Figure 3.17. We

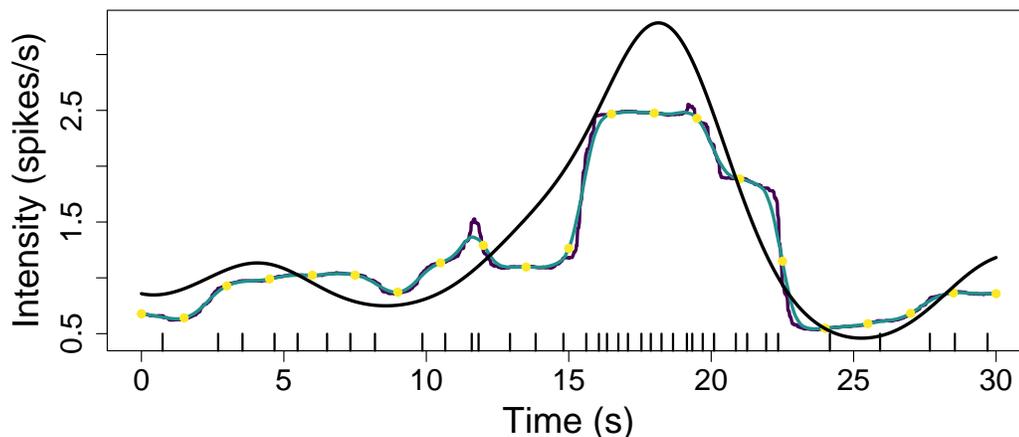


Figure 3.17: Illustration of obtaining l using the PWC prior. Begin by fitting an inhomogeneous Gamma ISI distribution with a PWC prior to the spike times shown as the black ticks. The parameters for the PWC prior were: $k_{\max} = 25$, $\kappa = \kappa_0 = 1$ and $\mu = 0.5$. The posterior mean of the intensity function \mathbf{x}_{pwc} is shown by the purple line. We smoothed \mathbf{x}_{pwc} by using local polynomial regression with automatic smoothing parameter selection — shown by the blue line. We estimate l by finding the MLE of l using a subset of the smoothed \mathbf{x}_{pwc} — shown by the yellow dots. The intensity function used to generate the spike sequence is shown by the black line.

use the same spike sequence from Figures 3.14 and 3.16. We begin by fitting an inhomogeneous Gamma ISI distribution with a PWC prior, whose mean intensity function \mathbf{x}_{pwc} is shown by the purple line. We then smooth \mathbf{x}_{pwc} which is shown by the blue line. We use a subset of the smoothed function — shown as the 20 yellow points — as the input to infer the length scale and we obtain $l = 2.74$.

The choice of M impacts the estimate for l , where the smaller we choose M the larger the estimate for l . This is because a sparser partition contains less details of intensity function inferred using a PWC prior. For example, if we took M equal to 5, 10, 20, 40, 80 in the above example the estimated length scale would be 12.97, 5.96, 2.74, 1.50, 1.18, respectively. As previously stated, we err on the side of using a

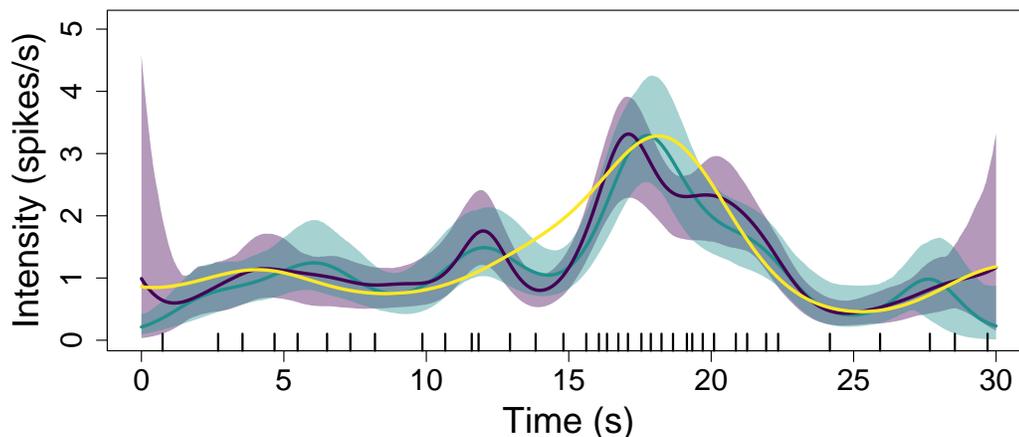


Figure 3.18: The posterior mean and 95% credible interval for the intensity function fitted using an inhomogeneous Gamma ISI distribution with a GP prior. We use blue to represent the fixed length scale $l = 2.74$ and purple is used when we draw l from a $\text{Gamma}(2.74 \times 8, 8)$ in each iteration. The inference parameters are identical to those used in Figure 3.16. The spikes used to fit the model are shown by the black ticks. The yellow line represents the intensity function used to generate the spike times.

smaller length scale therefore we often use M in the range of 20 – 40.

We can use the estimated l as a fixed length scale for our MCMC hyperparameter. Or, to allow more flexibility in the length scale, we can center a distribution about l and draw from this distribution each time we want to draw from the GP prior. This is shown in Figure 3.18, where l is fixed at 2.74 and given the prior $\text{Gamma}(2.74 \times 8, 8)$. This can be compared to Figure 3.16 where we used a variety of fixed length scales. We find that using the PWC to inform on l chooses a good length scale that is both small enough to capture features over the experiment’s length and large enough that the inferred intensity function does not oscillate between individual spike times.

Recall that the intensity function used to generate the spike sequence had l in the region $[4, 4.8]$ and when we fit 20 sequences simultaneously the inferred l was in $[6.17, 7.13]$. Therefore, estimating l via a PWC prior gives smaller l compared to the

model used to generate the spike sequences or the inferred l from 20 spike sequences. However, it is an improvement over inferring l directly from the single spike sequence where we obtained l in [1.26, 1.33], as the estimated l is approximately twice as large. We find that this approach leads to a desirable choice of l .

3.3.3 Summary

In this section we expressed the importance of the length scale to our model, and how different length scales can lead to drastically different posterior intensity functions. We found that we can infer l if we fit the model to several spike sequences collectively. However, with limited spiking data it may not be possible to directly infer l . In this case, we instead use the posterior intensity function from fitting the same model with a PWC prior to estimate l . We can either use the estimated l as a fixed parameter in the GP prior or use it to inform the prior on l . When fitting ISI distributions with a GP prior to a single Ca^{2+} spike sequence we use the PWC prior to estimate l and use this as a fixed hyper-parameter of the covariance function.

3.4 Speed of MCMC algorithms

In this section, we investigate the speed of our MCMC algorithms. We find that it is computationally demanding to sample the posterior distribution when using the GP prior. For example updating the length scale requires the generation and multiplication of large covariance matrices. To resolve this issue we use two approaches depending on whether we fit the length scale. When fitting the ISI model using several spike sequences we infer the length scale, and use *Mean Projection Approximation* (MPA) to improve the speed. Whereas, if we fit the ISI with a single spike sequence we fix the length scale — or fix its prior distribution —. In this case, we improve the speed by simulating from the GP prior using the spectral representation of the GP.

In this section, we investigate the speed of our MCMC algorithms. Namely, the algorithms used for the PWC prior and GP prior for the intensity function. It is important for the algorithms to be quick because they need to be applied repeatedly

for the different ISI distributions and for each Ca^{2+} spike sequence. Throughout this section we will give examples for the amount of time taken to perform MCMC algorithms and their subcomponents, all computations are computed on a Dell R720 rackmount server with: 2 x Intel(R) Xeon(R) CPU E5-2690 @ 2.90GHz each with 8 cores and 16 threads, 256Gb RAM and 2 x 2Tb SATA disks in RAID-1 mirror.

Recall for the PWC prior we use a RJMCMC algorithm to update the intensity function. The RJMCMC is comprised of three different types of update: a birth of a new change point, a death of a change point and an update to a height and change point location. For the GP prior we utilise an under-relaxed method to propose new intensity functions. This method creates a candidate function by combining the current intensity function with a draw from the prior — a high dimension multivariate Normal (MVN) distribution. The advantage of this approach is the acceptance probability simplifies to the ratio of likelihoods. A common issue with GPs is the computational demand of sampling from its distribution when the dimension is large [121, 122, 123]. We often use the recording rate of the Ca^{2+} experiments as our time-index which leads to a discretisation with size in the range of 2000 to 4000. Moreover, we often need to infer the length scale of the GP. We use a RW-Metropolis algorithm to propose candidate values. For each candidate length scale we need to calculate the corresponding covariance matrix which is computationally demanding when the dimension of the intensity function is large.

To visualise the limiting factor in the speed of our MCMC algorithms we record the time taken to compute 1000 iterations — after an initial burn in of 10000 iterations — using both a GP and PWC prior. We fit an inhomogeneous Gamma ISI distribution given a spike sequence with 34 spikes. The number of steps used for the GP prior was $N = 2000$. This was chosen so that the discretisation was similar to the recording rate of experimental Ca^{2+} spike sequences. We also record the time taken in each sub-compartment of the MCMC algorithms. In particular, for the PWC prior we record the time taken for each of the updates to the intensity function via the RJMCMC: a birth, a death or a change in height and partition, in addition to the time taken updating the ISI parameter. For the GP prior we assume the length scale

PWC				GP		
Birth	Death	Other	ISI	\mathbf{x}	ISI	l
0.00144s	0.00143s	0.00274s	0.0015s	0.0157s	0.0010s	11.03s

Table 3.4: The average time taken to compute a single MCMC loop using either a GP prior or a PWC prior over 1000 iterations. We fitted an inhomogeneous Gamma ISI distribution to a single spike sequence with 34 spikes. The ISI parameter had an exponential prior with rate 0.001. The inference parameters for the GP prior were: $N = 2000$, $w = 0.001$, $\sigma_h^2 = 1$, $\sigma_f^2 = 1000$ and $\sigma_n^2 = 10^{-5}$. The prior for the length scale l was an exponential with rate 0.01, the initial value was 1.5 and $\sigma_l^2 = 0.25$. For the PWC prior the parameters were: $k_{\max} = 25$, $\lambda = 10$, $\kappa = 1$, $\mu = 2$, $\kappa_0 = 1$ and $\sigma_h^2 = 1$. The column denoted ‘Other’ represents an update of both a height and a change point.

is unknown and infer it. We record the time spent updating: the intensity function, the ISI parameter and the length scale. The results are shown in Table 3.4. We find that the average time taken for a single iteration using the PWC and GP prior are 0.003242s and 11.046s, respectively. The vast difference in time taken comes from the time-required to update the length scale. This can be split into 10.93s to setup the GP for the candidate length scale and 0.10s to calculate the acceptance probability. Therefore, the bottleneck of the computation comes from varying the length scale.

To understand why changing the length scale is computationally demanding recall that the GP prior has mean zero and a square exponential covariance which between two times t_1 and t_2 is defined as

$$k(t_1, t_2) = \sigma_f^2 \exp \left[-\frac{(t_1 - t_2)^2}{2l^2} \right] + \sigma_n^2 \delta(t_1, t_2), \quad (3.4)$$

where σ_f^2 is the signal variance, l is the length scale and σ_n^2 is the noise variance. Suppose the domain of the intensity function is $[0, T]$. We discretise our domain into a time set $\mathbf{t} = \{t_i\}_{i=0}^N$ where $t_i = iT/N$ and N controls the fineness of our discretisation. Therefore, on this domain, the GP prior is the MVN distribution with mean zero and covariance matrix $\Sigma = \{k(t_i, t_j)\}_{i,j \in \{0, \dots, N\}}$, i.e $\mathbf{x} \sim \mathcal{N}_{N+1}(\mathbf{0}, \Sigma)$. When updating

the intensity function we need to sample from this MVN distribution to create a candidate intensity function. Furthermore, updating l requires the creation of a new covariance matrix for the candidate length scale.

A common method for sampling from a MVN distribution is to transform samples from a standard normal distribution. This approach relies on finding a matrix \mathbf{A} such that $\mathbf{A}\mathbf{A}^T = \Sigma$ [124]. A commonly used method to find the matrix \mathbf{A} is by a Cholesky decomposition. There exists other techniques to obtain \mathbf{A} — such as an eigenvalue or single value decomposition — however a Cholesky decomposition is often the quickest, where the computational complexity of it is $\mathcal{O}(N^3)$. Moreover, calculating the probability density function of the MVN distribution requires the inverse and determinant of Σ . The fastest approach to calculate these utilises the Cholesky decomposition [125].

Therefore, when given a new length scale we need to create its corresponding covariance matrix Σ and the Cholesky decomposition $\text{Chol}(\Sigma)$. Both of which are computationally demanding operations. Once calculated we reuse $\text{Chol}(\Sigma)$ in sampling from the GP prior and calculating the acceptance probability for l .

Note that for a smooth intensity function we would set $\sigma_n^2 = 0$. However doing so can render the covariance matrix computationally singular. To mitigate this, we employ the widely used approach of adding a small amount of noise to the model — for example setting $\sigma_n^2 = 10^{-5}$ — which improves the conditioning of the matrix and has little affect on the drawn functions [126].

In section 3.2, we discussed inferring the length scale. We concluded that depending on the amount of spiking data we choose to either infer l directly or approximate l using the PWC prior. If we choose to infer l directly we found that the average time for a single iteration is 11.046s. To infer the ISI distribution would take approximately 24.5 days — equating to 200,000 iterations. Therefore, we require methods to mitigate the time taken if we infer l . If there are not enough spikes to directly infer the l then we calculate an approximation l_{PWC} via the PWC prior. We can use l_{PWC} as a fixed length scale. In this case, we only need to calculate $\text{Chol}(\Sigma)$ once for l_{PWC} . We can then reuse it each time we draw from the prior. An average loop of the MCMC

algorithm would take 0.0157s, if $N = 2000$. This is significantly faster than inferring l , and only 5-fold slower than using a PWC prior. However, a fixed value for l will limit the shape of the intensity function. Another approach would be to use l_{PWC} as a basis for a prior on l . For example $l \sim \text{Uniform}(l_{\text{PWC}} - 1, l_{\text{PWC}} + 1)$. In this case, we do not infer l , rather when we draw from the GP prior we first draw l from its prior distribution. The cost of this approach is still large, because each time we draw l from its prior we need to calculate $\text{Chol}(\Sigma)$ to be able to sample from the prior. Indeed, this would require an average 10.94s per iteration when $N = 2000$.

Therefore, we require methods to mitigate the time taken if we vary l . In this sections we propose two methods to efficiently alleviate this issue. The first method uses a projection scheme, where we draw from a MVN distribution with smaller dimension and then project this onto the desired dimension. This can be used if we include the length scale as a model parameter. The second method uses the spectral representation of a GP and a fast Fourier transform (FFT) to quickly generate samples on the targeted domain when l has a prior distribution.

3.4.1 The Mean Projection Approximation

We first look at improving the computational demand of our MCMC algorithm when we infer l from the spiking data. One approach to mitigate the time taken is to use a projection method to construct an approximation of the intensity function. A projection method creates a similar model on a smaller set of time points and performs exact inference on this new model. We shall use the *Mean Projection Approximation* (MPA). The MPA gives a GP prior distribution over a smaller set of time points and then projects it onto the full set of time points, which we use to calculate the likelihood of our spike sequences [122].

To calculate the likelihood of a spike sequence we require an original time set \mathbf{t} of size N large. For the proposal of new intensity functions we create a smaller time set $\bar{\mathbf{t}}$ of size $Q < N$. The set $\bar{\mathbf{t}}$ must cover enough of the domain such that our intensity function still maintains any interesting features. We choose to construct $\bar{\mathbf{t}}$ as a set of uniformly placed points from the original time set, including 0 and T . We denote the

intensity function over the original time set by \mathbf{x} and the intensity function over the smaller time set by $\bar{\mathbf{x}}$. We place a joint GP prior distribution on \mathbf{x} and $\bar{\mathbf{x}}$ such that

$$\begin{pmatrix} \mathbf{x} \\ \bar{\mathbf{x}} \end{pmatrix} \sim \mathcal{N} \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \Sigma_{\mathbf{t},\mathbf{t}} & \Sigma_{\mathbf{t},\bar{\mathbf{t}}} \\ \Sigma_{\bar{\mathbf{t}},\mathbf{t}} & \Sigma_{\bar{\mathbf{t}},\bar{\mathbf{t}}} \end{bmatrix} \right), \quad (3.5)$$

where $\Sigma_{\mathbf{t},\mathbf{t}}$ is the covariance matrix for the original time set, $\Sigma_{\bar{\mathbf{t}},\bar{\mathbf{t}}}$ is the covariance matrix for $\bar{\mathbf{t}}$ and $\Sigma_{\mathbf{t},\bar{\mathbf{t}}}$ is the covariance between the original and smaller time sets. The MPA uses the conditional distribution of a MVN, which gives the distribution of \mathbf{x} given $\bar{\mathbf{x}}$ by

$$\mathbf{x}|\bar{\mathbf{x}} = \mathcal{N}_{N+1} \left(\Sigma_{\mathbf{t},\bar{\mathbf{t}}}\Sigma_{\bar{\mathbf{t}},\bar{\mathbf{t}}}^{-1}\bar{\mathbf{x}}, \Sigma_{\mathbf{t},\mathbf{t}} - \Sigma_{\mathbf{t},\bar{\mathbf{t}}}\Sigma_{\bar{\mathbf{t}},\bar{\mathbf{t}}}^{-1}\Sigma_{\bar{\mathbf{t}},\mathbf{t}} \right). \quad (3.6)$$

This projects $\bar{\mathbf{x}}$ onto the original time set \mathbf{t} . However, if we draw from $\mathbf{x}|\bar{\mathbf{x}}$ to obtain \mathbf{x} this would take a similar amount of time as drawing \mathbf{x} directly, as we would be sampling from a MVN distribution with dimension $N + 1$. Therefore, we take the projection to be the mean of $\mathbf{x}|\bar{\mathbf{x}}$, namely

$$\mathbf{x} = \Sigma_{\mathbf{t},\bar{\mathbf{t}}}\Sigma_{\bar{\mathbf{t}},\bar{\mathbf{t}}}^{-1}\bar{\mathbf{x}}. \quad (3.7)$$

The computational complexity of (3.7) comes from calculating $\Sigma_{\bar{\mathbf{t}},\bar{\mathbf{t}}}^{-1}$ with $\mathcal{O}(Q^3)$ and generating $\Sigma_{\mathbf{t},\bar{\mathbf{t}}}$ with $\mathcal{O}(NQ)$.

Implementing this approach only requires minor changes to our MCMC algorithm. In particular, the under-relaxed proposal (3.1) and length scale acceptance probability (2.82) change by substituting \mathbf{x} with $\bar{\mathbf{x}}$ and Σ with $\Sigma_{\bar{\mathbf{t}},\bar{\mathbf{t}}}$. In the likelihood we replace \mathbf{x} with $\Sigma_{\mathbf{t},\bar{\mathbf{t}}}\Sigma_{\bar{\mathbf{t}},\bar{\mathbf{t}}}^{-1}\bar{\mathbf{x}}$. Recall that the logarithm of the intensity function follows a GP prior not the intensity function itself. Therefore the projection needs to be applied to the logarithm of the intensity function.

Figure 3.19 illustrates the MPA method showing \mathbf{x} and $\bar{\mathbf{x}}$ in the i th and $(i + 1)$ th iteration, where $N = 200$ and $Q = 20$. The red arrows describe the methods used to draw a candidate function and to calculate the intensity function used in the likelihood.

The improvement of speed depends on how fine we choose the smaller time grid, where we have a trade off in the time taken against the accuracy of the intensity

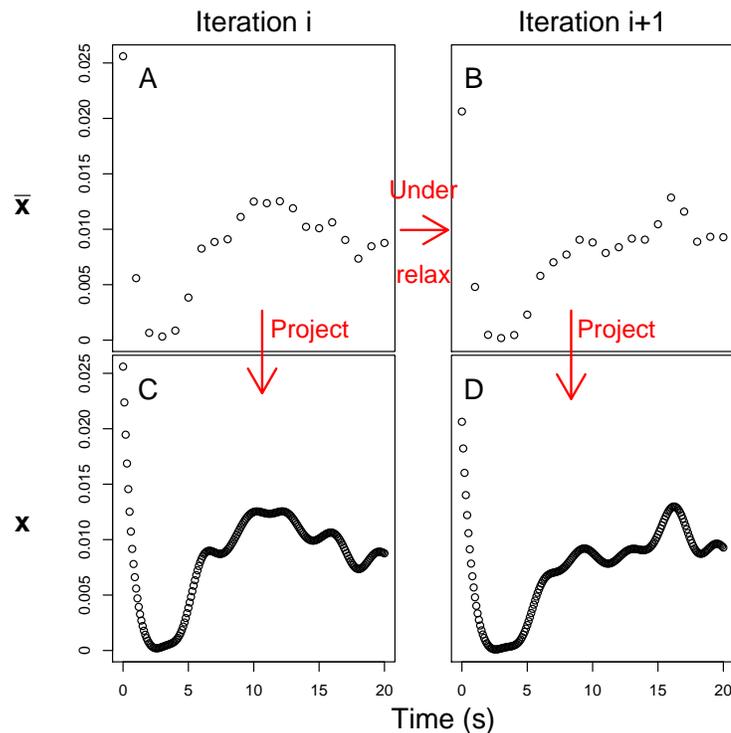


Figure 3.19: Illustration of MPA, showing how a smaller time index is used for proposing the new intensity function (A \rightarrow B) and how projection is used to create the intensity function on the desired time index (A \rightarrow C and B \rightarrow D). In this example we use $Q = 20$ and $N = 200$.

function. For example, returning to the example from Table 3.4 we found that the average time taken for a single iteration is 11.046s, when $N = 2000$. When we use the MPA we need to be careful defining the average time taken for a single MCMC loop. This is because if we accept the candidate l we need to recalculate $\Sigma_{\mathbf{t}, \bar{\mathbf{t}}}$. Therefore, when we use MPA with Q equal to 10, 20, 50 and 200. The average time for a single MCMC loop is 0.0633s (0.0038s), 0.1133s (0.0043s), 0.2705s (0.0131s) and 1.1748s (0.1571s), respectively. The first time and time given in the brackets correspond to the time taken if we accept and reject the candidate l , respectively. Hence, the smaller we choose Q , the larger the improvement in the time taken. However, the

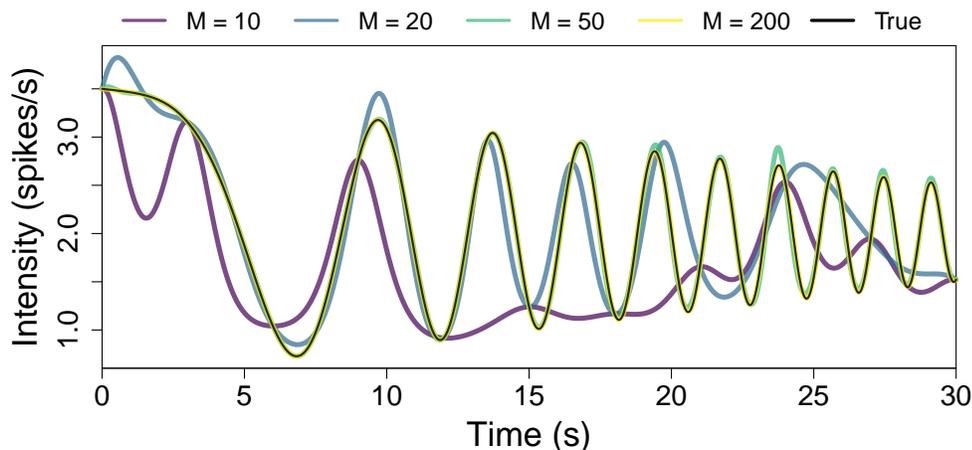


Figure 3.20: A known intensity function compared to its MPA approximation using a varying values of Q . The intensity function — shown in black — is defined by $x(t) = (1.5 - t/30) \cos(t^2/15) + 2$. The MPA is calculated for Q equal to 10, 20, 50 and 200, using the parameters $\sigma_f^2 = 1000$, $\sigma_n^2 = 10^{-5}$ and $l = 1$. The projected intensity function for each Q is shown by a coloured line described in the legend.

improvement comes at the cost of the accuracy of \mathbf{x} as the projection cannot contain the same level of detail as sampling directly from the GP prior distribution.

To visualise how the choice of Q impacts the intensity function we approximate a known function. We take the intensity function, defined on $[0\text{s}, 30\text{s}]$, to be $x(t) = (1.5 - t/30) \cos(t^2/15) + 2$. We chose this intensity function because as time progresses the function oscillates quicker. Therefore, we can see how different choices of Q affect the projected intensity function with different length scales. For a given choice of Q , we first restrict the intensity function to the time set $\{t_k\}_{k=0}^Q$ where $t_k = 30k/Q$. We then project the restricted intensity function via equation (3.7) where $N = 2000$. We do this for Q equal to 10, 20, 50 and 200. The resultant intensity functions are shown in Figure 3.20. We see that when $Q = 200$ to the eye there is little difference between the true intensity function and the projected intensity function. However, as we take Q smaller the difference between the projected intensity function and the truth

increases. When $Q = 10$ the projected intensity function differs greatly compared to the true intensity function. Therefore, the choice of Q is a trade off between the improvement in the computational time and the accuracy of the projected intensity functions.

Thus, we have found that the MPA approach reduces the computational demand of inferring the length scale from the spiking data. We have shown that this approach can lead up to a 200-fold improvement — when $N = 2000$ — of the time taken compared to the direct approach. However, this improvement in time comes at the cost of the accuracy of the intensity function, where the smaller we choose Q the fewer features the intensity function can exhibit.

3.4.2 Simulating using spectral representation

In this section, we utilise the spectral representation of our GP prior to improve the time taken in our MCMC algorithm when we assume that l has a given distribution. Therefore, each time we propose a candidate function we first draw l from its distribution and use the drawn l to sample from the GP prior.

So far we have defined a GP through its mean and covariance function. Note that the square exponential covariance function — see equation (3.4) — depends only on the time difference $\tau = t_1 - t_2$. As such we can write the covariance as a function of the time difference $k(\tau) = \sigma_f^2 \exp[-\tau^2/2l^2] + \delta_\tau$, where δ is the Dirac delta function. Recall that noise is added to the covariance function otherwise the covariance matrix is numerically singular. Since the spectral approach does not rely on the covariance matrix we can remove the noise and obtain $k(\tau) = \sigma_f^2 \exp[-\tau^2/2l^2]$.

We can remove the noise because we do not require σ_n^2 small to improve the conditioning of the covariance matrix. Therefore, our GP is a stationary stochastic process. This means that we can define the GP in terms of its spectral density

function $S(\omega)$ which is the Fourier transform of the covariance function

$$k(\tau) = \int_{-\infty}^{\infty} S(\omega) e^{i\omega\tau} d\omega, \quad (3.8)$$

$$S(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} k(\tau) e^{-i\omega\tau} d\tau. \quad (3.9)$$

For our square exponential covariance function the spectral density function is given by

$$S(\omega) = \sigma_f^2 (2\pi l^2)^{0.5} \exp[-2\pi^2 l^2 \omega^2] \quad [114]. \quad (3.10)$$

Following the approach used by Shinozuka and Deodatis [127] we can express the GP $f(t)$ in terms of its spectral density by

$$f(t) = \sqrt{2} \sum_{k=0}^{\infty} (2S(\omega_k) \Delta\omega)^{1/2} \cos(\omega_k t + \Phi_k). \quad (3.11)$$

where $\Delta\omega$ is the frequency step size which is taken to be small, $\omega_k = k\Delta\omega$ and Φ_k are independent random phase angles uniformly distributed in $[0, 2\pi]$. To obtain a sample $\bar{f}(t)$ from the spectral representation of the GP we approximate equation (3.11). We do this by restricting to the first N_ω terms and replace the random phase angles Φ_k with their realisations ϕ_k , thereby obtaining

$$\bar{f}(t) = \sqrt{2} \sum_{k=0}^{N_\omega-1} (2S(\omega_k) \Delta\omega)^{1/2} \cos(\omega_k t + \phi_k) \quad (3.12)$$

where

$$\Delta\omega = \omega_c / N_\omega. \quad (3.13)$$

The frequency step size $\Delta\omega$ depends on the cutoff ω_c where we assume the spectral density is zero for larger frequencies. N_ω can be viewed as the number of points we partition ω_c into. To find the value of the cutoff we use

$$\int_0^{\omega_c} S(\omega) d\omega = (1 - \epsilon) \int_0^{\infty} S(\omega) d\omega \quad (3.14)$$

where ϵ is small; we take $\epsilon = 0.001$.

To take advantage of FFT methods to sample the GP we express equation (3.12) by

$$\bar{f}(p\Delta t) = \text{Re} \left\{ \sum_{k=0}^{N_{\text{FFT}}-1} A_k e^{ikp2\pi/N_{\text{FFT}}} \right\}, \quad p = 0, 1, \dots, N_{\text{FFT}} - 1, \quad (3.15)$$

where

$$A_k = \sqrt{2} (2S(\omega_k)\Delta\omega)^{1/2} e^{i\phi_n}, \quad n = 0, 1, \dots, N_{\text{FFT}} - 1, \quad (3.16)$$

$$\Delta\omega = \omega_c/N_\omega \quad (3.17)$$

and Δt is the time step size, which must satisfy $\Delta t \leq \pi/\omega_c$. Moreover, we discretise time into N_{FFT} time points. The sample function $\bar{f}(t)$ is periodic with period T_0

$$T_0 = N_{\text{FFT}}\Delta t = 2\pi/\Delta\omega = 2\pi N_\omega/\omega_c \quad (3.18)$$

Therefore we see that the number of steps we partition time into (N_{FFT}) and the number of steps we partition the frequencies into (N_ω) are linked via equation (3.18). We can then apply FFT algorithms to equation (3.15) to sample from the GP prior. We employ the commonly used Cooley-Tukey FFT algorithm, details of it can be found in [128].

The main limitation of this approach is the length of the sampled function is fixed at T_0 , which is dependent on ω_c and N_ω . For example, if ω_c equals 2π rad/s for our choice of hyper-parameters and we choose $N_\omega = 128$ then the length of the sampled function is $2\pi N_\omega/\omega_c = 128\text{s}$. Suppose we require a step size of 0.1s then we would have to choose $N_{\text{FFT}} = 1280$ such that the condition $T_0 = N_{\text{FFT}}\Delta t$ is satisfied. Therefore, we find that the choice of N_{FFT} is informed by the step-size we require. Furthermore, often T_0 is not a number that is divisible by our choice of step size and since N_{FFT} is defined as an integer we must approximate our step size. For example if $T_0 = 128.38\text{s}$ and our step size is 0.1 then $N_{\text{FFT}} = 1283.8$. Rounding gives $N_{\text{FFT}} = 1284$ from which the step size used in the sampled function is $128.38/1284 = 0.09998\text{s}$. Another issue of having N_{FFT} informed by our step-size is that to take full advantage of the Cooley-Tukey FFT algorithm we require N_{FFT} to have multiple factors [129]. This is illustrated in Figure 3.21 where we compute the time taken to sample 100 draws from a GP where N_{FFT} varies between 3000 and

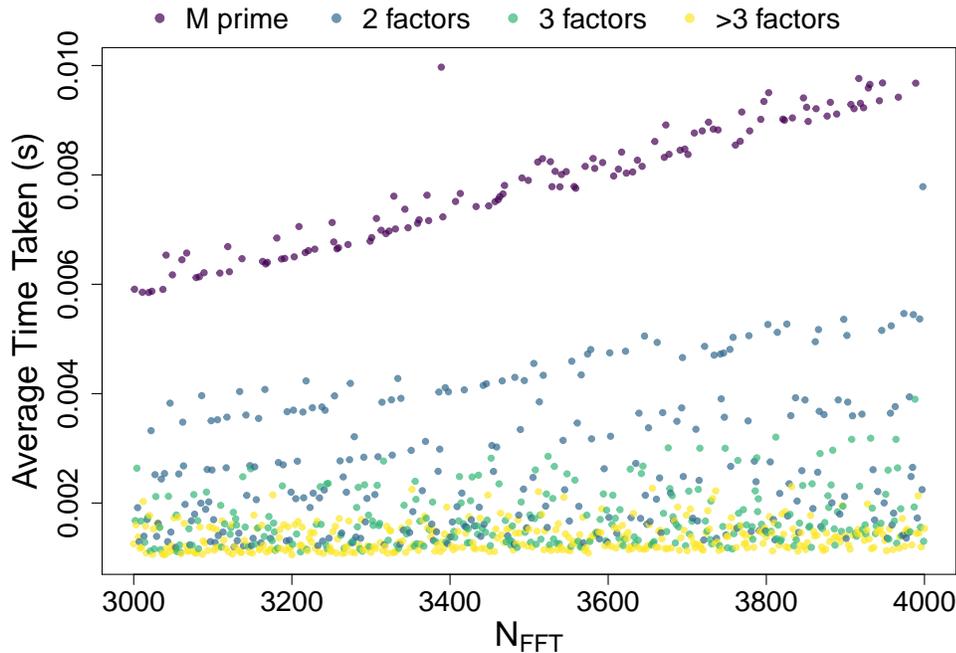


Figure 3.21: The average time taken to simulate an intensity function over 100 samples using the Cooley-Tukey FFT algorithm applied to the spectral representation of a GP against N_{FFT} . The GP has mean zero and square exponential hyper-parameters $\sigma_f^2 = 1000$, $\sigma_n^2 = 0$ and $l = 1.3$. The points are coloured depending on the number of factors N_{FFT} has, shown by the legend.

4000. We see that the speed to sample functions depends on the number of factors N_{FFT} has. Indeed, we see that it is 10-fold faster to draw samples when $N_{\text{FFT}} = 3492$ compared to $N_{\text{FFT}} = 3491$ — a prime.

To mitigate this issue we take advantage of the cutoff frequency to give values of N_{FFT} that allow for quick computation via FFT and an exact time step. Recall that ω_c is calculated by equation (3.14) where we took ϵ small. Therefore, increasing ω_c is equivalent to choosing a smaller ϵ .

Decreasing ϵ effects the simulated GP in two ways: the spectral cutoff increases and the frequency step size $\Delta\omega = \omega_c/N_\omega$ increases. Provided that we choose N_ω

large increasing ω_c has little affect on the simulated intensity function.

Suppose we want to sample an intensity function from a GP in $[0, T]$ with time step Δt . We begin by choosing N_ω and calculating ω_c given $\epsilon = 0.001$. We can then calculate N_{FFT} that gives us our desired time step by $N_{\text{FFT}} = 2\pi N_\omega / \omega_c \Delta t$. However, this N_{FFT} does not have to be an integer and its rounded value need not have many factors. Therefore we decrease N_{FFT} and increase ω_c while maintaining their product. We do this by choosing a new N_{FFT}^* taken to be a whole number smaller than N_{FFT} with at least 8 prime factors. We then choose new cutoff frequency by $\omega_c^* = 2\pi N_\omega / N_{\text{FFT}}^* \Delta t$. Therefore, using N_{FFT}^* and ω_c^* takes advantage of Cooley-Tukey FFT algorithm whilst using the exact time step we desire.

Recall that simulating a GP using this approach gives a sample function with fixed length, defined by $T_0 = 2\pi N_\omega / \omega_c$. Rather than choose N_ω such that T_0 is equal to the length of our intensity function, we choose N_ω large — often $N_\omega = 128$ — and restrict the sampled function to our desired length. This is to ensure the accuracy of the sampled function. For example, suppose we want to sample from the GP prior in $[0\text{s}, 40\text{s}]$ with a step size of 0.01s , where the covariance parameters are $\sigma_f^2 = 1000$ and $l = 1.3$. For this covariance the spectral cutoff is $\omega_c = 0.3915$. Taking $N_\omega = 128$ gives a sampled function with length $T_0 = 2\pi N_\omega / \omega_c = 314.88\text{s}$, and $N_{\text{FFT}} = 31488$. Therefore, to obtain the intensity function on $[0\text{s}, 40\text{s}]$ we restrict the sampled function to the first 4000 entries.

The length of the sampled function depends on the length scale. This is because the larger l is the smaller ω_c . For example, if we increase l from 1.3 to 4 in the above example ω_c decreases from 2.55 to 0.829 leading to a period of 969.6s. Thus, we would require $N_{\text{FFT}} = 96960$, and the length of the sampled function is approximately 3 times longer. With a larger N_{FFT} the Cooley-Tukey FFT algorithm takes longer to compute. Therefore, the larger the length scale the longer it takes to sample intensity functions.

In Figure 3.22 we show the average time taken over 100 samples of using the spectral approach on the region $[0\text{s}, 20\text{s}]$ with $\sigma_f^2 = 1000$ and length scale varying between 0.5 and 10, with time step 0.01s . We describe the length scale as a percent

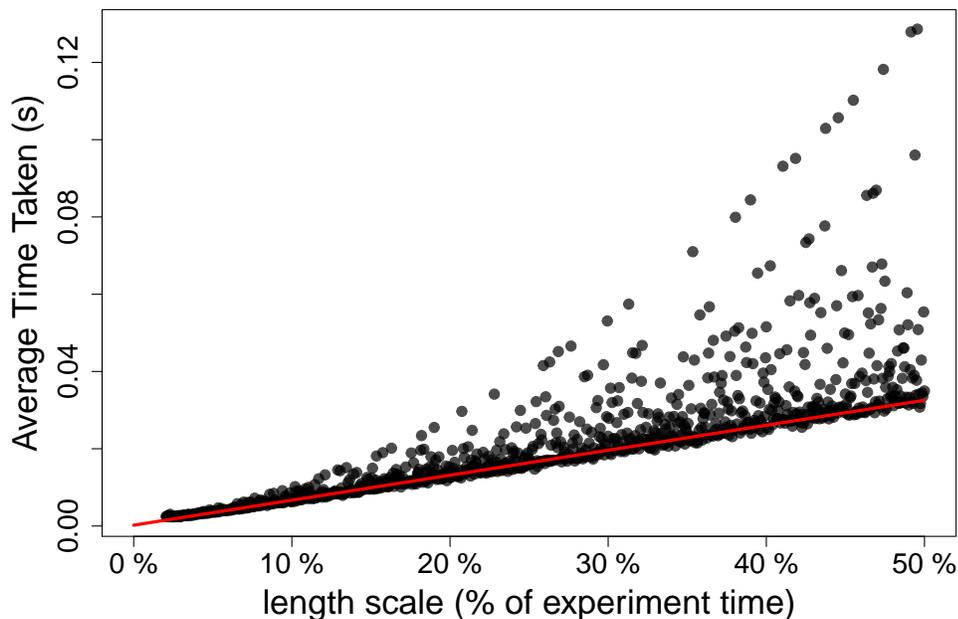


Figure 3.22: The average time taken over 100 samples to draw an intensity function with varying length scale. We sampled functions in $[0s, 20s]$ with a step size of $0.01s$ and $\sigma_f^2 = 1000$. The length scale is given in percentage of the experiment length, e.g $l = 4$ is given by a length scale of 20%.

of the experiment length. For example $l = 5$ is equivalent to 20%. This is because the length of the experiment varies for Ca^{2+} spike sequences. Thus, using a percentage gives an approach to compare length scales when the experiment length varies. Indeed, scaling doesn't affect the computational cost, for example $T = 20$ and $l = 2$ produces N_{FFT} which is the same if we used $T = 40$ and $l = 4$. We find a linear trend between the length scale and the average time taken to sample from the prior for the majority of the points, shown by the red line. However, as the length scale increases we find that for some values the time taken increases up to 6-fold compared to surrounding length scales. This is due to amount of factors N_{FFT} has, even after choosing a suitable value. Recall, that small changes in the length scale only has a minimal affect

on the sampled functions. Therefore, we restrict our choice of l to those near the linear trend. The inferred intensity function to Ca^{2+} spike sequences usually require a length scale in the region [5%, 15%] due to variation in spiking over time. Therefore, the average time taken to sample an intensity function is in [0.0035s, 0.0100s].

Although above we chose $N = 2000$ picking a larger N has little affect on the time required to sample an intensity function. This is because the spectral approach simulates a function with length N_{FFT} which is often far greater than N . For example, suppose we require a sample on [0s, 20s] with a step size of 0.01s, and our GP hyperparameters are $l = 2$ and $\sigma_f^2 = 1000$. Sampling from the GP via the spectral approach — where we take $N_\omega = 100$ — we find that the period of the simulated function is $T_0 = 378.56\text{s}$, and $N_{\text{FFT}} = 37856$. Therefore, increasing N up to 37856 will have no effect on the time taken to sample from the GP with the same step size.

If we were to require N larger than N_{FFT} we would have to increase N_ω which increases the period of the sampled function. For example, with the same parameters increasing N_ω to 128 leads to a sampled function with $T_0 = 484.8\text{s}$, and $N_{\text{FFT}} = 48480$.

We now compare the spectral approach to both the direct method and MPA approximation when l has a prior distribution. We do this for the spike sequence used in Table 3.4. For each method we calculate the time taken per iteration of the MCMC loop where we fit an inhomogeneous Gamma ISI distribution. The shared GP prior parameters were: $\omega = 0.01$, $\sigma_h^2 = 0.25$, $\sigma_f^2 = 1000$ and $N = 2000$. In the direct approach and MPA $\sigma_n^2 = 10^{-5}$, whereas for the spectral method $\sigma_n^2 = 0$. The prior for l was a uniform distribution between 1 and 5. The average time taken over 100 iterations is shown in Table 3.5 for each method, where for MPA we use $M = 10, 20, 50, 200$.

Approach	Direct	MPA ($M =$)				Spectral
		10	20	50	200	
Avg. time taken	10.87s	0.0658s	0.1106s	0.2701s	1.163s	0.0101s

Table 3.5: Comparison of the average time taken for a single MCMC iteration using varying approaches when l has a prior distribution.

We find that the spectral approach is substantially quicker than the direct approach and MPA, even for small M . This vast increase in speed comes from removing the requirement of computing covariance matrices and their decompositions. Thus, the best method to use if l is given a prior distribution is the spectral approach.

3.4.3 Summary

To summarise, we have considered two different approaches to resolve the computational demand of our MCMC algorithms. We found that the MPA is best used when we fit the length scale at the same time as the intensity function and ISI parameter. We use this approach if we fit our model to several spike sequences simultaneously. In particular, when we fit our model to a dataset of Ca^{2+} spike sequences obtained from cells exposed to the same stimulus. If we fit the ISI distribution to a single spike sequence there is often not enough information to infer l . Therefore for single spike sequences we choose to give l a prior distribution and we use the spectral approach because it is the fastest in this case. We base the prior on the estimate of l obtained using the PWC prior l_{PWC} , often by setting the mean of the distribution to be l_{PWC} .

CHAPTER 4

Refractory Period

The refractory period is the minimum amount of time after a Ca^{2+} spike before another Ca^{2+} spike can be triggered. A large number of experiments have shown the existence of a refractory period and that the refractory period is cell type specific [24, 61, 103, 130]. Therefore, it would be beneficial to extend our ISI model to include the refractory period as a model parameter.

In this chapter, we extend the current modelling framework to include the refractory period and we develop an MCMC algorithm to sample the posterior distribution of the refractory period. We then explore the advantages and disadvantages of including the refractory period as a model parameter. In particular, we investigate how the posterior distribution changes the effect of the refractory period on the other model parameters: the intensity function $\lambda(t)$ and ISI parameter θ . We also consider if a Ca^{2+} spike sequence contains enough information to accurately estimate the refractory period.

4.1 Including the refractory period in the ISI model

To add the refractory period into the ISI model we begin with an inhomogeneous ISI distribution, such as those defined in Section 3.1. Without loss of generality, assume the inhomogeneous ISI distribution is given by $p(s, t|x, \theta)$ where s is the time of the previous spike, t is the time of the current spike, x is the intensity function and θ is the ISI parameter. To construct an inhomogeneous ISI model with a refractory period we add the parameter T_{\min} to represent the refractory period. With the inclusion of T_{\min} the probability density function of the inhomogeneous ISI distribution becomes

$$p_{\text{refract}}(t, s|x, \theta, T_{\min}) = \begin{cases} 0 & \text{if } t - s < T_{\min}, \\ p(t, s + T_{\min}|x, \theta) & \text{otherwise.} \end{cases} \quad (4.1)$$

For example the probability density function of the inhomogeneous Gamma ISI distribution with refractory period is

$$p_{\text{refract}}(t, s|x, \gamma, T_{\min}) = \begin{cases} 0 & \text{if } t - s < T_{\min}, \\ \frac{\gamma x(t)}{\Gamma(\gamma)} [\gamma X(s + T_{\min}, t)]^{\gamma-1} e^{-\gamma X(s+T_{\min}, t)} & \text{otherwise,} \end{cases} \quad (4.2)$$

where

$$X(s, t) = \int_s^t x(u) du. \quad (4.3)$$

The extension of the inhomogeneous ISI distribution with the refractory period contains the original ISI distribution as a special case when $T_{\min} = 0$. Constructing an ISI distribution with the refractory period can be viewed as shifting the inhomogeneous ISI distribution in the time axis by T_{\min} . To illustrate this, we consider the Gamma ISI distribution with the refractory period where the intensity function $x = 1$ is fixed. We do this because the model becomes stationary and we can plot the probability density function of the ISI distribution. This is illustrated in Figure 4.1 where we show the ISI distribution when $T_{\min} = 0\text{s}$, 0.5s and 1s . We see that an increase in T_{\min} shifts the ISI distribution to the right.

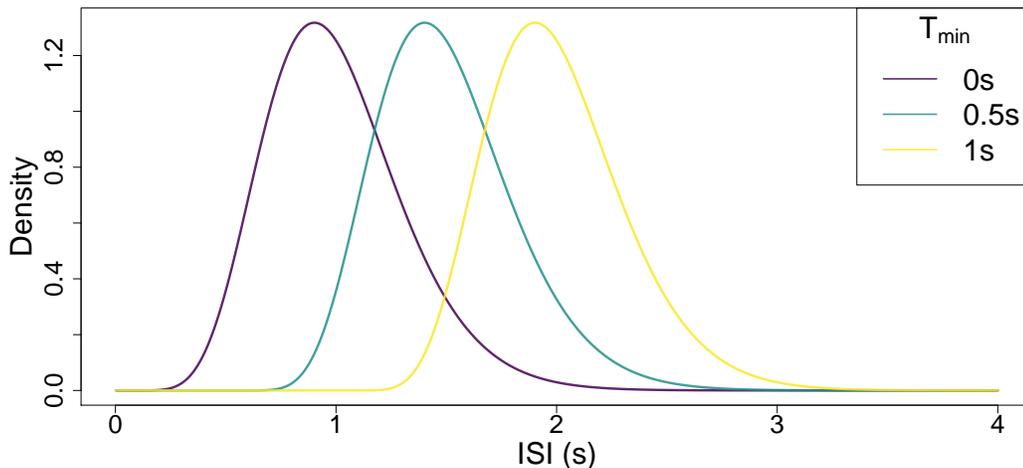


Figure 4.1: Illustration of the ISI distribution with varying values of T_{\min} . The purple, blue and yellow lines correspond to a Gamma ISI distribution with $x = 1$, $\gamma = 10$ where $T_{\min} = 0\text{s}$, 0.5s and 1s , respectively.

4.1.1 Affect on the intensity function

In Section 3.1 we explained how we parameterised the ISI distributions such that the intensity function aligns with the average Ca^{2+} spiking rate irrespective of the spiking history. However, with the addition of T_{\min} this property no longer holds. This can be seen in Figure 4.1 since increasing T_{\min} while keeping the intensity function the same increases the average ISI time. If we fix the intensity function $x(t) = a$ for some $a > 0$, without the refractory period the mean ISI is $1/a$. If we include the refractory period the mean ISI is shifted by the refractory period giving $1/a + T_{\min}$.

To visualise the impact of the refractory period on the mean spiking rate, we simulate 1000 spike sequences from two inverse Gaussian ISI distributions with the refractory period which only differ in T_{\min} , where for one $T_{\min} = 0\text{s}$ and the other $T_{\min} = 0.2\text{s}$. We then bin the spike times to create peristimulus-time histograms. This is shown in Figure 4.2. We see that the model without the refractory period has a larger mean spiking rate which coincides with the intensity function of the ISI

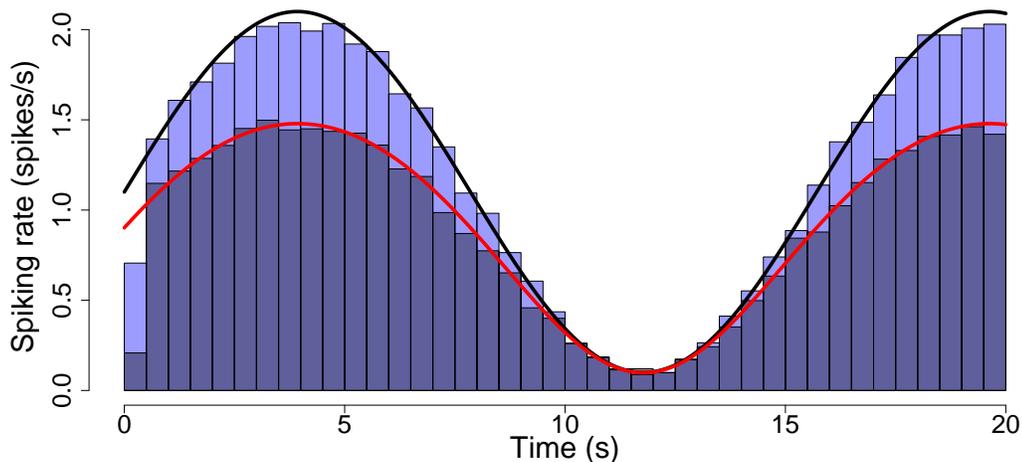


Figure 4.2: Illustration of the affect of including the refractory period as a model parameter on the intensity function. We simulated 1000 spike sequences from two ISI distributions. Namely, an inhomogeneous Inverse Gaussian ISI distribution with the refractory period where the ISI parameter $\lambda = 2$ and intensity function $x(t) = \sin(t/2.5) + 1.1$ — shown by the black line — and $T_{\min} = 0$ s or 0.2s. The lighter blue histogram is obtained by binning the spike sequences when $T_{\min} = 0$ and the darker blue when $T_{\min} = 0.2$. The red line corresponds to the intensity function sifted by T_{\min} as described in equation (4.4).

model — shown by the black line. When $T_{\min} > 0$ the mean spiking rate decreases. This means that the intensity function no longer describes the mean spiking rate when $T_{\min} > 0$. Notice that when the intensity function is larger, the refractory period causes a larger decrease in the mean spiking rate. This occurs because the larger the intensity function, the smaller the average ISI and therefore the larger the affect of the refractory period. For example, suppose we have the intensity function $x = 5$, then the mean spiking rate without a refractory period is $1/x = 0.2$ s. If we include a refractory period of $T_{\min} = 0.2$ s, then the mean spiking rate becomes $1/x + T_{\min} = 0.4$ s. Therefore the average ISI time has doubled which is equivalent to 50% of the spiking rate without the refractory period. On the other hand, if $x = 0.5$

this gives a mean spiking rate of $1/x = 2\text{s}$ and adding $T_{\min} = 0.2$ changes the mean spiking rate to $1/x + T_{\min} = 2.2\text{s}$ which is equivalent to a spiking rate of $5/11$ which is approximately 90.1% of the spiking rate without the refractory period.

We can calculate the resultant mean spiking rate by generalising the above calculations by

$$(\text{Mean spiking rate}) = (x + T_{\min})^{-1}. \quad (4.4)$$

This result is a quasi-stationary approximation that is only valid for slowly changing $x(t)$.

This is shown by the red line in Figure 4.2, which coincides with the mean spiking rate for the model with the refractory period.

Therefore, when considering models including the refractory period we cannot directly compare intensity functions. Instead, we need to compare the mean spiking rate which is calculated using the intensity function and refractory period, as shown in (4.4).

4.2 Estimating T_{\min}

In this section we explore how the addition of the refractory period as a model parameter affects the computation of the posterior distribution. Firstly, recall that we use a Bayesian framework where the posterior distribution is calculated using Bayes' theorem. More precisely, consider the inhomogeneous ISI model with the refractory period with model parameters: the intensity function x , the ISI parameter θ and the refractory period T_{\min} . Given a spike sequence $\mathbf{y} = \{y_i\}_{i=1}^N$ the posterior ISI distribution is given by

$$\pi(x, \theta, T_{\min} | \mathbf{y}) = \frac{\pi(\mathbf{y} | x, \theta, T_{\min}) \pi(x, \theta, T_{\min})}{\pi(\mathbf{y})}. \quad (4.5)$$

Therefore to calculate the posterior distribution we need: the likelihood $\pi(\mathbf{y} | x, \theta, T_{\min})$, the prior $\pi(x, \theta, T_{\min})$ and the normalising constant $\pi(\mathbf{y})$.

First we consider the likelihood. Similar to the ISI model without the refractory

period, the likelihood factorises into individual ISIs giving

$$\pi(\mathbf{y}|x, \theta, T_{\min}) = p_1(y_1|x, \theta, T_{\min})p_T(T, y_N|x, \theta, T_{\min}) \prod_{i=2}^N p_{\text{refract}}(y_{i-1}, y_i|x, \theta, T_{\min}). \quad (4.6)$$

The ISI distribution p_{refract} is defined in equation (4.1). However, we need to decide how T_{\min} affects p_1 , the time until the first Ca^{2+} spike, and p_T , the time after the last Ca^{2+} spike until the end of the experiment. 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 have already occurred. As in the model without the refractory period, we assume p_1 comes from an inhomogeneous Exponential distribution

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

The refractory period does affect p_T . This is because if the time difference between the last spike y_N and the end of the experiment T is less than T_{\min} , then the cell is in the refractory period and no Ca^{2+} spike can occur. Therefore $p_T = 1$ if $T - y_N < T_{\min}$. To be consistent with the model without the refractory period we choose p_T to represent the probability of no spike in $[y_N, T]$ from an inhomogeneous Exponential distribution with a refractory period

$$p_T(T, y_N|x, \theta, T_{\min}) = \begin{cases} 1 & \text{if } T - y_N < T_{\min}, \\ e^{-X(y_N + T_{\min}, T)} & \text{otherwise.} \end{cases} \quad (4.8)$$

Consider next the prior distribution. We choose to give each model parameter an independent prior. Therefore, the prior factorises giving $\pi(x, \theta, T_{\min}) = \pi(x)\pi(\theta)\pi(T_{\min})$. Details of the prior for the intensity function and ISI parameter can be found in Chapter 2. Therefore, to generalise the prior to include the refractory period we only need to add a prior for T_{\min} . As T_{\min} can only take values between zero and the minimum ISI time $T_{\text{ISI}} = \min_{i \in \{1, \dots, N-1\}} y_{i+1} - y_i$, we require a prior distribution on $[0, T_{\text{ISI}}]$. As a priori we have no knowledge of the refractory period, we use a non-informative prior, namely the uniform distribution

$$\pi(T_{\min}) = 1/T_{\text{ISI}}. \quad (4.9)$$

Similar to the model without the refractory period, the normalising constant $\pi(\mathbf{y})$ is computationally challenging. Therefore, to avoid calculating it we utilise MCMC algorithms to sample from the posterior distribution. In particular, in Chapter 2 we used a Gibbs sampler to update each model parameter individually, while the remaining parameters are fixed. With the addition of T_{\min} this creates an extra step where we update the refractory period. We choose to use a RW-Metropolis algorithm to update T_{\min} , which is described in further detail later in this section. Therefore, to sample from the posterior distribution, the MCMC algorithm begins by choosing initial values x_0 , θ_0 and $(T_{\min})_0$. Then we iterate the process

Step 1: Update $x(t)$ applying either: a RW-Metropolis algorithm for the constant prior (Section 2.4.1); a RJMCMC algorithm for a piecewise constant prior (Section 2.4.2) or an under-relaxed algorithm for a GP prior (Section 2.4.3).

Step 2: Update θ using a RW-Metropolis algorithm (Section 2.4.4).

Step 3: Update T_{\min} using a RW-Metropolis algorithm.

In Steps 1 and 2 the only difference from the model without the refractory period comes from the likelihood function, which is updated to include the refractory period as shown in equation (4.6).

To update the refractory period we use a RW-Metropolis algorithm, similar to the method used for the ISI parameter shown in Section 2.4.4. Recall that as we use the RW-metropolis algorithm within a Gibbs sampler when updating T_{\min} we treat the intensity function x and ISI parameter θ as fixed. Therefore we evaluate the conditional distribution $\pi(T_{\min}|\mathbf{y}, x, \theta)$ rather than the full posterior distribution. Note that any terms not involving T_{\min} will cancel in the acceptance probability. Therefore the conditional distribution is proportional to the likelihood function, as

shown below

$$\pi(T_{\min}|\mathbf{y}, x, \theta) \propto \pi(\mathbf{y}|x, \theta, T_{\min})\pi(x, \theta, T_{\min}), \quad (4.10)$$

$$= \pi(\mathbf{y}|x, \theta, T_{\min})\pi(x)\pi(\theta)\pi(T_{\min}), \quad (4.11)$$

$$\propto \pi(\mathbf{y}|x, \theta, T_{\min})\frac{1}{T_{\text{ISI}}}, \quad (4.12)$$

$$\propto \pi(\mathbf{y}|x, \theta, T_{\min}). \quad (4.13)$$

Since we are only updating T_{\min} we remove any terms that are purely in x or θ in the likelihood $\pi(\mathbf{y}|x, \theta, T_{\min})$. This is shown below for the inhomogeneous Gamma ISI distribution with the refractory period, in the case where each ISI and the time after the last spike are larger than T_{\min} .

$$\pi(T_{\min}|\mathbf{y}, x, \theta) \propto \pi(\mathbf{y}|x, \theta, T_{\min}) \quad (4.14)$$

$$= x(y_1)e^{-X(0,y_1)}e^{-X(y_N+T_{\min},T)} \quad (4.15)$$

$$\times \prod_{i=2}^N \frac{\gamma x(y_i)}{\Gamma(\gamma)} [\gamma X(y_{i-1} + T_{\min}, y_i)]^{\gamma-1} e^{\gamma X(y_{i-1}+T_{\min},y_i)}, \quad (4.16)$$

$$\propto e^{X(y_N+T_{\min},T)} \prod_{i=2}^N X(y_{i-1} + T_{\min}, y_i)^{\gamma-1} e^{\gamma X(y_{i-1}+T_{\min},y_i)}. \quad (4.17)$$

If the time after the last spike is smaller than T_{\min} then the term $e^{X(y_N+T_{\min},T)}$ disappears.

In order to sample the refractory period we first initialise T_{\min} to be half the minimum ISI time $(T_{\min})_0 = T_{\text{ISI}}/2$. Suppose the current refractory period is $(T_{\min})_{\text{cur}}$. To propose a candidate refractory period $(T_{\min})_{\text{can}}$ we draw from the normal distribution with mean $(T_{\min})_{\text{cur}}$ and variance $\sigma_{T_{\min}}^2$, $(T_{\min})_{\text{can}} \sim N((T_{\min})_{\text{cur}}, \sigma_{T_{\min}}^2)$. The sampling variance is often taken to be 1/20 of the minimum ISI time $\sigma_{T_{\min}}^2 = T_{\text{ISI}}/20$, to ensure proposed values often occur in the region $[0, T_{\text{ISI}}]$.

By using a RW-Metropolis algorithm the transition probabilities cancel and the probability we accept $(T_{\min})_{\text{can}}$ becomes

$$p_{\text{acc}} = \min \left\{ 1, \frac{\pi((T_{\min})_{\text{can}}|\mathbf{y}, x, \theta)}{\pi((T_{\min})_{\text{cur}}|\mathbf{y}, x, \theta)} \right\}. \quad (4.18)$$

Converting to the logarithmic scale we accept the candidate value if

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

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

4.3 Advantages and disadvantages of including T_{\min}

With the introduction of the refractory period, the model remains theoretically identifiable, where we can learn the true values of our model's underlying parameters given an infinite number of observations. However, with the addition of the refractory period we have another parameter to infer. With the limited information available in a single Ca^{2+} spike sequence we need to investigate whether the extended model leads to a more informative posterior ISI distribution compared to the model without the refractory period. We do this by considering two examples using simulated spike sequences to fit the inhomogeneous Exponential ISI distribution with and without the refractory period and the inhomogeneous inverse Gaussian ISI distribution with and without the refractory period.

We begin by simulating spike sequences from a known model, and fitting the ISI distribution with and without the refractory period. We generate 50 spike sequences in [0s, 20s] from an inhomogeneous inverse Gaussian ISI distribution with intensity function $x(t) = 2$, ISI parameter $\lambda = 4$ and $T_{\min} = 0.2$. This gives an average of 28.5 spikes per spike sequence. For both the simulated spike sequences and the models we infer we will assume the intensity function is constant. This allows comparison of the ISI distribution directly in addition to the model parameters. We use the MCMC algorithms to sample 100,000 iterations from each model. We remove the first 20,000 iterations as a burn-in period. The MCMC parameters were chosen to be $\sigma_x^2 = 0.25$ and $\sigma_h^2 = 0.25$. When we infer the refractory period we set $\sigma_{T_{\min}}^2 = T_{\text{ISI}}/20$, where T_{ISI} is the minimum ISI time. To visualise the posterior distribution we randomly choose 200 iterations. For each iteration we convert the model parameters to an ISI probability distribution and plot each ISI probability distribution on the same graph. Each line has a large transparency, therefore darker regions correspond to more likely values of the ISI distribution. We use this approach because it illustrates the uncertainty in the estimation of the ISI distribution and the refractory period.

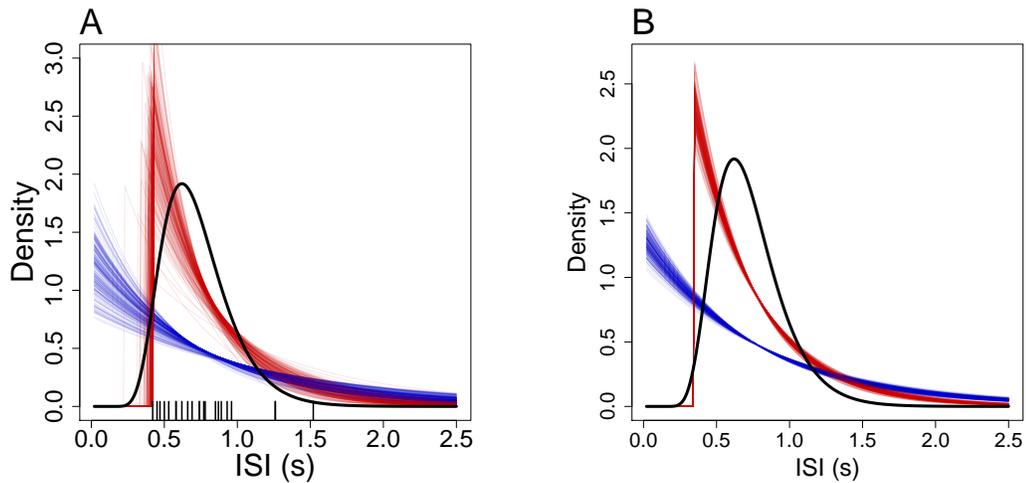


Figure 4.3: Illustration of the posterior inhomogeneous Exponential ISI distribution with and without the refractory period. (A,B) show samples of the posterior ISI distribution when fitted to one and ten simulated spike sequences, respectively. The black line represents the ISI distribution the simulated spike sequences were drawn from. Red and blue lines correspond to samples from the model with and without the refractory period, where the overlap is shown in purple. In (A) the ISIs used to fit the models are shown by black ticks on the x axis.

We first consider the inhomogeneous Exponential distribution with and without the refractory period. This is the only ISI distribution considered with no ISI parameter. Moreover, an Exponential distribution is decreasing over time therefore no parameterisation can capture a refractory period. Therefore, the inhomogeneous Exponential ISI distribution with the refractory period is unique from the distribution without it.

This is illustrated in Figure 4.3, where the models with and without the refractory period are shown in red and blue, respectively. We see that neither model can capture the underlying inverse Gaussian ISI distribution — shown by the black line — whether the models are fitted to one spike sequence (Figure 4.3(A)) or ten spike sequences (Figure 4.3(B)). However, the model with the refractory period improves upon the

model without it. In particular, with the refractory period the model captures that no ISIs are smaller than 0.4s. This example shows that the addition of the refractory period for the Exponential ISI distribution allows the model to capture a larger range of ISI distributions. Moreover, the posterior distribution of the Exponential ISI model with the refractory period is unique from the posterior distribution of the inhomogeneous Exponential without the refractory period.

All other ISI distributions — Gamma, inverse Gaussian, log-normal and Weibull — contain an ISI parameter, which allows for flexibility in the variance of the ISI distribution. In particular, for each the ISI parameter allows the density of the ISI distribution to remain close to zero when the ISI time is small. This can be seen for the Gamma ISI distribution in Figure 4.1, where the density is close to zero over the first 0.2s when $T_{\min} = 0$ s. Therefore, it is unclear whether the refractory period can be disentangled from the intensity function and ISI parameter.

To investigate, we repeat the process shown for the Exponential ISI distribution in Figure 4.3 for the inverse Gaussian ISI distribution. This is shown in Figure 4.4. We find that the ISI distribution with or without the refractory period is indistinguishable when the models are fit to either one or ten spike sequences shown in Figure 4.4(A-B), respectively. This is illustrated by samples of the posterior ISI distribution with and without the refractory period (red and blue lines) overlapping — creating the illusion of purple lines. This example illustrates the difficulty in separating the refractory period from the intensity function and ISI parameter. This is because the refractory period can be subsumed into a combination of the intensity function and ISI parameter. Although shown for the inverse Gaussian ISI distribution the same holds true for the Gamma, log-normal and Weibull ISI distributions. Therefore, the addition of the refractory period has not provided any extra information about the ISI distribution when the models are fitted to one or ten spike sequences.

Although the refractory cannot be accurately estimated as a model parameter we can approximate it by considering samples of the posterior distribution shown by the blue lines in Figure 4.4. We find that the samples of the posterior ISI distribution contain the refractory period since the probability density is close to zero in $[0\text{s}, 0.3\text{s}]$.

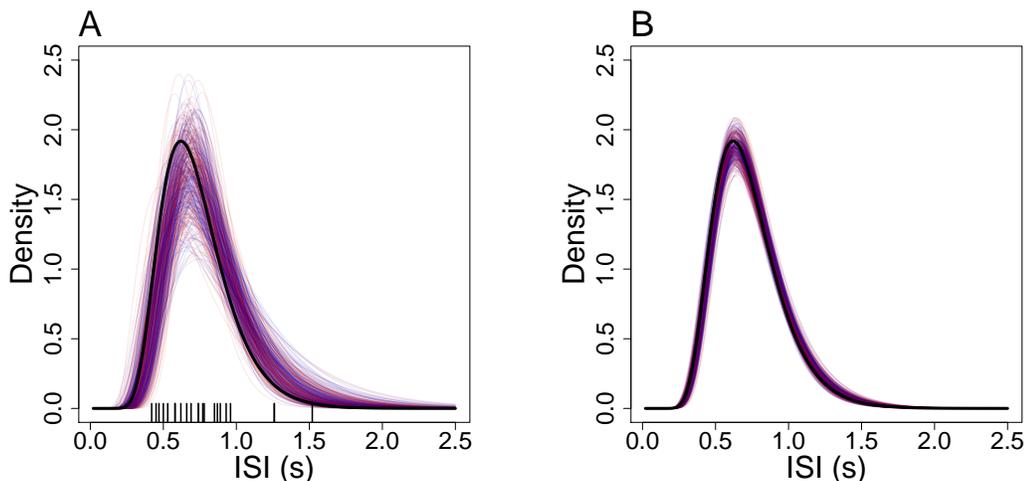


Figure 4.4: Illustration of the posterior inhomogeneous inverse Gaussian ISI distribution with and without the refractory period. (A,B) show samples of the posterior ISI distribution when fitted to one and ten simulated spike sequences, respectively. The black line represents the ISI distribution the simulated spike sequences were drawn from. Red and blue lines correspond to samples from the model with and without the refractory period, where the overlap is shown in purple. In (A) the ISIs used to fit the models are shown by black ticks on the x axis.

Therefore, for each sample we can find the first time that the probability density function reaches a threshold, such as 0.01. This will provide an upper bound for the refractory period in each sample. For the samples shown in Figure 4.4(A,B) we find that the upper bound for the refractory period — for threshold equal to 0.01 — is $[0.21\text{s}, 0.30\text{s}]$ and $[0.23\text{s}, 0.27\text{s}]$, respectively. Therefore, from the ISI distribution plots we have estimated an upper bound for the refractory period.

We now investigate the effect of the refractory period on the posterior distribution of the intensity function and ISI parameter. This is illustrated in Figure 4.5, where we plot the value of the constant intensity function against the ISI parameter corresponding the models shown in Figure 4.4(A). We see that adding the refractory period increases the range of values that the intensity function takes and shifts the values of

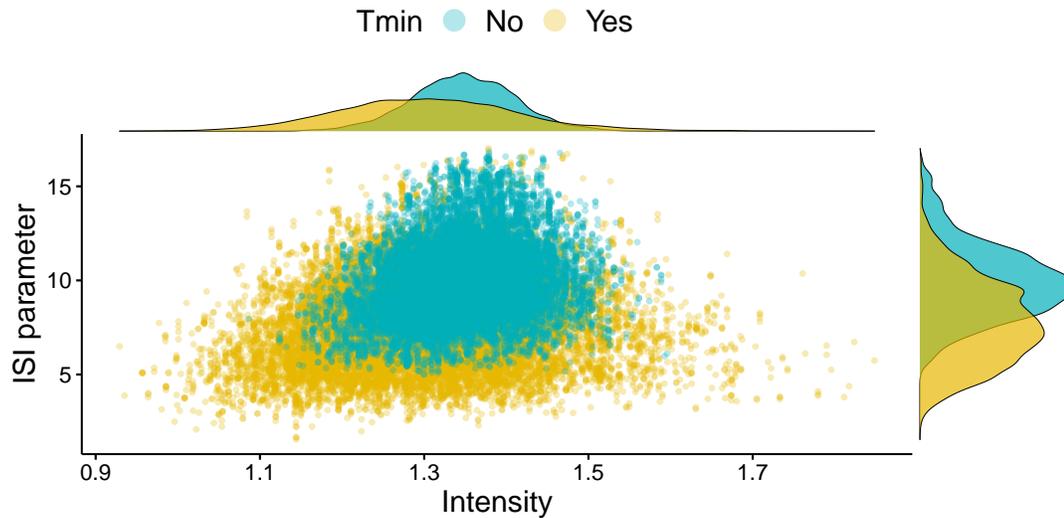


Figure 4.5: Samples of the posterior intensity function and ISI parameter from an inhomogeneous inverse Gaussian ISI distribution with and without the refractory period, shown in blue and yellow, respectively. The posterior distribution coincides with the ISI distribution shown in Figure 4.4(A), where the intensity function is constant. Above and to the right of the plot we provide probability density functions of the intensity function and ISI parameter, respectively.

the ISI parameter. In particular, the 95% credible region of the intensity function went from $[1.22, 1.47]$ to $[1.09, 1.53]$ with the inclusion of the refractory period. This corresponds to a 73% increase in the range of the 95% credible region, whereas the ISI parameter changed from $[6.53, 14.52]$ to $[3.90, 12.32]$ with the inclusion of the refractory period. This demonstrates that the ISI parameter of the inverse Gaussian ISI distribution becomes smaller with the inclusion of the refractory period and this also corresponds to a 5% increase in the range of the 95% credible region.

In the above example we used the inverse Gaussian ISI distribution, however the same behaviour occurs for the Gamma, log-normal and Weibull distributions. This is because in each model the intensity function and ISI parameter can be used in combination to create near zero probability density for small ISI times, mirroring the refractory period. In the example provided we concentrate on a constant intensity

function to create clear visualisations of the issue. The constant intensity function can be viewed as a special case of the inhomogeneous ISI distributions and the issue remains when allowing for time-varying intensity functions.

In summary, we have shown that the addition of the refractory period to the inhomogeneous Exponential ISI distribution allows for a wider variety of Ca^{2+} ISI dynamics to be captured. However, for all other considered ISI distributions — Gamma, inverse Gaussian, log-normal and Weibull — it is difficult to untangle the refractory period from the intensity function and ISI parameter. We find that the models without the refractory period capture a near identical posterior ISI distribution to the model with the refractory period. Moreover, inclusion of the refractory period leads to more uncertainty in the parameter estimates. Therefore, we only include the inhomogeneous Exponential ISI distribution with refractory period when analysing Ca^{2+} spikes from HEK293 cells and astrocytes in Chapter 5.

CHAPTER 5

Application to Real Data

In this chapter, we use the model and methods developed in Chapters 2-4 to fit the previously discussed inhomogeneous ISI distributions to Ca^{2+} spike sequences from astrocytes and HEK293 cells. We use these models to gain insight into the Ca^{2+} response of the two cell types, which is driven by changes in the stimulus and stimulus strength. In particular, since our models are inhomogeneous we examine how the Ca^{2+} response varies over time. This is crucial in understanding Ca^{2+} dynamics as many pathways occur on different time scales.

We also identify parameter values for our ISI models that can be used to easily create surrogate Ca^{2+} spike sequences with similar properties to the real data. This is beneficial because if we have a signalling cascade that is dependent on the timing of Ca^{2+} oscillations then the surrogate spike sequences can be used as a realistic input. This provides a cheaper alternative to simulating the Ca^{2+} concentration from a detailed mechanistic model, which is computationally more expensive because it generates additional characteristics such as the amplitude or width of the Ca^{2+} oscillations.

We first describe the methods to assess and compare the inhomogeneous ISI

distributions. We then use these methods to analyse Ca^{2+} spike sequences obtained from astrocytes and HEK293 cells under constant stimulus. Finally, we apply our methods to Ca^{2+} spike sequences obtained from HEK293 cells challenged with a time dependent stimulus. Specifically, the cells are challenged with a step-change, waves and pulses of carbachol.

5.1 Model Assessment

In this section, we explore methods to assess the quality of our ISI models. In particular, we require methods to distinguish the most appropriate ISI distribution and prior for the intensity function.

We follow and build upon the approach used by Tilunaite et al., [16] which utilises the time-rescaling theorem to rescale the ISI times, such that the rescaled ISI times are independent and identically distributed (i.i.d) exponential random variables with unit rate. To assess the goodness of fit they employ Quantile-Quantile (Q-Q) plots and Kolmogorov-Smirnov (K-S) plots.

We begin by first illustrating how to create Q-Q and K-S plots and use them for model assessment. We then improve on the methods developed by Tilunaite et al., namely by using radians for the slope of the Q-Q and K-S plots and employing violin plots instead of box plots. We also create other tools to aid model assessment. Specifically, simulating spike sequences and plots of the rescaled ISIs in the order that the ISIs occur. These two methods can detect whether the model can generate spike sequences similar to those used to fit the model and dependence in the rescaled ISIs.

5.1.1 Kolmogorov-Smirnov and Quantile-Quantile plots

Recall that we define our models in terms of an intensity function $x(t)$ and ISI parameter θ , where the ISI probability density function of spikes occurring at times t and s with no spikes between is denoted by $p(t, s|x(t), \theta)$.

The aim of this section is to assess how well our model — defined by $x(t)$ and θ and ISI distribution — describes the Ca^{2+} spike sequence with N spikes $\{y_i\}_{i=1}^N$ used

to fit the model. To assess the quality of our ISI models we utilise the time rescaling transformation. It states that any point process with an integrable conditional intensity function can be transformed into a homogeneous Poisson process with unit rate [97, 107, 131]. The conditional intensity function is defined as

$$q(t) = \lim_{\Delta t \rightarrow 0} \frac{\Pr(N(t + \Delta t) - N(t) = 1 | \mathcal{H}_t)}{\Delta t}, \quad (5.1)$$

where $N(t)$ is the number of spikes in the interval $(0, t)$ and \mathcal{H}_t is the spiking history up to time t — but not including time t . The conditional intensity function $q(t)$ defines the instantaneous spike rate at time t given the history of spikes up to time t . To test our models it is advantageous to define the conditional intensity function in terms of the ISI probability density function by

$$q(t) = \frac{p(t, y_k | x(t), \theta)}{1 - \int_{y_k}^t p(u, y_k | x(t), \theta) du} \quad (5.2)$$

for $t > y_k$, where y_k is the last spike before time t [105].

Let $R(t) = \int_0^t q(u) du$ denote the integrated conditional intensity function. The time rescaling theorem [107] states that rescaled spike times $R(y_i)$, for $i = 1, \dots, N$ are from a Poisson process with unit rate. Therefore, the rescaled ISI times τ_k

$$\tau_k = R(y_k) - R(y_{k-1}) = \int_{y_{k-1}}^{y_k} q(u) du \quad (5.3)$$

are i.i.d exponential random variables with unit rate.

A Q-Q plot is a graphical method to compare two probability methods by plotting their quantiles against each other. In our case, we compare the quantiles of the exponential distribution with rate one and our empirical quantiles — the rescaled ISI times. If our model is correct the rescaled ISIs should be drawn from an exponential distribution with rate one and therefore the points of the Q-Q plot should lie on the line $y = x$. To construct a Q-Q plot, we first generate the empirical quantiles $\hat{\tau}_k$ as the set of ordered τ_k from smallest to largest. The quantiles $\bar{\tau}_k$ for an exponential distribution with rate one are given by $\bar{\tau}_k = -\log\left(1 - \frac{k-0.5}{N}\right)$ for $k = 1, \dots, N$ where N is the number of Ca^{2+} spikes. If the model is correct then the Q-Q plot should form a straight line through the origin with slope one [132, 133].

The K-S plot is another goodness of fit graphical test[134]. It claims the model is correct if $u_k = 1 - e^{-\tau_k}$ are i.i.d uniform random variables on $[0, 1]$. Therefore, plotting the ordered u_k against $s_k = \frac{k-0.5}{N}$ for $k = 1, \dots, N$ forms a straight line through the origin with slope one. The K-S plot is similar to the Q-Q plot except the quantiles are transformed to the uniform distribution on $[0, 1]$. The K-S plot is more sensitive at the center of the distribution, whereas Q-Q plots are more sensitive at the tails.

To ascertain which ISI model best describes a dataset of Ca^{2+} spike sequences we calculate the slope of the Q-Q and K-S plots for each individual spike sequence in the dataset. The slope is calculated using linear regression using least squares estimation. An ISI model performs well if the slopes obtained from individual spike sequences are close to one. To visualise the set of slopes obtained from the Q-Q and K-S plots Tilunaite et al. [16] used box and whisker plots. This is a good approach because it indicates the spread of the slopes as well as any skewness. However, we choose to use violin plots to visualise the slopes. A violin plot, is similar to a box and whisker plot, except in addition we have a kernel density plot which gives a visual representation of how the slopes are distributed. This is beneficial since it informs on how many of the slopes are close to one. Examples of violin plots compared to box and whisker plots are shown in Figure 5.1, for three sets of slopes E , F and G . E contains 200 draws from a uniform distribution on $[0.3, 0.99]$, $F = 1/E$ and G contains the smallest 100 values from E and the largest 100 from F . We see that the white point and black line in the violin plot agrees with the box of the box and whisker plot — consisting of the 25%, 50% and 75% quantiles. Furthermore, the range of the box and whisker plot and violin plots are the same. However, the violin plot's shape informs on the distribution of the slopes, unlike the box and whisker plot. This can be seen for the set G in Figure 5.1(A), where the violin plot is narrower in $[0.8, 1.2]$. This indicates that few of the slopes are in this region. No such information is provided in the box and whisker plot.

Note that the slope obtained by least square linear regression is in the region $[0, \infty]$. We cannot have negative slopes since the experimental quantiles are ordered

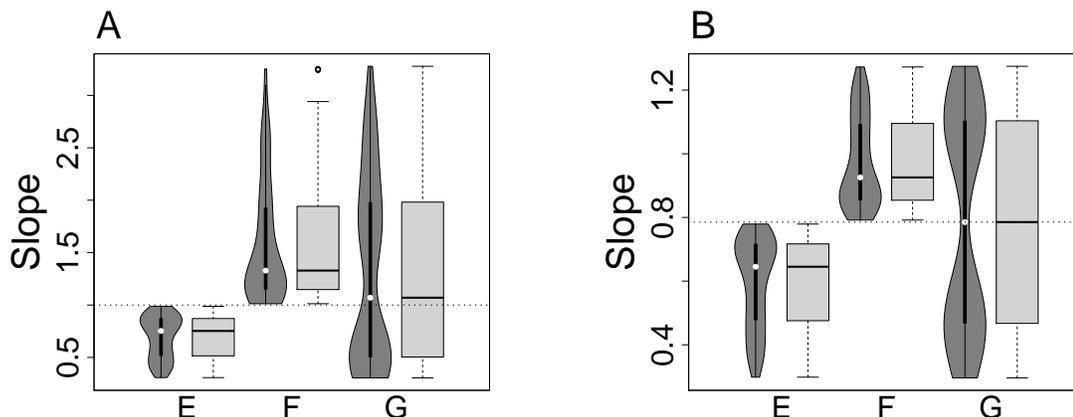


Figure 5.1: Illustration of violin plots compared to box and whisker plots, for 3 datasets E , F and G . E consists of 200 draws from a uniform distribution on $[0.3, 0.99]$, $F = 1/E$ and G consists of the smallest 100 draws from E and the largest 100 from F . (A,B) show the difference from using the slope compared to the angle of the slope in radians.

from smallest to largest. The ideal slope equals one whereas slopes in $[0, 1)$ and $(1, \infty)$ are too flat and steep, respectively. We have no preference whether the slope is flatter or steeper than the ideal slope. Therefore, a slope of say 1.5 is as far from the ideal slope as the slope of $1/1.5 = 0.67$ as they are the mirror of each other through the line with slope 1. However, since 0.67 is closer to one than 1.5, we prefer the shallower slope over the steeper one. This can be seen in Figure 5.1(A) where the slopes in datasets E and F are equally far from the ideal slope, i.e $F = 1/E$. However, for both the violin plots and box and whisker plots we find that F is stretched over a larger range than E and further from ideal slope line — shown by the dashed line. To mitigate this issue we transform the slope s_{slope} to the angle of the slope in radians s_{angle} by $s_{\text{angle}} = \tan^{-1}(s_{\text{slope}})$, mapping $[0, \infty]$ into $[0, \pi/2]$. The ideal slope of 1 is transformed to the angle $\pi/4$. Using this transformation mirrored slopes are now equidistant from $\pi/4$. For example slopes 1.5 and 0.67 are transformed to angles

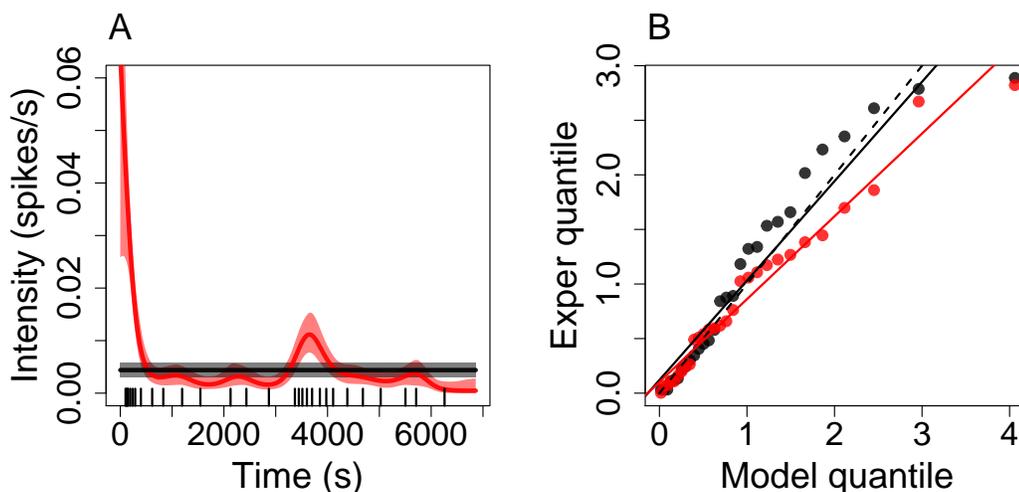


Figure 5.2: Posterior intensity distribution and Q-Q plot using an inhomogeneous Gamma ISI distribution with a constant and GP prior — shown in black and red, respectively. (A) The 95% confidence interval and posterior mean of the intensity function fitted to a spike sequence shown by the black ticks. (B) Q-Q plot for the models shown in (A), with the slope obtained by least square linear regression. The dashed line represents the line with slope one.

0.983 and 0.588, respectively, and both are 0.197 away from $\pi/4$. This can be seen in Figure 5.1(B) where the violin plots of E and F are now mirror images through the line $\pi/4$ and G is symmetric. From now on we use the angle of the slope rather than the slope directly.

5.1.2 Issue with Q-Q and K-S plots

Note that for both the Q-Q and K-S plot we order the experimental quantiles from smallest to largest. Therefore, neither approach accounts for the dependence between the τ_k . This can cause an issue where the Q-Q or K-S plots show that a model fits the data well, even if the τ_k are dependent. To visualise this, we fit an inhomogeneous Gamma ISI distribution to a Ca^{2+} spike sequence using a constant and GP prior for

the intensity function. The spike sequence comes from a step-change experiment, where a HEK293 cell was initially stimulated with $10\mu\text{M}$ carbachol. The stimulus was exchanged at 3430s for $50\mu\text{M}$ carbachol. The posterior mean of the intensity function with 95% credible interval is shown in Figure 5.2(A), where the spike times are given by the black ticks. From this figure it appears that the model fitted with the GP prior performs better than the model with the constant prior because the intensity is larger in the regions $[0\text{s}, 1000\text{s}]$ and $[3300\text{s}, 4200\text{s}]$ where there are more spikes. In Figure 5.2(B) we calculate the Q-Q plot for the models using a constant and GP prior shown by the black and red points, respectively. The Q-Q plot suggests that the constant prior leads to a better model than the GP prior. This is because the points are closer to the line through the origin with slope one — shown by the dashed line. Furthermore, by fitting the best line through the Q-Q points we find slopes of 0.91 and 0.76 for the constant and GP priors, respectively. Therefore, it appears that using a constant prior leads to the model which better describes the Ca^{2+} spike sequence.

To validate the findings of the Q-Q plot we plot the rescaled ISIs in the order that the ISIs occur. Namely, $\tau_1, \tau_2, \dots, \tau_N$. We denote this as the rescaled ISI occurrence plot. If the model is correct each τ_k should be exponentially distributed with rate 1 and independent of other rescaled ISIs. Thus, in this plot we look to identify any dependence between the rescaled ISIs. Moreover, we simulate spike sequences from the posterior distribution. If the model is suitable the simulated spikes should mirror those used to fit the model. This is shown in Figure 5.3.

First consider the rescaled ISI occurrence plot in Figure 5.3(A,B) for the constant and GP prior, respectively. For the constant prior we find the τ_k appear to depend on the order the ISIs occur. In particular, the smaller τ_k are often grouped together — i.e quantile numbers 1-6 and 16-22 in (A). Whereas, if we use a GP prior there is no evidence from the rescaled ISI occurrence plot that the τ_k are dependent. This provides evidence against using the constant prior compared to the GP prior. Moreover, we simulate 10 spike sequences from the posterior ISI distribution for both the constant and GP prior — shown in Figure 5.3(C,D). We find that the simulated

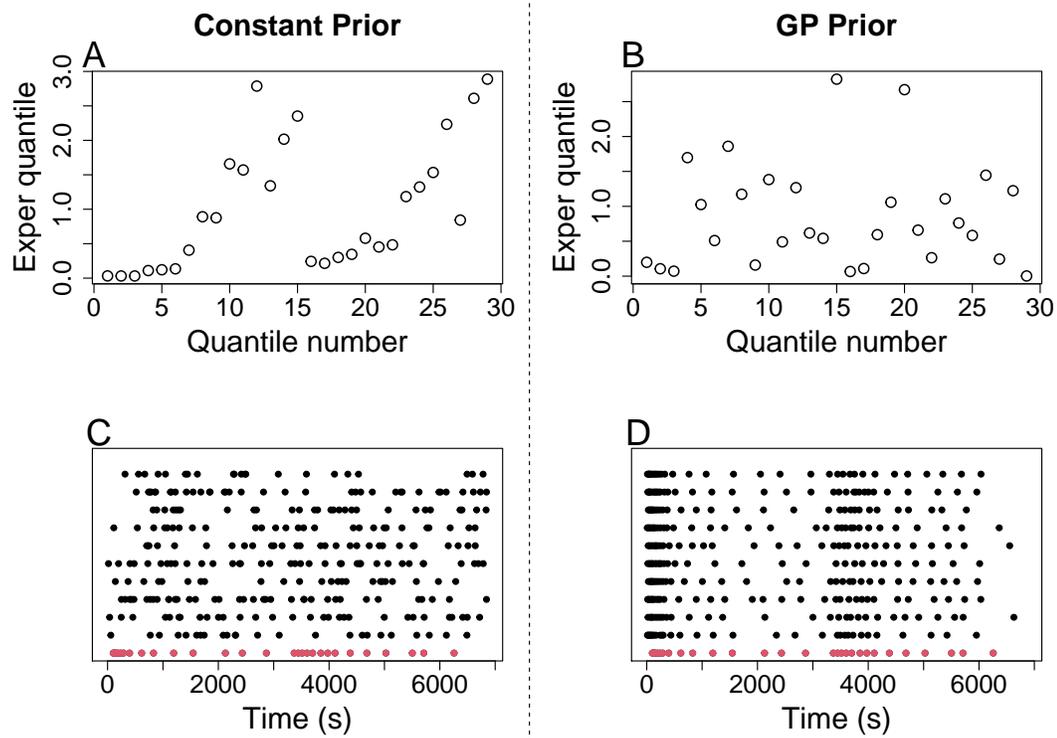


Figure 5.3: Comparison of rescaled ISI in the order they occurred (A,B) and ten simulated spike sequences (C,D) for the ISI distributions with a constant (A,C) and GP prior (B,D). In (A,B) the rescaled ISIs are obtained by rescaling the ISIs of the spike sequence shown in Figure 5.2(A) using the posterior Gamma ISI distribution fitted to the same spike sequence with a constant and GP prior, respectively. In (C,D) each black row corresponds to a simulated spike sequence and the red row corresponds to the spike sequence used to fit the models.

spike sequences generated from the model assuming a constant prior are not similar to the spike sequence used to infer the model. This is because the spike times occur randomly over the entire interval. When we used a GP prior the simulated spike sequences have a similar behaviour to the spikes used to fit the model, where we get more spikes in $[0s, 1000s]$ and $[3300s, 4200s]$ and less spikes elsewhere.

Therefore, using the rescaled ISI occurrence plot and simulated spike sequences shows that, for this example, the model using a GP prior is preferable to the model

with a constant prior. This, confirms that the Q-Q and K-S plots alone are not enough to assess the models. To decide which model best describes the Ca^{2+} spike sequences we use a combination of rescaled ISI occurrence plots, simulated spike sequences, Q-Q plots and K-S plots.

5.2 Constant Stimulus

In this section we apply our methods to HEK293 cells and astrocytes that have been challenged with a constant stimulus.

One could argue that the ISIs would be independent of time due to the stimulus not depending on time. Indeed, previous investigates have made this assumption, such as [6, 61]. However, such studies often require cleaning of the data. This is because once the stimulus is applied there is often a region of transients and potentially a linear trend in the ISIs, see for example [6]. However, since our models do not require stationary ISIs we can use the entire Ca^{2+} spike sequences without the removal of transients.

Therefore, by fitting our models we look to gain insight into the mechanics that drive the Ca^{2+} oscillations, and search for any time-dependent properties of the Ca^{2+} oscillations in cells challenged with a constant stimulus. Moreover, we investigate the difference in response between HEK293 cells and astrocytes, together with the difference caused by varying the stimulus strength.

5.2.1 Astrocytes

We first consider astrocytes that are stimulated with glutamate. We have two datasets containing 56 and 73 Ca^{2+} spike sequences where astrocytes were challenged with $10\mu\text{M}$ and $100\mu\text{M}$ glutamate, respectively. Each Ca^{2+} spike sequence is obtained from a unique cell. In both experiments the Ca^{2+} concentration was recorded every 0.5s and the stimulus was first applied at 120s. The experiments lasted 1192.5s and 1200s when the stimulus strength was $10\mu\text{M}$ and $100\mu\text{M}$, respectively. The experimental data was kindly provided by Dr. Alexander Skupin and for the detailed experimental

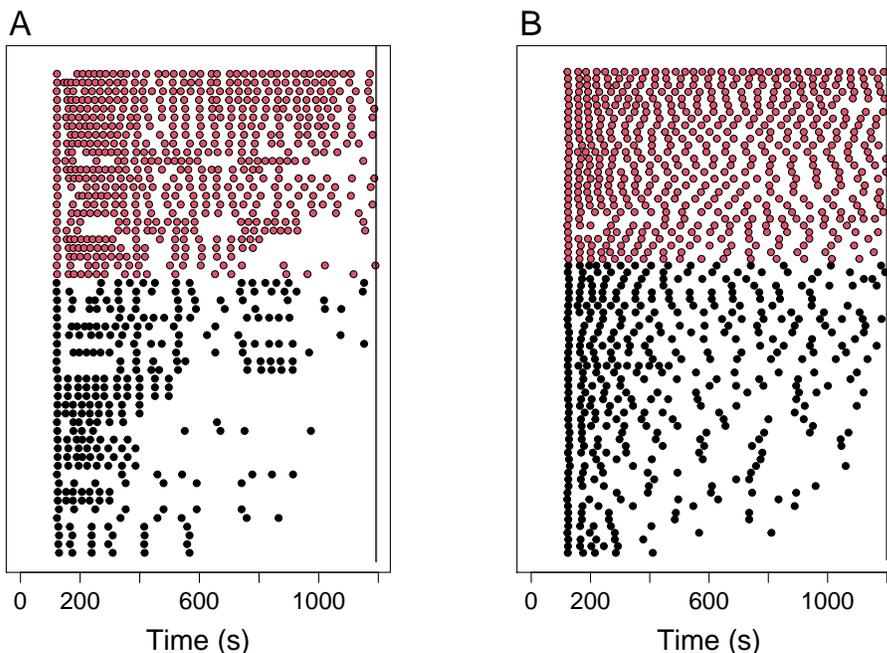


Figure 5.4: Raster of Ca^{2+} spike sequences obtained by stimulating astrocytes with $10\mu\text{M}$ and $100\mu\text{M}$ glutamate in (A,B), respectively. Each Ca^{2+} spike sequence corresponds to the response from a unique astrocyte. We split the datasets into the Ca^{2+} spike sequences with at least 15 spikes and those with fewer spikes, shown by the red and black points, respectively. The solid black line represents the length of the experiments.

protocol we refer the reader to [61].

We illustrate the Ca^{2+} spike sequences in Figure 5.11, where each row corresponds to the response of an individual cell, and each dot in a row corresponds to a spike time. In both datasets there are no Ca^{2+} spikes until the stimulus is applied. Once the stimulus is applied a spike is recorded in all cells within the next 7s. In general, we find there are more spikes just after the stimulus is applied than later in the experiment and some cells cease spiking before the end of the experiment. Comparing Figure 5.4(A) and Figure 5.4(B) we find it is more common for cells to stop spiking

before the end of the experiment if challenged with the smaller stimulus. Moreover, the ISIs appears more regular when cells are challenged with the larger stimulus. This is especially true for cells that have more Ca^{2+} spikes. Surprisingly, the cell with the most Ca^{2+} spikes is challenged with the smaller stimulus.

In Figure 5.4 we observe that all cells exhibit a Ca^{2+} spike at the onset of stimulus. Moreover, Ca^{2+} spikes appear to occur more frequently in the first half of the experiment. Therefore, we first fit the datasets as a whole to a single ISI model. This approach aims to learn whether a single ISI model can capture the behaviour of all responses to the same stimulus. In this approach each spike sequence is a realisation from the same ISI distribution. Another way to view this approach, is that we look at the mean response of astrocytes to the same stimulus.

We fit the datasets using an inhomogeneous Exponential, Gamma, inverse Gaussian, log-normal and Weibull ISI distributions. For each ISI distribution we use a constant, PWC and GP prior for the intensity function. We sample from the posterior with 100,000 iterations after an initial burn in of 100,000. The parameters for the constant prior were $\sigma_x^2 = 0.5$ and $\sigma_h^2 = 1$. The parameters for the PWC prior were $k_{\max} = 25$, $\lambda = 10$, $\kappa = \kappa_0 = 1$, $\mu = 0.5$ and the PWC heights had a martingale structure. Finally, the parameters for the GP prior were $w = 0.001$, $N = 2400$, $\sigma_v^2 = 1000$ and $\sigma_n^2 = 10^{-6}$. Since each dataset contains a large quantity of spike sequences we fit the length scale of the GP where the prior of l is an Exponential distribution with rate 0.01 and $\sigma_l^2 = 0.5$.

To decide which models perform best we consider simulated spikes sequences and consider the likelihood of the model. Of the three prior types we find that the PWC performs the best. This is because the PWC framework has the largest flexibility due to the intensity function being discontinuous, which allows the intensity function to abruptly vary over time. In this case, with a PWC prior we are able to capture the sharp increase in spiking rate at the onset of the stimulus followed by a brief plateau at a lower intensity — see Figure 5.5. Neither the GP prior nor constant prior are able to capture this feature.

Due to using a large number of spike sequences the posterior intensity function

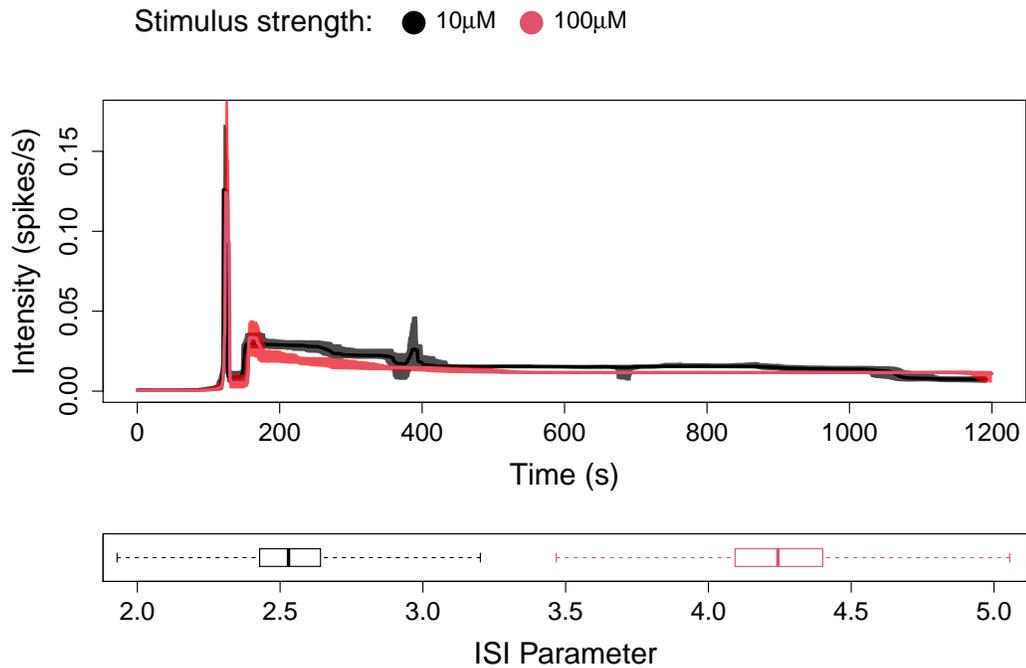


Figure 5.5: Comparison of the posterior of the inverse Gaussian ISI distribution with a PWC prior fitted to datasets of Ca^{2+} spike sequences when the applied stimulus was 10 μM (black) and 100 μM (red) glutamate. For each we plot the mean and 95% credible region of the posterior intensity function distribution and a box plot summary of the posterior ISI parameter distribution.

is similar across all choices of ISI distribution for the PWC prior. This is expected because the intensity function describes the mean spiking rate. We find that the inhomogeneous inverse Gaussian ISI distribution is the most suitable model for both the Ca^{2+} spike datasets.

To visualise the posterior distribution we show the mean and 95% credible region of the posterior intensity function distribution and a box plot summary of the posterior ISI parameter distribution. This is shown in Figure 5.5 when the applied stimulus is 10 μM and 100 μM glutamate, respectively. We find that the posterior intensity function has a similar shape irrespective of the strength of the stimulus. The posterior

intensity function remains close to zero until the onset of the stimulus, at which point a spike in the intensity function occurs. This shows that a Ca^{2+} spike occurs at the onset of stimulus. After this spike the intensity function immediately decreases from 0.126 to 0.0087 for approximately 20s. This shows that after the first Ca^{2+} spike there is often a recovery time before the next spike can occur. After 20s the intensity increases 4-fold indicating that the astrocytes begin to spike. We see that the intensity function gradually decreases until approximately 400s. Afterwards the intensity functions are approximately constant for the remainder of the experiment. However, we notice that the posterior intensity function is larger when the stimulus is $10\mu\text{M}$ in [140s, 1000s], indicating that cells present more spikes when applied with a smaller concentration of glutamate. Furthermore, the posterior intensity function decreases after 1000s when the stimulus strength is $10\mu\text{M}$. This shows that some cells do not spike until the end of the experiment, unlike when the stimulus strength is $100\mu\text{M}$. Finally, there is a small peak in the posterior intensity function fitted from cells challenged $10\mu\text{M}$ glutamate around 400s. Returning to the raster plot in Figure 5.4(A) we see that a subset of cells spike in this region causing the peak in the intensity function. Since this is not common to all cells, it is most likely that the underlying spiking rate would continue to decrease in [350s, 400s] and this is not a feature of the smaller stimulus. We find that the ISI parameter is larger when a larger stimulus is used. Recall that a larger ISI parameters gives rise to a larger variance in the ISI distribution — see Section 3.1. This suggests that there is less variability in the ISIs when the stimulus applied is larger.

To ascertain if the inverse Gaussian ISI model consistently describes all Ca^{2+} spike sequences in the dataset we simulate spike sequences from the posterior distribution. Each simulated spike sequence is created by first drawing a random iteration from the samples obtained using the MCMC, consisting of an intensity function and ISI parameter. We then follow the approach in Chapter 2 Section 2 to simulate spike sequences. 100 samples from the posterior ISI distribution fitted from astrocytes challenged with $10\mu\text{M}$ glutamate are shown in Figure 5.6. The simulated sequences are ordered by their total number of spikes. Comparing the simulated spike sequences to

Ca^{2+} spike sequences — see Figure 5.4(A) — we find less variability in the simulated spike sequences. In particular, simulated spike sequences do not capture that some Ca^{2+} spike sequences only have spikes in the first half of the experiment. This provides evidence that the mean response of astrocytes does not accurately describe the Ca^{2+} dynamics of individual cells. Therefore, we need to fit our ISI models to individual Ca^{2+} spike sequences to account for the cell-to-cell variability of Ca^{2+} oscillations.

Note that an individual Ca^{2+} spike sequence contains far less information than an ensemble of spike sequences. Therefore, when fitting ISI distributions to individual spike sequences we require the spike sequence to contain at least 15 spikes, such that there is enough information to infer the ISI distribution. These Ca^{2+} spike sequences are shown in red in Figure 5.4.

For each individual spike sequence we fit an inhomogeneous Exponential, Exponential with refractory period, Gamma, inverse Gaussian, log-normal and Weibull ISI distributions, where the intensity function has a constant, PWC or GP prior. We sample from the posterior distribution for 200,000 iterations after an initial burn in of 200,000. The parameters for the constant prior were $\sigma_x^2 = 0.5$ and $\sigma_h^2 = 1$. The parameters for the PWC prior were $k_{\max} = 25$, $\lambda = 10$, $\kappa = \kappa_0 = 1$ and $\mu = 0.5$ where the PWC heights have a martingale structure. Finally, the parameters for the GP prior were $w = 0.001$, $N = 2400$, $\sigma_v^2 = 1000$ and $\sigma_n^2 = 0$. The length scale l of the GP prior has a prior distribution utilising the PWC approximation, l_{pwc} by $l \sim \text{Uniform}(l_{\text{pwc}}, l_{\text{pwc}} + 5)$ — see Section 3.3 for more detail.

To visualise the difference between fitting ISI distributions to individual spike sequences and a dataset of spike sequences we compare the posterior intensity function distributions of both, where we use an inhomogeneous inverse Gaussian ISI distribution with a PWC prior. This is shown in Figure 5.6(B) when the Ca^{2+} spike sequences were obtained from astrocytes challenged with $10\mu\text{M}$ glutamate. The black lines correspond the mean of the posterior intensity function distribution fitted using individual Ca^{2+} spike sequences. The mean of these intensity function is shown by the yellow line. Furthermore, the mean and 95% credible region of the posterior

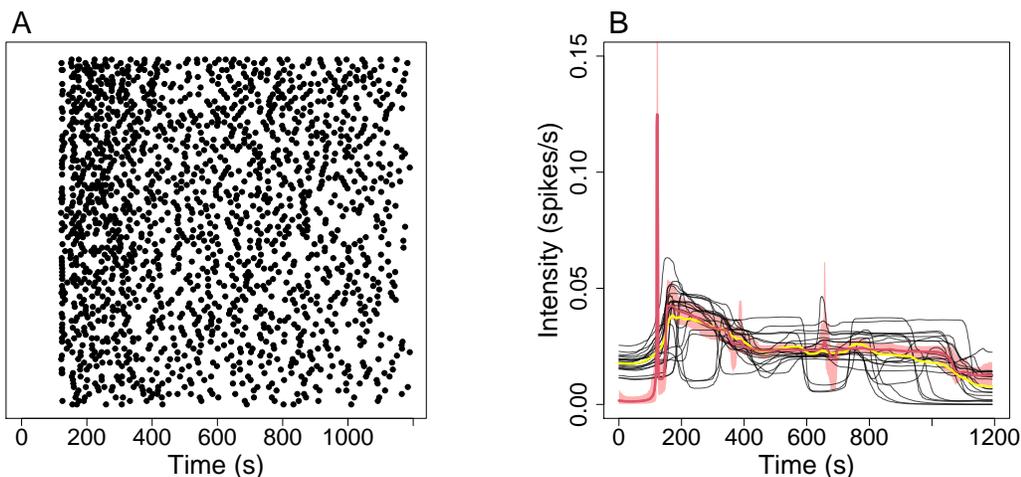


Figure 5.6: (A) 100 Samples taken from the posterior inverse Gaussian ISI distribution with a PWC prior fitted from the dataset of astrocytes challenged with $10\mu\text{M}$ glutamate — shown in Figure 5.4(A). (B) Comparison of posterior intensity functions fitted either from individual Ca^{2+} spike sequences (black) or the dataset of all Ca^{2+} spike sequences (red). For each ISI distribution fit from an individual Ca^{2+} spike sequence we show the mean of the posterior intensity function distribution and the average over these intensity functions is shown by the yellow line. For the ISI distribution fit from the dataset of all Ca^{2+} spike sequences we present the mean and 95% credible region of the posterior intensity function distribution.

intensity function distribution fitted using the entire dataset is shown by the red line and region — we denote this the collective fit. The mean of the individual intensity functions is close to the collective intensity function in $[180\text{s}, 1200\text{s}]$. However, there is a large difference in $[0\text{s}, 180\text{s}]$. Firstly, we find that the individual intensity functions do not converge to zero in $[0\text{s}, 120\text{s}]$, even though most spike sequences contain zero spikes in this region. This is due to the martingale prior for the heights of the PWC. In particular, note that the second step of the PWC intensity function often occurs at the peak in intensity at 120s. The prior for this height has a mean of the previous height — e.g. the intensity in $[0\text{s}, 120\text{s}]$. Thus, the prior distribution penalises large

changes in intensity, so that we have an artificially large intensity in $[0s, 120s]$. This can be countered in two manners: add more spike sequences or change the prior for the heights. As shown when considering the entire dataset the intensity converges to zero in $[0s, 120s]$ with adequate data. Changing to the independent prior for the heights would also alleviate this issue. However, recall that we chose the martingale prior over the independent to give structure to the intensity functions. This is because heuristically we want our intensity function to mirror the continuous nature of the arrival of stimuli to the cell. Notice that the mean of the individual intensity functions does not capture the initial spike after the onset of stimulus followed by the brief period of low intensity. Thus, when fitting our ISI model the additional information of multiple Ca^{2+} spike sequences captures this feature, whereas the individual Ca^{2+} spike sequences do not.

Comparing the posterior mean intensity functions inferred from individual spike sequences to the collective fit we see large differences in the intensity outside the 95% credible interval. In particular, some intensity functions have a dip between $[600s, 800s]$ and some converges to zero before the end of the experiment. This shows that the collective fit does not capture the spiking rate of individual astrocytes. In summary, we have shown quantitatively that astrocytes respond heterogeneously when applied with the same stimulus and that the intensity function inferred from the entire dataset gives a poor representation of the spiking behaviour. This also holds true for astrocytes challenged with $100\mu M$ glutamate.

To assess each choice of ISI distribution and prior for the intensity function when fitted to individual Ca^{2+} spike sequences we use Q-Q and K-S plots together with simulated spikes and rescaled ISI occurrence plots as outlined in Section 5.1.

We begin by considering ISI distributions inferred from individual spike sequences from the dataset where the applied stimulus was $10\mu M$ glutamate. Since the applied stimulus is constant a natural assumption would be that the spiking rate would mirror this. However, we find that the models with a constant prior for the intensity function are outperformed by the PWC and GP prior. This can be seen in Figure 5.7 where the violins for the Q-Q and K-S plots for the constant prior have a larger

range than the respective violins for the PWC and GP prior. Moreover, the rescaled ISI occurrence plots indicate that the ISIs are dependent for the constant prior — see Appendix B. We see that the slopes resulting from the inhomogeneous exponential ISI distribution are far from the line of $\pi/4$, irrespective of the prior used for the intensity function. We also find that the addition of the refractory period as a parameter for the inhomogeneous exponential ISI distribution improves the resultant slopes. However, the slopes are still far from the line of $\pi/4$ and other ISI models are more suitable. Furthermore, we see that the inhomogeneous log-normal distribution performs poorly. Each of the remaining three ISI distributions — Gamma, inverse Gaussian and Weibull — perform well for both the PWC and GP prior. However, the Weibull ISI distribution with a PWC prior performs the best with slope centered about $\pi/4$ with the narrowest violins for both the slopes of the K-S and Q-Q plots.

We find similar results when considering individual spike sequences obtained from astrocytes challenged with $100\mu\text{M}$ glutamate. The slope summaries of the Q-Q and K-S plots are shown in Figure 5.8. By considering the rescaled ISI occurrence plot and simulated spike sequences — shown in Appendix B — we can rule out the constant prior for the intensity function. Consulting the slopes of the Q-Q and K-S plots we find that the inhomogeneous Weibull distribution with a PWC prior perform the best.

In Figure 5.9 we present the mean of the posterior intensity function distribution and the mean of the posterior ISI parameter distribution when the stimulus applied was either $10\mu\text{M}$ or $100\mu\text{M}$ glutamate. We first consider the intensity functions. Concentrating on the region [100s, 400s] we see that the intensity function is larger when the applied stimulus is smaller. This is surprising as often a larger stimulus leads to larger Ca^{2+} response. One potential explanation could be the larger stimulus depletes resources within the cell, such as the ER. Furthermore, the intensity functions are more likely to converge to zero before the end of the experiment when the stimulus is smaller. Finally, intensity functions vary less when the applied stimulus is larger, i.e. the intensity functions are grouped closer together. Whereas, intensity functions are more likely to take different shapes when the applied stimulus is smaller. For

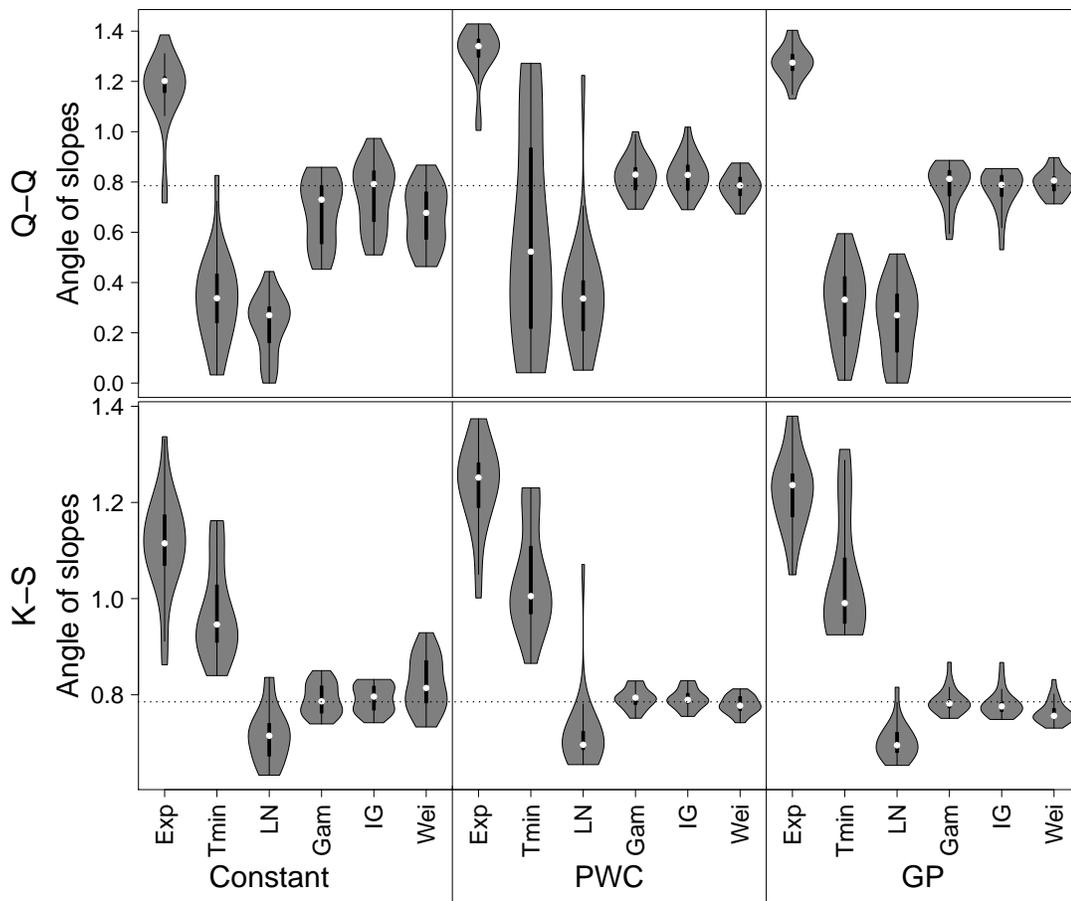


Figure 5.7: Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models using the data of 24 astrocytes challenged with $10\mu\text{M}$ glutamate.

example not all intensity functions have a trough in intensity in $[600\text{s}, 800\text{s}]$ and some intensity function do not follow the decreasing pattern in $[180\text{s}, 400\text{s}]$. This implies that a larger stimulus reduces the variability in the cell's Ca^{2+} response. Indeed, it has been shown that applying larger stimuli to astrocytes leads to a response that is more deterministic, see [135]. Further evidence is provided by looking at the inferred ISI parameter for each cell – shown in Figure 5.9 — where the ISI parameter is larger

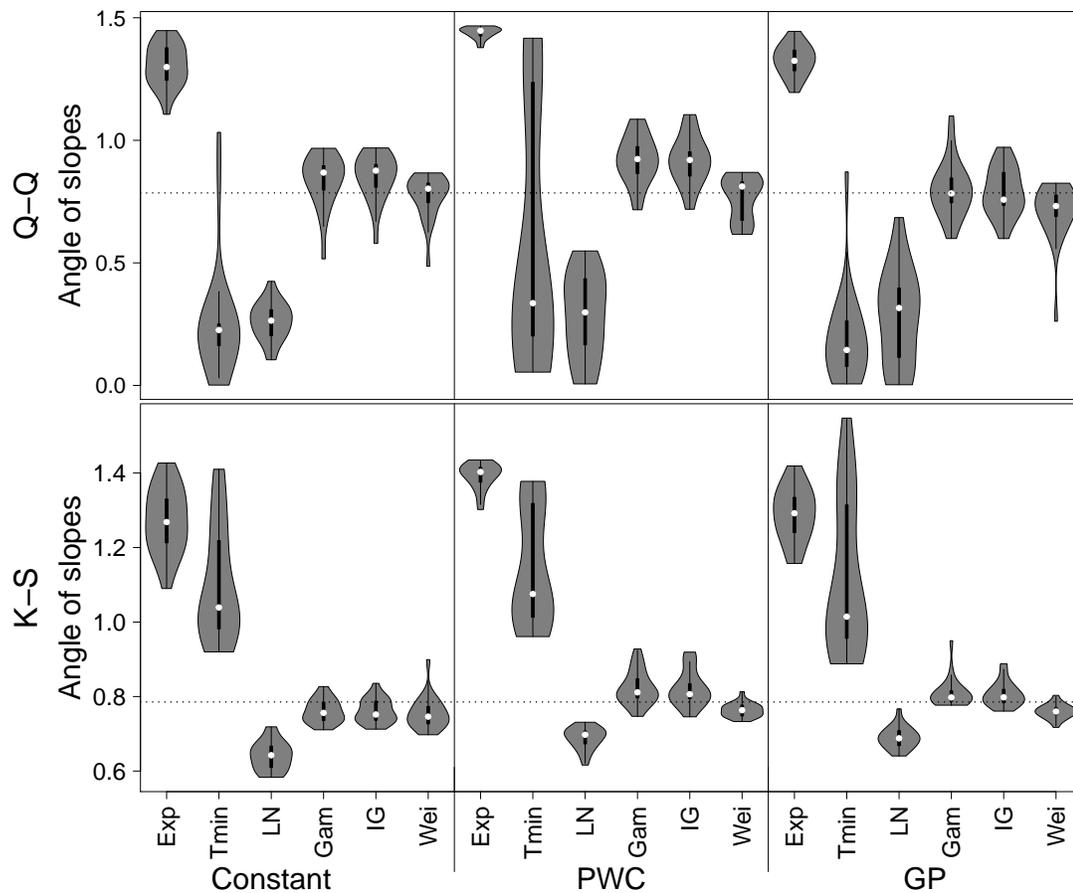


Figure 5.8: Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models using the data of 29 astrocytes challenged with $100\mu\text{M}$ glutamate.

when the applied stimulus is larger. Note that for the inhomogeneous Weibull ISI distribution the larger the ISI parameter the smaller the variance.

In summary, we have found that astrocytes challenged with a constant stimulus react in a time dependent manner. By considering the ensemble of astrocytes tested we found that cells respond to the stimulus by initiating a Ca^{2+} spike immediately. This is followed by a brief — approximately 20s — drop in spiking rate, where we

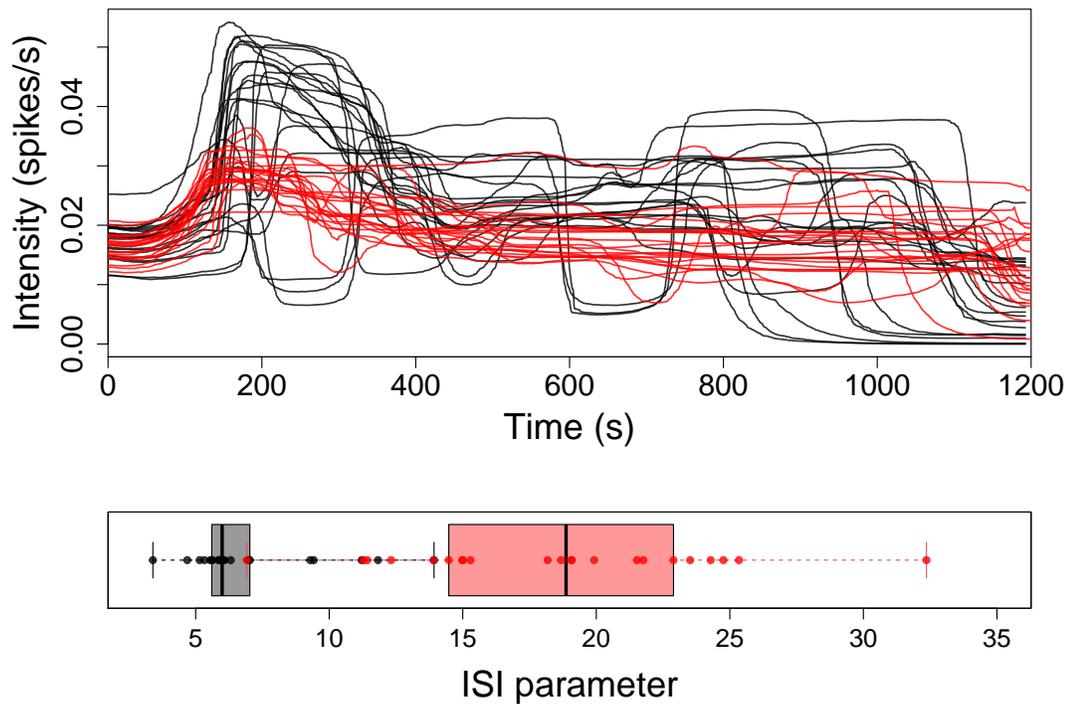


Figure 5.9: Comparison of the posterior Weibull ISI distribution with a PWC prior fitted to individual Ca^{2+} spike sequences from astrocytes cells challenged with $10\mu\text{M}$ (black) and $100\mu\text{M}$ (red) glutamate. Each Ca^{2+} spike sequence comes from a unique astrocyte. For each fit we plot the mean of the posterior intensity function distribution. We also plot the mean of the posterior ISI parameter distribution, where we produce box plots for each stimulus strength.

often do not find a Ca^{2+} spike. Afterwards, the cells begin to exhibit regular Ca^{2+} spiking and over time the spiking rate decreases. We found that a single spiking model does not describe the varying Ca^{2+} oscillations of all the cells suggesting we should treat cells individually. However, we found a common property across most cells where the first Ca^{2+} spike occurs at the onset of stimulus. Moreover, there is period of 20s after the first spike where there is a reduced spiking rate. For the individual fits, the inhomogeneous Weibull distribution best describes the Ca^{2+} spikes of astrocytes, irrespective of the strength of the stimulus. Since the intensity function

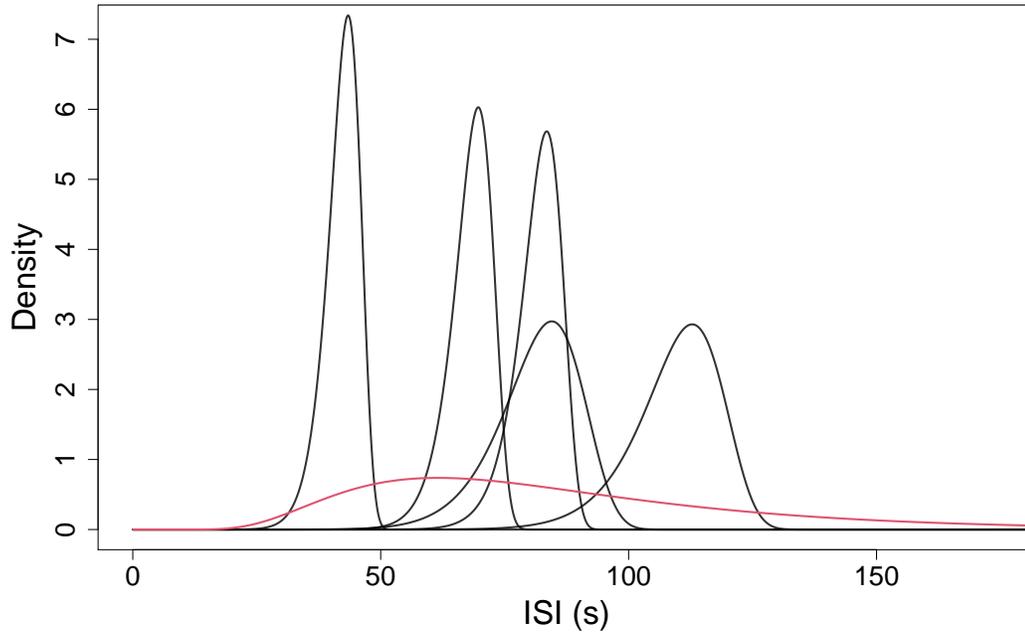


Figure 5.10: Comparison of posterior ISI distributions when approximated by a constant intensity over the region [400s, 1100s]. This is shown for 5 posterior Weibull ISI distributions inferred from individual Ca^{2+} spike sequences (black) and the fit from the whole dataset (red) for cells challenged with $100\mu\text{M}$ glutamate.

is almost constant in [400s, 1100s] for cells challenged with the larger stimulus we can calculate the ISI distribution in the region. This is shown for 5 individual fits and the collective fit in Figure 5.10. We see large variability in the ISI distributions, and that the model fitted from the dataset of Ca^{2+} sequences is spread across the range of all potential cell responses but does not agree with a single individual fit.

Interestingly, the strength of the applied stimulus actually decreases the rate at which the cell's Ca^{2+} oscillations occur when initially exposed. However, the fluctuations occur more regularly and often last longer compared to a weaker stimulus. Therefore, in general we find that using a stronger stimulus weakens the initial Ca^{2+} response to the stimulus, but after the initial transients the Ca^{2+} spikes are more regular and last further into the experiment.

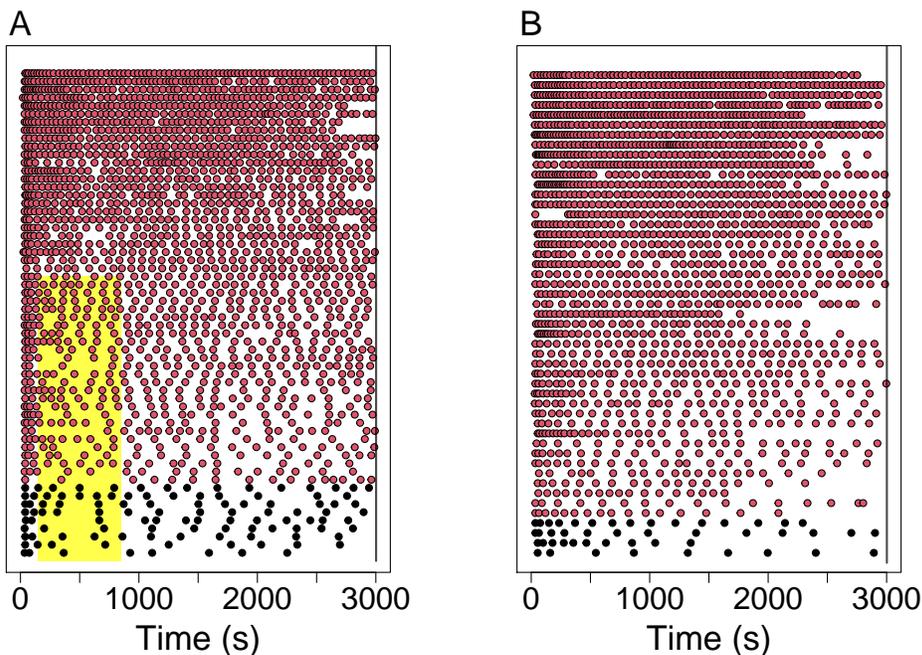


Figure 5.11: Raster of Ca^{2+} spike sequences obtained by stimulating HEK293 cells with $10\mu\text{M}$ and $30\mu\text{M}$ carbachol in (A,B), respectively. We split the dataset into the spike sequences with at least 15 spikes and those with fewer spikes, shown by the red and black points, respectively. The black line shows the length of the experiments. We provide a yellow box in (A) to indicate a region containing fewer spikes than expected.

5.2.2 HEK293 cells

In this section we apply our methods to HEK293 cells challenged with constant stimulus. We have two datasets containing 60 and 49 Ca^{2+} spike sequences where HEK293 cells are challenged with $10\mu\text{M}$ and $30\mu\text{M}$ carbachol, respectively. Both experiments lasted 3000s where the Ca^{2+} concentration was recorded every 5s. The stimulus was applied at 30s. The experimental data was kindly provided by Dr. Colin Taylor, for experimental protocol details we refer the reader to [6].

The Ca^{2+} spike times for both datasets are shown in Figure 5.11. We see that

after the onset of stimulus the cells exhibit Ca^{2+} spikes. For both stimulus strengths we witness more Ca^{2+} spikes at the beginning of the experiment and the rate of the spikes decreases over the first 500s. After this time it is unclear whether the rate of Ca^{2+} spikes plateaus or continues to decrease. The majority of cells express Ca^{2+} spikes over the entire experiment. Observe that the number of Ca^{2+} spikes varies considerably from cell-to cell. For example HEK293 cells challenged with $30\mu\text{M}$ carbachol have between 9 and 94 Ca^{2+} spikes. Comparing the datasets we notice that the number of Ca^{2+} spikes appears larger for cells stimulated with $30\mu\text{M}$ carbachol. Therefore, a larger stimulus strength may lead to a higher spiking rate. We also find that cells challenged with the smaller stimulus with less than 35 Ca^{2+} spikes in total appear to have a smaller spiking rate in [250s, 750s] then after 750s. This is shown by the yellow region in Figure 5.11(A). Cells that have a larger number of Ca^{2+} spikes or challenged with the larger stimulus do not have this behaviour.

To examine if any features are cell type specific we begin by fitting our ISI models to the entire dataset of Ca^{2+} spike sequences. For both datasets we found that the inverse Gaussian ISI distribution has the largest likelihood where we assume that the intensity function has a PWC prior.

In Figure 5.12 we illustrate the posterior ISI distribution fitted to the dataset of Ca^{2+} spike times. This is shown by the mean and 95% credible region of the posterior intensity function distribution and box plots summaries of the posterior ISI parameter distribution. We see that the shape of the posterior intensity function is the same irrespective of the strength of the stimulus. Namely, the intensity is initially close to zero and rapidly increases and peaks at approximately 40s. Then the intensity function gradually decreases over the next 500s and plateaus, at which point the intensity function remains almost constant. In comparison, we find that the intensity function is greater when fitted to the dataset with the larger stimulus.

We see that the posterior ISI parameter distribution is similar for both datasets — marginally smaller for the weaker stimulus. Therefore, the variability in the spiking rate appears similar irrespective of the stimulus strength.

Similar to astrocytes we find that the average response of cells to constant stimulus

does not accurately describe the response from individual cells. This can be seen in simulated spike sequences provided in Appendix B. This is also evident in Figure 5.12, as the 95% credible regions are narrow. Therefore, spike sequences generated from the posterior ISI distribution do not contain the range of spike's per spike sequence that the original data has. Thus, to capture the cell-to-cell variability we fit our ISI models to individual spike sequences.

We use the following parameters to fit our ISI models to individual spike sequences. When the intensity function has a constant prior we choose $\sigma_x^2 = 0.5$ and $\sigma_h^2 = 1$. For the PWC prior we set $k_{\max} = 25$, $\lambda = 10$, $\kappa = \kappa_0 = 1$ and $\mu = 0.5$ where the PWC

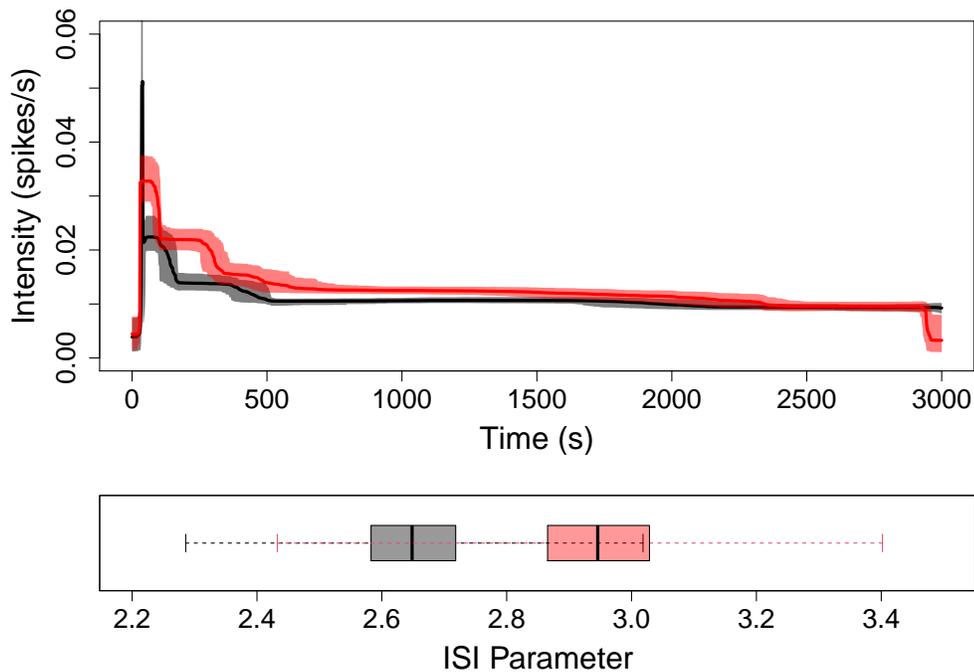


Figure 5.12: Comparison of the posterior inverse Gaussian ISI distributions with a PWC prior fitted from the datasets where the applied stimulus was 10µM (red) and 30µM (black) carbachol, respectively. We present the mean and 95% credible region of the posterior intensity function distributions and box plots of the posterior ISI parameter distributions.

heights have a martingale structure. The parameters for the GP prior were $w = 0.001$, $N = 1200$, $\sigma_v^2 = 1000$ and $\sigma_n^2 = 0$. The length scale l of the GP prior has a prior distribution utilising the PWC approximation, l_{pwc} by $l \sim \text{Uniform}(l_{\text{pwc}}, l_{\text{pwc}} + 5)$. For each choice of intensity function prior we recored 200,000 iterations of the MCMC after an initial burn-in of 200,000.

To ascertain which ISI model best describes the Ca^{2+} spike sequences we consider the slopes of the Q-Q and K-S plots. This is shown in Figure 5.13 when the HEK293 cells were challenged with $30\mu\text{M}$ carbachol. We refer the reader to Appendix B for the Q-Q and K-S plot when the applied stimulus was $10\mu\text{M}$ carbachol. We find that the most suitable model is either the inhomogeneous Gamma or inverse Gaussian ISI distributions, where both have a PWC prior for the intensity function. However, since the inverse Gaussian performed best when fitting the datasets of spike sequences, we choose this distribution to consider the parameters of the individual fits.

In Figure 5.14 we compare the mean of the posterior inverse Gaussian ISI distribution fitted to individual Ca^{2+} spike sequences. In (A,B) the ISI model is inferred from HEK293 cells challenged with $10\mu\text{M}$ and $30\mu\text{M}$ carbachol, respectively. Almost all posterior intensity functions begin with a large spiking rate which reduces quickly. The time taken for the intensity functions to plateau varies cell-to-cell, between [150s, 750s]. HEK293 cells challenged with the larger stimulus tend to take a longer time to plateau than other cells. We see that most cells experience a near constant spiking rate during the experiment after the first 500s. This spiking rate varies between 0.43 and 3.0 spikes per 100s. Therefore, these intensity functions capture the variability of Ca^{2+} response in HEK293 cells. Comparing the two stimulus strengths, we find that the average number of spikes over 100s is 1.25 and 1.46 spikes for cells challenged with $10\mu\text{M}$ and $30\mu\text{M}$ carbachol, respectively. This provides evidence that a larger stimulus may lead to a faster spiking rate. Across the cells we find regions where the intensity function includes sharp peaks and troughs. These correspond to regions where multiple spikes occur quickly or no spikes at all. Towards the end of the experiment the spiking rate of some cells converges towards 0. This shows that the intensity functions capture that some cells do not spike over the entire

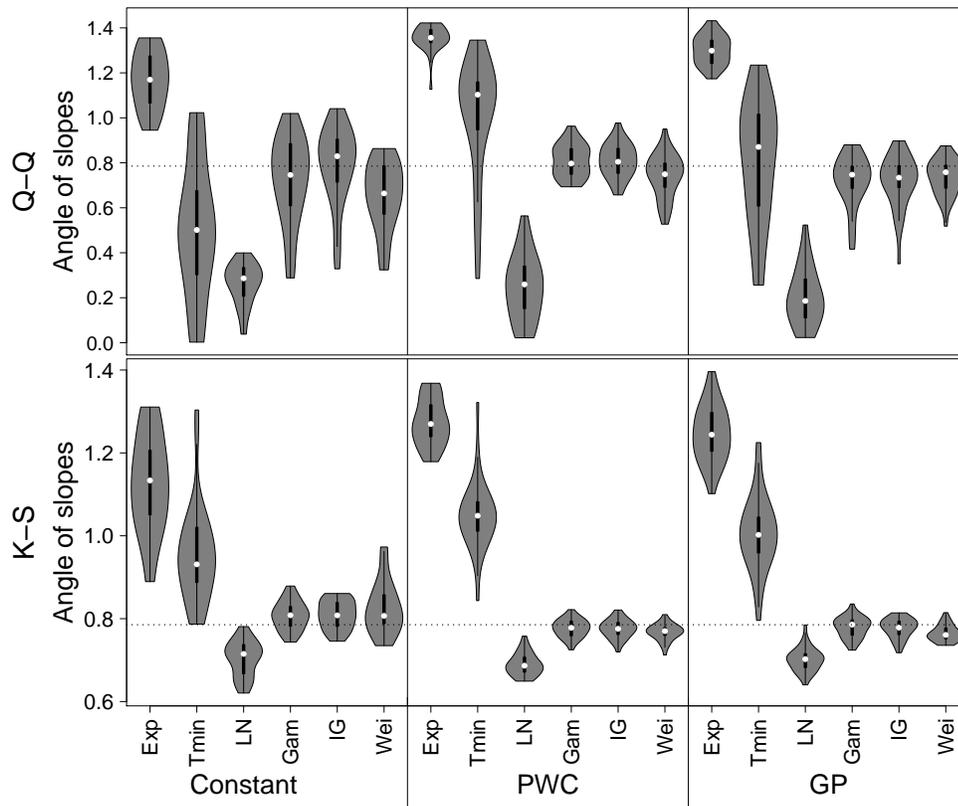


Figure 5.13: Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models using the data of 56 HEK293 cells challenged with $30\mu\text{M}$ carbachol.

experiment's length. Comparing the intensity functions for the two datasets we see that the shape of posterior intensity functions are broadly similar. However, we find that for some cells challenged with the smaller stimulus after the initially peak intensity at the beginning of the experiment there is a region of lower intensity that takes approximately 500s. This does not occur for cells challenged with the larger stimulus. The region of lower intensity could indicate that HEK293 cells require a threshold of stimulus to arrive to the cell before regular spiking can occur.

We see that the mean posterior ISI parameter takes a large range of values between

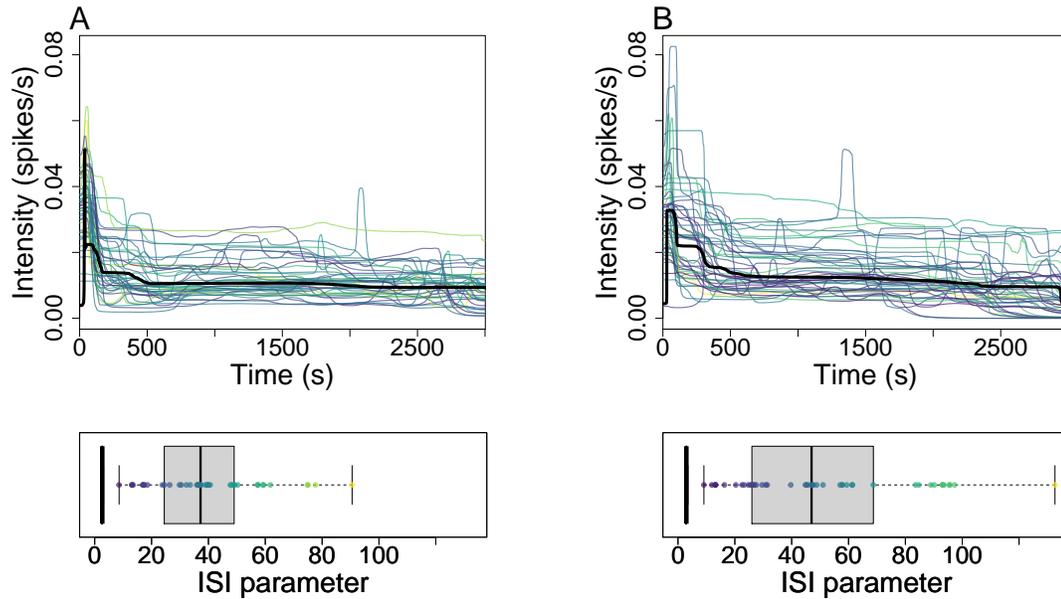


Figure 5.14: Overview of the posterior inverse Gaussian ISI distribution with PWC prior fitted to individual Ca^{2+} spike sequences. The Ca^{2+} spikes come from HEK293 cells challenged with $10\mu\text{M}$ and $30\mu\text{M}$ carbachol in (A) and (B), respectively. For each fit we plot the mean of the posterior intensity function distribution. We also plot the mean of the posterior ISI parameter distribution, where we produce box plots for each stimulus strength. The ISI parameters are coloured from purple to yellow depending on their value. The matching intensity function is shown in the same colour. The posterior mean intensity function and box plot of the ISI parameter fitted from the datasets of Ca^{2+} spike sequences are shown in black — this corresponds to the posterior mean and boxplots shown in Figure 5.12.

8 and 90 for cells exposed to $10\mu\text{M}$ carbachol and between 9 and 132 when challenged with $30\mu\text{M}$ carbachol. By considering the colour of the intensity functions compared to the mean ISI parameter in Figure 5.14 we can see a relationship between the intensity functions and ISI parameters. When the stimulus strength was $30\mu\text{M}$ we find that the smaller ISI parameters (purple) tend to correspond to intensity functions that have a lower spiking rate. This could indicate that the ISIs are more variable

when the spiking rate is smaller. However, for $10\mu\text{M}$ carbachol there is no clear relationship.

For both stimuli strengths we find that the ISI parameter fitted using the datasets of Ca^{2+} spike sequences is smaller than ISI parameters inferred from individual sequences. This indicates there is larger variability of the ISIs is subsumed into the ISI parameter, where a smaller ISI parameter implies a larger variance in the ISI distribution. We find that the posterior mean of the intensity function fitted from datasets approximately aligns with the mean of the individual fits. This shows that the collective fit does not capture the cell-to-cell variability of the Ca^{2+} spikes.

In summary, HEK293 cells appear to respond to a constant stimulus in a time dependent manner. We find the cells respond over two different time scales. The shorter time scale consists of a burst of Ca^{2+} spikes just after the stimulus is first applied. The longer time scale consists of regular spiking whilst the cell is still challenged with the stimulus but far from — often around 500s — the time the stimulus was applied. These two response could lead to the activation of different Ca^{2+} sensitive processes.

We find that the models fitted to the dataset of Ca^{2+} spike sequences does not mirror the variability of the Ca^{2+} response across cells. However, it does find a strong response at the onset of stimulus is common across all cells. We find that our models fitted from individual Ca^{2+} spike sequences captures this variability.

Considering the two datasets our models agree that cells challenged with a larger stimulus on average respond with a larger number of Ca^{2+} spikes. We find that the shape of the intensity functions remains broadly similar between the two stimulus strengths. This shows that cells respond in the same manner irrespective of the strength of carbachol. However, for HEK293 cells challenged with the smaller stimulus we occasionally infer an intensity function whose spiking rate decreases close to zero after the initial burst of activity, before recovering later on in the experiment. This suggests under small stimulus there may be a lag between the burst of Ca^{2+} spikes at onset and regular Ca^{2+} spikes later in the experiment. This could be caused by the cell requiring a threshold of stimulant to arrive to the cell prior to regular Ca^{2+}

spikes. Or with a smaller stimulus, the burst of spikes at the onset of stimulus could deplete other resources in the HEK293 cells that require a period of time to recover.

5.2.3 Summary

In the above sections we have concentrated on cells challenged with a constant stimulus. In each case we have seen that cells respond in a time-dependent manner. For example in both astrocytes and HEK293 cells the Ca^{2+} concentration oscillates quicker at the onset of the stimulus than later in the experiment. This is confirmed in our models since the constant prior for the intensity function performs badly. This shows the importance of using a time-dependent model for the ISI even when the stimulus is constant.

We have found that the mean population response of astrocytes and HEK293 cells to constant stimulus is different. Namely, astrocytes have a region of low spiking rate after the initial spike before rebounding, unlike HEK293 cells. This property gives us a distinguishing feature between Ca^{2+} spikes from astrocytes and HEK293 cells. Furthermore, this could indicate that different elements of the Ca^{2+} signalling toolbox are required for HEK293 cells and astrocytes.

Our models have found that the strength of the applied stimulus affects the Ca^{2+} response in different ways. Increasing the strength of glutamate in astrocytes causes Ca^{2+} spikes to occur less often and with less variability, and the spiking occurs over a longer duration. Whereas, increasing the strength of carbachol in HEK293 cells leads to a larger spiking rate over the entire experiment and the time taken for the spiking rate to plateau increases. This demonstrates that increasing the stimulus strength affects Ca^{2+} spikes dependent on the cell type and the stimulant used.

We have found that the simplest ISI model — the inhomogeneous Exponential ISI distribution — does not accurately describe Ca^{2+} spike sequences. This remains true even with the addition of a refractory period. Therefore we have shown that Ca^{2+} spike sequences require more-complex ISI models such as the inhomogeneous Gamma, inverse Gaussian or Weibull ISI distributions.

5.3 Time-dependent Stimulus

So far we have only considered cells that experience a constant stimulus. In vivo cells experience a plethora of stimulants and inhibitors which arrive to a cell in a time dependent manner. Therefore, it is vital to understand how Ca^{2+} oscillates when a cell is exposed to a time-dependent stimulus. In this section we consider HEK293 cells challenged with three different stimulus profiles: a step-change, waves and pulses. For each we analyse the Ca^{2+} spike times and fit inhomogeneous ISI distributions. We endeavour to find an ISI model that accurately describes the Ca^{2+} spikes, and link back to biological implications.

5.3.1 Step-change stimulus

We now consider HEK293 cells that have been challenged with carbachol in a step-change experiment. This means that a cell is challenged with carbachol of strength $A\mu\text{M}$ for the first half of the experiment at which point the stimulus is exchanged for carbachol of strength $B\mu\text{M}$. We denote this stimulus by $A\mu\text{M} \rightarrow B\mu\text{M}$. We have been kindly provided with four datasets of HEK293 cells challenged with a step-change in carbachol from Dr. Falcke — for the experimental protocol see [6]. In the four datasets the strength of carbachol varies, where the experiments consist of $10\mu\text{M} \rightarrow 50\mu\text{M}$, $20\mu\text{M} \rightarrow 50\mu\text{M}$, $20\mu\text{M} \rightarrow 100\mu\text{M}$ and $50\mu\text{M} \rightarrow 100\mu\text{M}$. The number of HEK293 cells in each dataset is 19, 11, 39 and 14, respectively. The experiments vary in length but all are within the range of 6474s and 7414s. In both experiments, the Ca^{2+} concentration was recorded every 2s.

In Figure 5.15 we illustrate the response of HEK293 cells to a step-change in carbachol. In Figure 5.15(A) we show a Ca^{2+} trace of a single HEK293 cell challenged with a step-change in carbachol. The stimulation profile is shown above the trace plot. We see more Ca^{2+} spikes when the stimulus is first applied and when the stimulus is exchanged. We find that before the exchange of stimulus there were 18 Ca^{2+} spikes and 47 afterwards. Therefore, the larger stimulus strength leads to more Ca^{2+} spikes. This illustrates the well-known phenomenon of frequency encoding, by which the

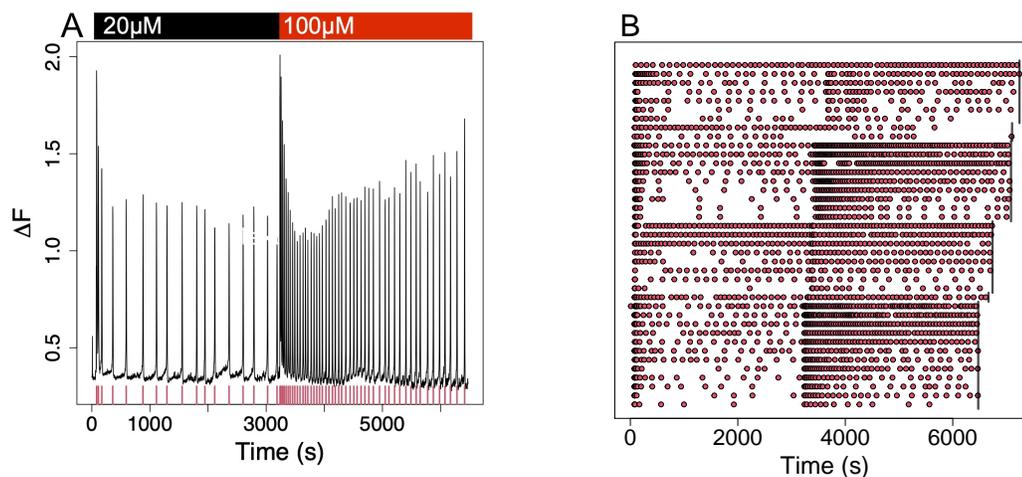


Figure 5.15: (A) Fura-2 fluorescence intensity traces of a HEK293 cell challenged initially with $20\mu\text{M}$ carbachol and then with $100\mu\text{M}$ carbachol, as shown in the bar. The corresponding spike times are shown as red ticks on the x-axis. (B) Raster of Ca^{2+} spike times obtained from HEK293 cells in a step-change experiment where the stimulus strength $20\mu\text{M} \rightarrow 100\mu\text{M}$. The black line represents the length of the experiments across the dataset.

frequency of Ca^{2+} spikes increases with an increase in stimulation strength. The Ca^{2+} spike times for every cell in the $20\mu\text{M} \rightarrow 100\mu\text{M}$ dataset is shown in Figure 5.15(B). We see that most cells display the same characteristics. Namely, more Ca^{2+} spikes when the stimulus is first applied or when the stimulus is exchanged. Moreover, across most cells we find more Ca^{2+} spikes after the stimulus is exchanged for a larger concentration. We find that the total number of Ca^{2+} spikes varies considerably — between 19 and 122 spikes — between HEK293 cells, showing the large cell-to-cell variability. The Ca^{2+} spike sequences are consistent with joining two constant stimulus experiments together.

We want to fit our ISI models to entire datasets of Ca^{2+} spike sequences created from cells that experience the same stimulus. However, as some datasets contain experiments of varying length we cannot fit all the sequences to the same model. This

is because we require the intensity function to be defined over the experiment's length. Therefore, we fit the ISI models to the subset containing the most Ca^{2+} spike sequences with the same experiment length and stimulus exchange times. Consequently, we consider three sub-datasets where the stimulus varies from $10\mu\text{M} \rightarrow 20\mu\text{M}$, $20\mu\text{M} \rightarrow 50\mu\text{M}$ and $50\mu\text{M} \rightarrow 100\mu\text{M}$. They contain 11, 12 and 14 spike sequences respectively. Each of these datasets is fitted using all our ISI models. When the intensity function has a constant prior we set $\sigma_x^2 = 0.5$ and $\sigma_h^2 = 1$. For the PWC prior we choose $k_{\max} = 25$, $\lambda = 10$, $\kappa = \kappa_0 = 1$ and $\mu = 0.5$ where the PWC heights have a martingale prior. The parameters for the GP prior were $w = 0.001$, $\sigma_v^2 = 1000$ and $\sigma_n^2 = 0$. N is taken such that the step size matches the recording rate of the experiment. We infer the length scale of the GP prior using the projection method with $\sigma_l^2 = 0.5$, see Section 3.3.1. For each choice of prior for the intensity function we recored 200,000 iterations of the MCMC after an initial burn-in of 200,000.

Figure 5.16 gives an overview of the posterior ISI distribution fitted using the three datasets. As expected we see that the cells challenged with the larger stimulus strength respond by spiking more frequently. Moreover, we see large peaks in the intensity function when the stimulus begins or is exchanged. As anticipated the intensity function has a similar shape to adjoining two constant stimulus experiments. However, the cells converge to a flat spiking rate quicker when the stimulus is exchanged compared to the onset of the stimulus. We see that the larger stimulus strength used leads to a smaller ISI parameter. This implies cells exposed to a larger stimulus have a larger variability in the ISIs.

We find that the ISI model fitted to entire datasets does not encapsulate the cell-to-cell variability of the individual Ca^{2+} spike sequences. Therefore, we now fit our ISI models to individual Ca^{2+} spike sequences.

The parameters used were the same as when fitting the ISI models to the dataset of spikes sequences. Except in the GP prior where the length scale l has a prior distribution centered via the PWC approximation, l_{pwc} by $l \sim \text{Uniform}(l_{\text{pwc}}, l_{\text{pwc}} + 5)$.

To find which ISI model best describes the step-change in stimulus irrespective of the stimulus strength we include Q-Q and K-S slopes across all datasets. This

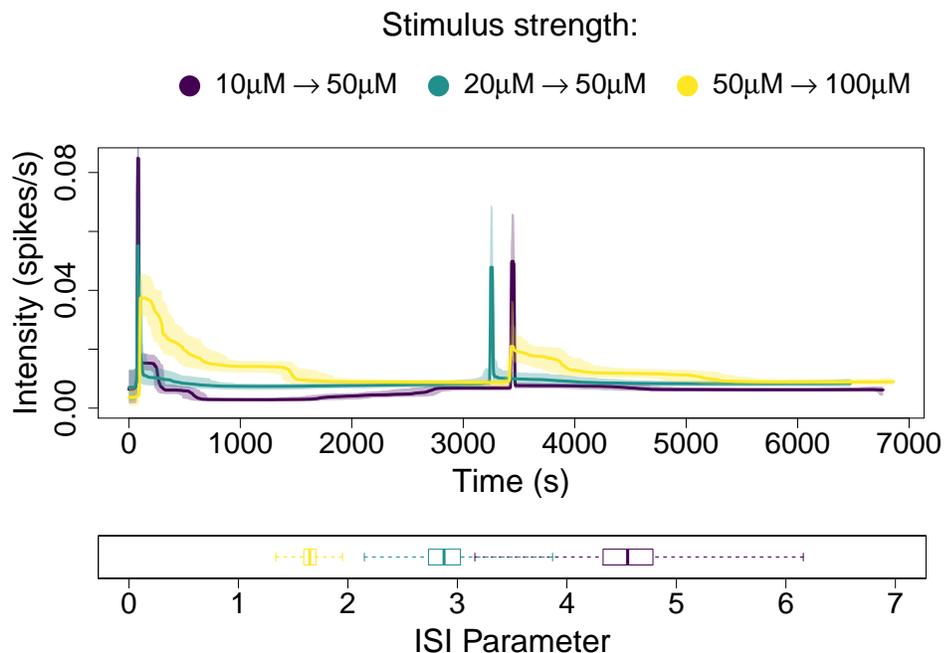


Figure 5.16: Comparison of the posterior of the inhomogeneous inverse Gaussian ISI distribution with a PWC prior fitted from datasets of HEK293 cells under a step-change experiment. We show the mean of the posterior intensity function distribution together with the 95% credible region. We also present box plots of the mean of the ISI parameter over all spike sequences in the datasets.

is shown in Figure 5.17. Similar to analysis with constant stimulus we see that the inhomogeneous Exponential and log-normal ISI distributions do not represent the Ca^{2+} spike times. From the Q-Q and K-S slope plots there appears to be a similar performance across the priors for the intensity function. However, considering rescaled ISI occurrence plots we find dependence between the ISIs for the constant prior. This is shown in Appendix B. Of the remaining ISI models we find that the inhomogeneous Weibull ISI distribution with a PWC prior is most suitable. This is because both the Q-Q and K-S slopes are centered about the line $\pi/4$ and the K-S slopes have the smallest range.

Above we compare the Q-Q and K-S slope plots for all cells challenged with a

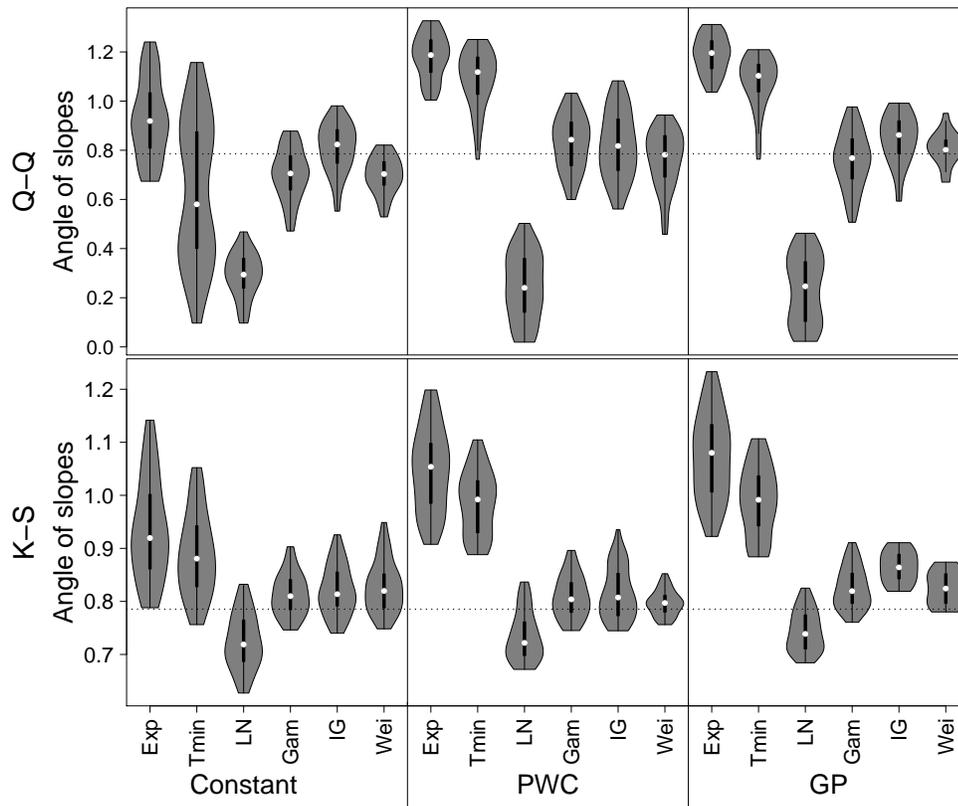


Figure 5.17: Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models using the data of 78 HEK293 cells subject to a step-change experiment with varying stimulus strengths.

step change in stimulus irrespective of the stimulus strength. In Appendix B we shown that the Weibull is also a suitable model when the step change in stimulus is: $10\mu\text{M} \rightarrow 20\mu\text{M}$, $20\mu\text{M} \rightarrow 50\mu\text{M}$ or $50\mu\text{M} \rightarrow 100\mu\text{M}$.

We compare the Weibull ISI distributions fitted from individual Ca^{2+} spike sequences in Figure 5.18. In general, the intensity functions have a similar shape across datasets and individual cells. Specifically, we find peaks in spiking rate at the onset of stimulus and when the stimulus is exchanged and the spiking rate remains

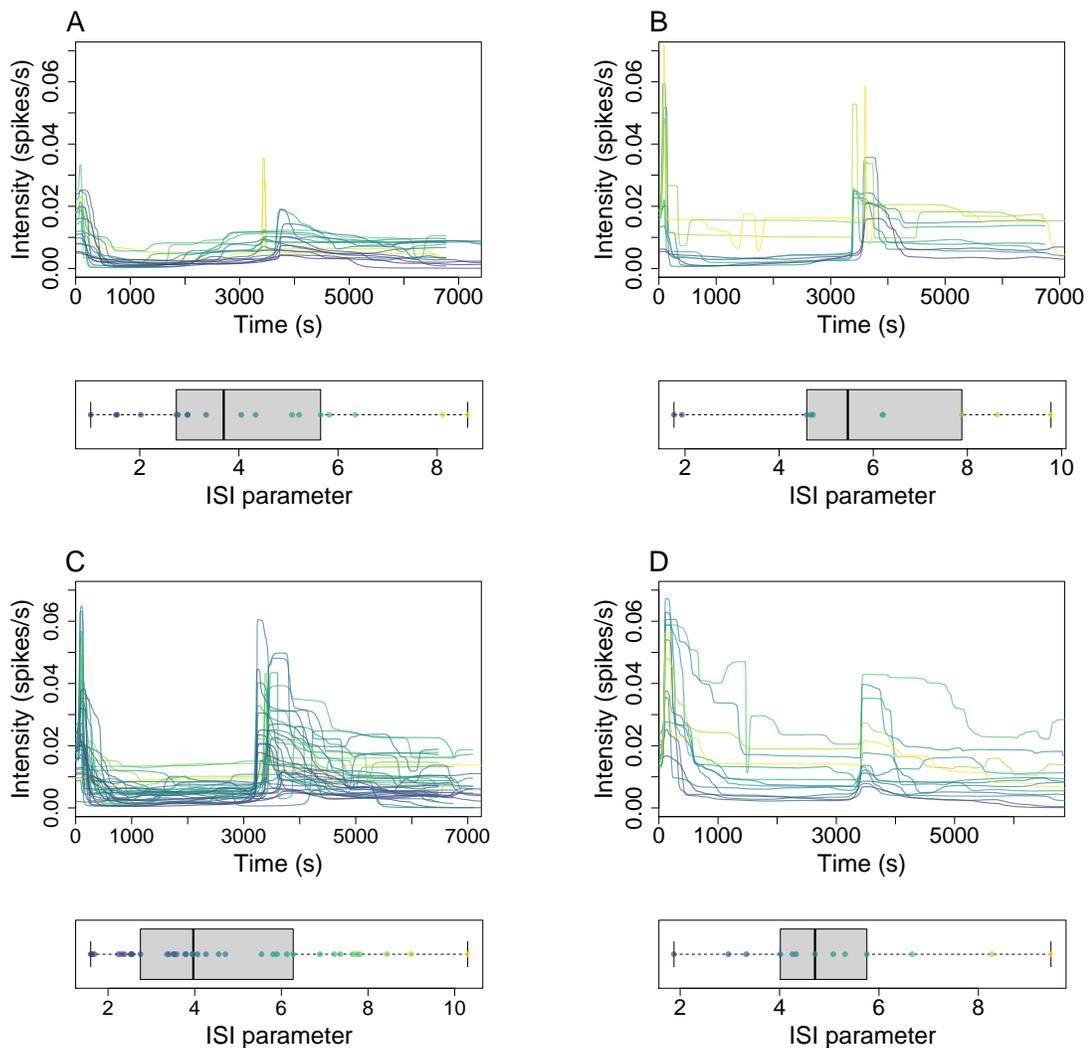


Figure 5.18: Comparison of the posterior intensity function for the inhomogeneous Weibull ISI distribution with a PWC prior fitted to individual Ca^{2+} spike sequences obtained from HEK293 cells under a step-change experiment. (A,B,C,D) correspond to the mean of the posterior distributions fitted to spikes sequences in the datasets where the stimulus strengths were $10\mu\text{M} \rightarrow 50\mu\text{M}$, $20\mu\text{M} \rightarrow 50\mu\text{M}$, $20\mu\text{M} \rightarrow 100\mu\text{M}$ and $50\mu\text{M} \rightarrow 100\mu\text{M}$, respectively.

near constant elsewhere. This is the same shape as found when fitting the datasets to a single ISI model. However, we find a large variability in the strength of the

response, across all datasets. For example when the stimulus is $50\mu\text{M} \rightarrow 100\mu\text{M}$ the largest spiking rate found in [2000s, 3000s] is 15 times bigger than the smallest. We see that cells challenged with a larger stimulus tend to have a posterior intensity functions that are larger in [1000s, 3000s] and [4000s, 6000s]. In other words, cells challenged with a larger stimulus have a bigger spiking rate after the transient periods when the stimulus is first applied or exchanged. We also find that cells challenged with $10\mu\text{M} \rightarrow 50\mu\text{M}$ do not mirror the other datasets up to the stimulus being exchanged. We see that there is initially a peak in the spiking rate at the start of the experiment. Afterwards the spiking rate remains low before recovering after approximately [1500s, 2000s]. Furthermore, we notice that the time taken for the intensity function to plateau varies with the stimulus strength, where the larger the stimulus the longer the time.

The ISI parameter lies between 2 and 10 for the majority of the cells. We see that values taken are similar for each dataset. This suggests that the ISI parameter could be independent of the strength of the stimulus used, when considering individual cells. Notice across all datasets the intensity functions with smaller magnitude tend to have a smaller ISI parameter, i.e the purple intensity functions are the ones with the smallest spiking rate. Therefore, we find that there is more variability in the ISIs when the spiking rate is smaller.

In summary, we have found that our posterior ISI models are synonymous with two adjoined constant stimulus experiments. Similar to HEK293 cells under constant stimulus we find that the larger the concentration of stimulus the larger the Ca^{2+} spiking rate and a lag to regular Ca^{2+} spikes when the stimulus was $10\mu\text{M}$. In other words, the intensity function decreases to close to zero after the initial peak before recovering.

5.3.2 Wave stimulus

In this section we study a single dataset of 63 HEK293 cells that were challenged with carbachol. The carbachol concentration varies over the experiment in three waves, where each lasts 650s. Each experiment was recorded over 3000s, where the

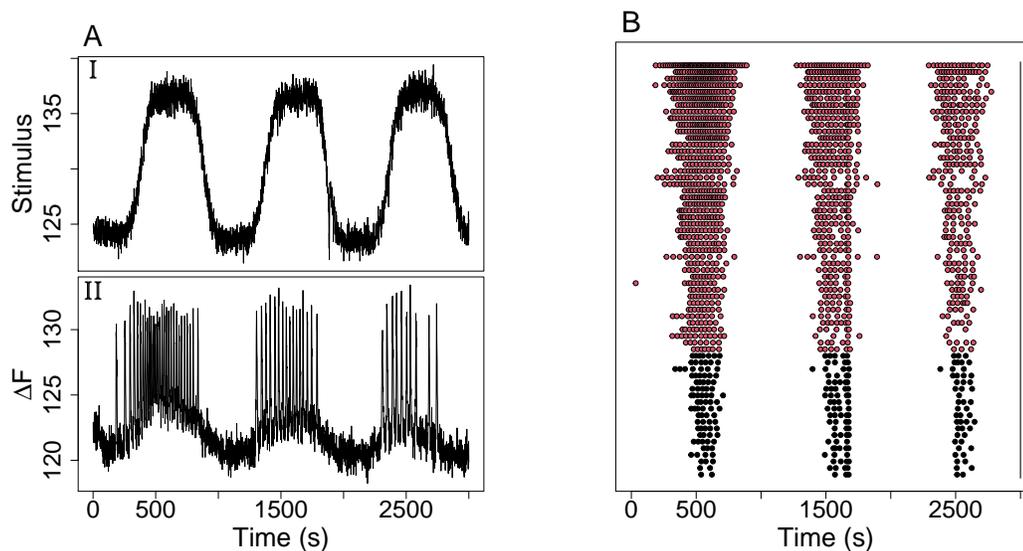


Figure 5.19: (A) An example of the wave stimulus applied to a HEK293 cell (I) and the cell's Ca^{2+} response (II). (B) Raster of Ca^{2+} spike times obtained from HEK293 cells challenged carbachol which varies over three waves. The black line indicates the length of the experiment.

Ca^{2+} concentration was recorded every 1s. The data have been kindly provided by Dr. Bellamy — for the experimental protocol see [16].

In Figure 5.19(A) we show the stimulus (I) and Ca^{2+} response (II) from one of the HEK293 cells. From the stimulus, we find that the level of carbachol oscillates over the experiment in three waves at [300s, 950s], [1300s, 1950s] and [2300s, 2950s]. Each wave has approximately the same amplitude. The Ca^{2+} spikes mirror the applied stimulus with three bursts of Ca^{2+} activity. We find that the bursts begin at the times when the stimulus begins to increase. However, we see that subsequent bursts contain fewer Ca^{2+} spikes — 29, 16 and 10, respectively. This could be due to the fluorescent dye buffering the Ca^{2+} response over time or the cell desensitising to the stimulus over repeated waves.

The Ca^{2+} spike times for each HEK293 cell in the dataset are displayed in Figure

5.19(B). All cells experience Ca^{2+} spikes in three distinct bursts surrounded by intervals with no spikes. We find large cell-to-cell variability both in the number of Ca^{2+} spikes and in the width of the spiking regions.

To explore if there are any common features across all cells we fit the whole dataset to our ISI models. Due to the large cell-to-cell variability we also fit our ISI models to individual Ca^{2+} spike sequences, provided they have at least 16 spikes. The parameters of the ISI models are the same as those used to fit the Ca^{2+} data from a step-change experiment, for the whole dataset and individual spike sequences. To find which ISI model best describes individual Ca^{2+} spike sequences we consider the slopes of the Q-Q and K-S plots, this is shown in Appendix B. We identify the Gamma ISI model with a PWC prior to be the most suitable. When fitting the whole dataset we find that the Gamma ISI distribution also performs well. Therefore, to allow comparisons between the different fits we consider the Gamma ISI model for both. An overview of the posterior Gamma ISI distribution with a PWC prior is given in Figure 5.20. In (A) we illustrate the posterior intensity function distribution from fitting a Gamma ISI distribution to all Ca^{2+} spike sequences. We see that the posterior intensity function mimics the applied stimulus, containing three waves. We see that subsequent waves have smaller amplitudes and widths. We find a large peak in the intensity function at 1670s. This corresponds to a region where some cells experience a large spike in the stimulus. In the raster plot we also see that all cells have a Ca^{2+} spike at 1670s. The mean of the posterior intensity function distribution fitted from individual spike sequence is shown in (B). Again, the intensity function mirrors the applied stimulus consisting of three waves. Furthermore, subsequent waves appear to decrease in amplitude and width. However, the individual fits capture the cell-to-cell variability of the Ca^{2+} spikes, where the width and height of each wave varies substantially over all intensity functions shown. For example at 500s the maximum intensity is 6.5 times larger than the minimum. In Figure 5.20(A) we plot the mean intensity function over all the individual fits — shown in yellow. We find that the collective fit is approximately followed the mean over all individual intensity functions shown, except in the regions [0s, 250s] and [1650s, 1700s]. The difference

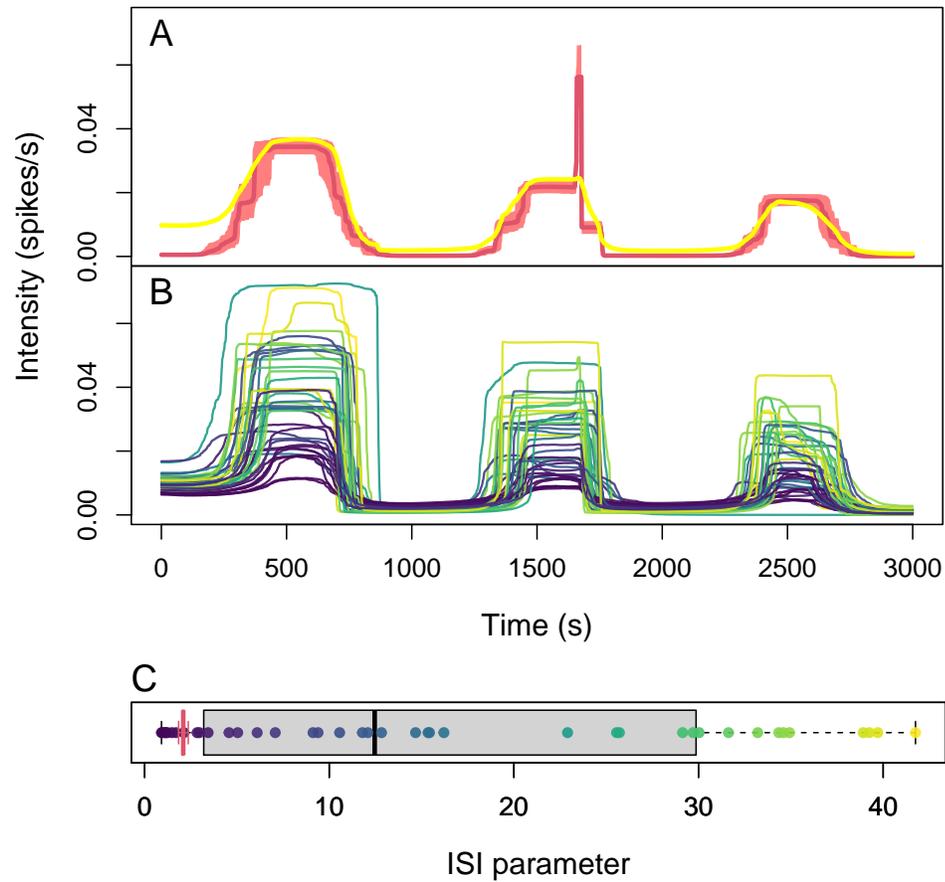


Figure 5.20: Comparison of the posterior of the inhomogeneous Gamma ISI distribution with a PWC prior fitted to individual Ca^{2+} spike sequences and the whole dataset. (B) the posterior mean of the intensity function for each fit from an individual Ca^{2+} spike sequence. The mean of these intensity functions (yellow) is shown in (A) compared with the posterior mean and 95% credible region of the intensity function fitted from the entire dataset (red). (C) comparison of the posterior ISI parameter when fitting the Ca^{2+} spike sequences collectively (red) or individually (grey). The ISI parameters are coloured depending on their size and their corresponding mean intensity function is coloured the same.

in [0s, 250s] occurs due to the martingale prior used for the heights of the PWC — which reduces large jumps in the intensity function — and limited information contained in a single Ca^{2+} spike sequence. In [1650s, 1700s] the collective fit captures that almost all Ca^{2+} cells spike in this region, which leads to a peak in the intensity function. Whereas, a single Ca^{2+} spike sequence does not find this feature. This is due to the limited information a single spike sequence contains compared to an entire dataset. We see that the ISI parameter takes a large range of values in [1, 41]. We find that a smaller ISI parameter corresponds to an intensity function that has a smaller amplitude of the waves — as shown by the colour of the points and lines in Figure 5.20(B,C). In contrast, in the collective fit we find a narrow range for the posterior ISI parameter — as shown by the red box plot at 2.1. Note that a smaller ISI parameter leads to an ISI distribution with a larger variance. Thus, the collective fit's ISI parameter is small to account for the variability in spiking rate across all Ca^{2+} spike sequences. Whereas, when considering individual Ca^{2+} spike sequences the ISI parameter can take a plethora of values. This indicates that some cells spike with minimal variance (large ISI parameters) and others contain a large amount of variability (small ISI parameter).

In summary, we found that the posterior intensity function mirrors the applied stimulus, with three waves of larger spiking rate. We find that the collective fit captures a spike in the intensity at 1670s that the models fitted to individual Ca^{2+} spike sequences do not. However, only the individual fits capture the cell-to-cell variability in the Ca^{2+} response.

We find that although the strength of the stimulus is the same for each wave, the Ca^{2+} response decreases over subsequent waves, as captured in the intensity function. This could be caused by the cell desensitising to the stimulant or the fluorescent dye used to record the Ca^{2+} concentration acting as a Ca^{2+} buffer.

5.3.3 Pulse stimulus

We now consider Ca^{2+} spikes obtained from HEK293 cells where the cells are exposed to pulses of carbachol. A single dataset containing 77 cells was kindly provided

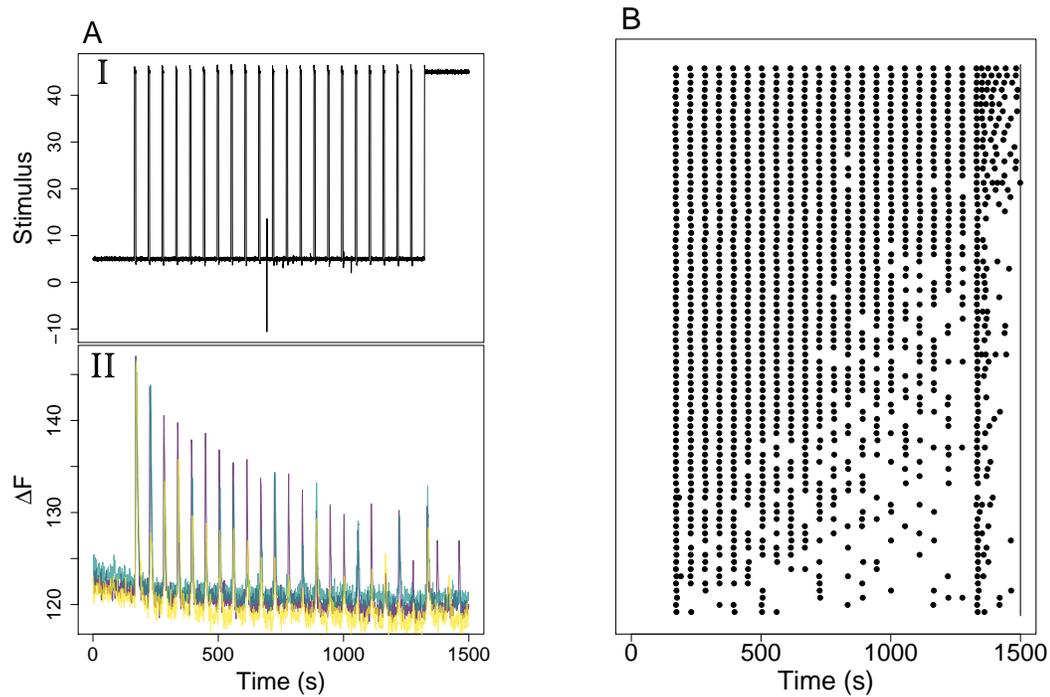


Figure 5.21: (A) The applied pulse stimulus (I) and three trace plots of HEK293 cells response to the stimulus (II). (B) Raster of Ca^{2+} spike sequences obtained from HEK293 cells challenged with pulses of carbachol. The black line represents the length of the experiment.

by Dr Bellamy, for experimental protocol see [16]. In each experiment the Ca^{2+} concentration was recorded every 1s for a total of 3000s.

The stimulus time course is shown in Figure 5.21(AI). We see that the stimulus is initially low at approximately $5\mu\text{M}$ for the first 400s. Between 405s and 1320s there are 21 pulses of carbachol, for each pulse the concentration of carbachol is increased from $5\mu\text{M}$ to $45\mu\text{M}$ for 6s. Finally, the concentration remains at the high level after 1320s until the end of the experiment. In Figure 5.21(AII) we produce Ca^{2+} trace plots for three HEK293 cells. We see that Ca^{2+} oscillations are locked to the times of the stimulus pulses for the three cells. Interestingly, the amplitude of the Ca^{2+} oscillations tend to decrease as the experiment proceeds. Furthermore, the chance of

a Ca^{2+} spike occurring at a stimulus pulse decreases further into the experiment. For example we see that the blue trace only has two spikes in [900s, 1300s], when there are 7 pulses.

Indeed, considering the raster plot of all Ca^{2+} spike times in Figure 5.21(B), we see that the further along the experiment the fewer spikes occur across the whole dataset. When the stimulus remains high at 1320s almost all cells respond with a Ca^{2+} spike. After this time we find a variability in response, where some cells repeatedly spike until the end of the experiment and others only once.

Due to the locking of the spike times it is clear that there is correlation between the spike sequences. Therefore, it is best to fit the ISI models to the dataset containing all Ca^{2+} spike sequences. We do this for all our ISI models. The parameters used were the same as the step-change experiment, except for the PWC. We set $k_{\max} = 60$ and $\lambda = 35$. We have increased k_{\max} and λ because the stimulus has 21 pulses and therefore we expect the intensity function will change at the times that the pulses occur. The inferred intensity function using a PWC prior outperforms the GP and constant priors. This is because the PWC allows for the sharp changes in the intensity function which can capture the quick pulses of the stimulus used to challenge the HEK293 cells. Moreover, the applied stimulus — shown in Figure 5.21(AI) — is PWC itself. Therefore, no matter the choice of the length scale for the GP it will struggle to capture the shape of the intensity function. To visualise the difference we show the mean and 95% credible region of the posterior intensity function distribution for the inhomogeneous Gamma ISI distribution with each type of prior in Figure 5.22(A). We find that only the PWC prior has the flexibility to mirror the pulse stimulus applied to the cells. In Figure 5.22(B) we present 10 simulated spike sequences generated from the posterior Gamma ISI distribution with each prior. We find that only simulated spike sequences from the PWC capture the locking of the Ca^{2+} spikes used to fit the model. Occasionally, we find that the model simulates double spikes in a single pulse which the true data does not contain. We also find that after 1320s — the time of the last pulse in stimulus — the intensity functions of the GP and PWC are consistent. In this case the most suitable model is the inhomogeneous Gamma ISI distribution

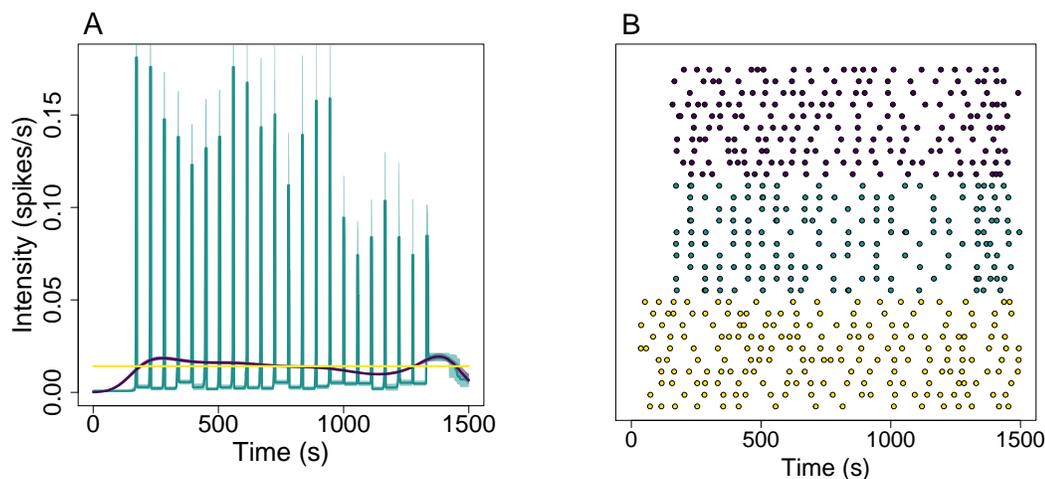


Figure 5.22: (A) The posterior mean and 95% credible regions for the inhomogeneous Gamma ISI distribution with a Constant, PWC and GP prior, shown in yellow, green and purple, respectively. (B) 10 Simulated spike sequences for each of the posterior ISI distributions shown in (A).

with a PWC prior.

At this stage we often fit individual Ca^{2+} spike sequences to our ISI models. However, in this case the Ca^{2+} spike times are locked onto the timing of the carbachol pulses. Therefore, the Ca^{2+} spike sequences do not contain any cell-to-cell variability. Thus, there is no need to fit our ISI models to individual Ca^{2+} spike sequence. Moreover, if we consider a single Ca^{2+} spike sequence from 5.21(B) we do not know that the spike times are locked onto the times of the carbachol pulses — assuming the stimulus is unknown. This property only becomes apparent when inspecting multiple spike sequences. Thus, when fitting our ISI models for a single spike sequence there is not enough information to capture the pulse like behaviour of the intensity function.

In summary, we have found that the Ca^{2+} oscillations of HEK293 cells challenged with pulses of carbachol are best described by the inhomogeneous Gamma ISI distribution with a PWC prior for the intensity function. Doing so leads to a posterior ISI distribution which can faithfully capture the locking of the Ca^{2+} spike

times. Only the PWC is suitable due to its flexibility to quickly change values at the times of the pulses.

5.4 Summary

In this chapter we have shown how inhomogeneous ISI models capture the behaviour of astrocytes challenged with constant stimulus and HEK293 cells under a variety of carbachol stimulation profiles. It is notable across all Ca^{2+} spike sequences studied that a constant intensity function in the ISI distribution does not accurately represent the spikes. Therefore, it is vital that our model incorporates time-heterogeneity. Moreover, we found that the inhomogeneous Exponential ISI distribution with or without a refractory period cannot faithfully represent Ca^{2+} spikes. Thus, we required the added complexity of either the Gamma, inverse Gaussian or Weibull ISI distributions to describe Ca^{2+} spike sequences.

For cells challenged with constant stimulus we have found that both HEK293 cells and astrocytes respond on two different time scales, where we find an initial burst of Ca^{2+} spikes at the onset of stimulus and more regular Ca^{2+} spikes further into the experiments. This is mirrored in our ISI models where we also find that the strength of the applied stimulus acts differently for HEK293 cells and astrocytes, indicating the stimuli affect the cells in different ways.

We found that HEK293 cells challenged with a step-change stimulus is synonymous with joining two constant stimuli together, and this is also shown in the posterior intensity function. Moreover for the waves or pulse stimulus the Ca^{2+} spikes mirror the stimulus, as does the intensity function. This suggests it may be possible to build intensity functions which describe specific stimulus profiles. If achievable, this approach could then be used as an inexpensive way to create realistic spike sequences without requiring to fit the model to experimentally recorded Ca^{2+} spike sequences.

When fitting our ISI models to Ca^{2+} spike sequences we have found that it is important to consider both the fits to individual spike sequences and to the dataset of spike sequences. This is because the individual fits captures the cell-to-cell variability

of the Ca^{2+} response, whereas the collective response can find common properties across cells challenged with the same stimulation profile. For example from Ca^{2+} spikes obtained from HEK293 cells challenged with pulses of carbachol.

Suppose we want to create simulated spike sequences to have the same properties as real Ca^{2+} spike sequences. By fitting our ISI models we have the framework to do exactly this. However, using one of the collective fit or the individual fits does not tell the entire story of the Ca^{2+} dynamics. This is because the collective fit is sensitive to common features across cells and the individual fit to cell-to-cell variability. For example consider the inhomogeneous inverse Gaussian ISI distribution fit to HEK293 cells challenged with a constant stimulus — as shown in Section 5.2.2. We saw that the collective fit captured the captured the behaviour in [0s, 50s] but not the remainder of the experiment. Whereas, the individual fits captured the cell-to-cell variability in [50s, 3000s]. Therefore, neither approach captured all features of the Ca^{2+} spike sequences. Therefore, to create surrogate spike sequences we can take properties both approaches. We do this by joining the intensity function obtained from the collective fit in [0s, 50s] to the individual intensity functions for [50s, 3000s].

In Figure 5.23(A) we illustrate the combined intensity functions fitted to the dataset of Ca^{2+} spike sequences and to individual spike sequences. We see that in [0s, 50s] all intensity functions take the same value — following the collective fit. However, after 50s the intensity functions vary to express the cell-to-cell variability. The intensity functions are coloured to indicate the value of the ISI parameter. Thus, to create a simulated spike sequences we first sample uniformly over the intensity function and ISI parameter pairs. Then we generate the spike sequence using the method outlined in Section 2.2.

In Figure 5.23(B,C) we show the real Ca^{2+} spike sequences used to fit the ISI models and the simulated spike sequences generated from the ISI models. Comparing we find the simulated spike sequences contain all the properties of the original dataset. Namely, a large range of Ca^{2+} spikes per sequence, a higher spiking rate closer to the onset of stimulus, and some cells having fewer spikes in [150s, 750s].

Therefore, given a dataset of Ca^{2+} spike sequences we can create surrogate spike

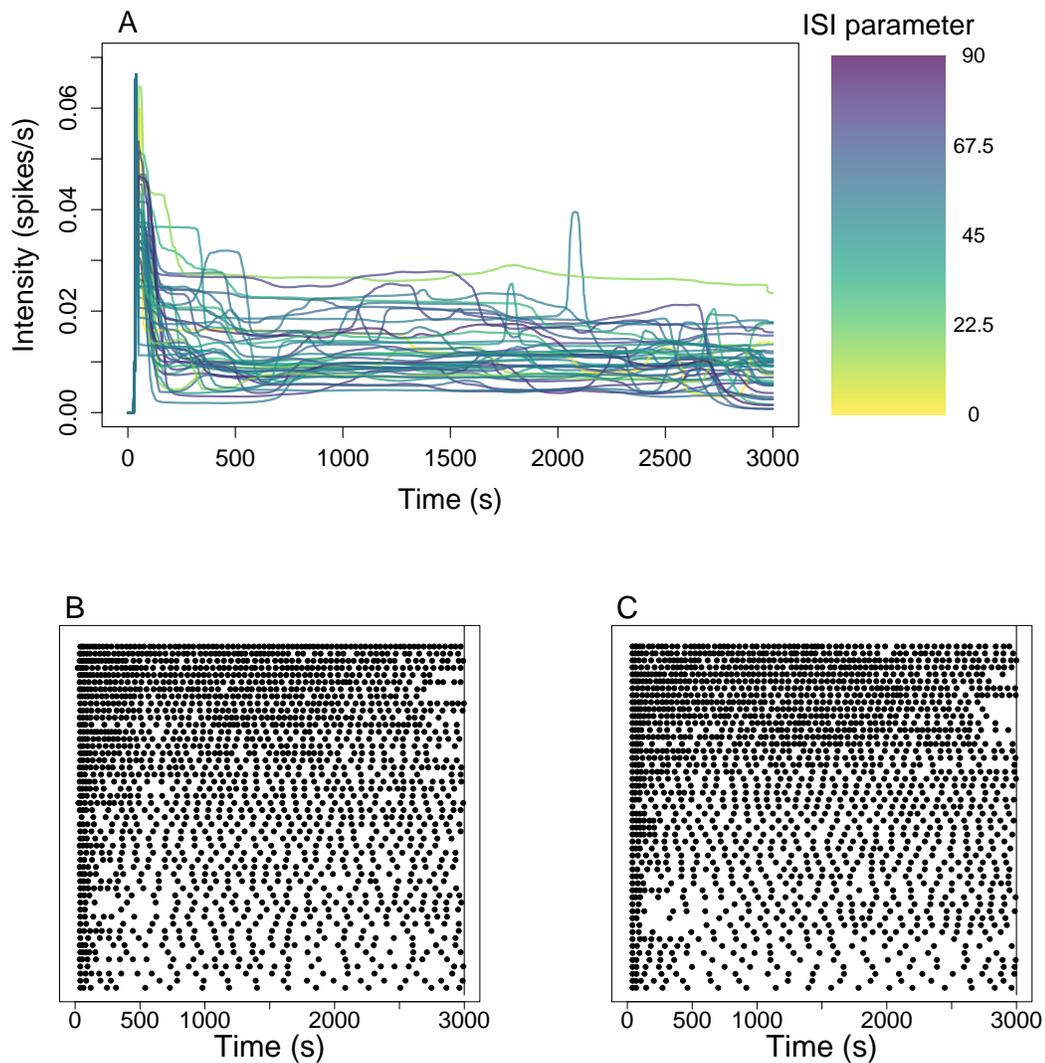


Figure 5.23: Visualisation of simulating spike sequences using ISI distributions fitting to HEK293 cells challenged with 10 μ M carbachol. (A) the intensity function — created by combining individual and collective fits — and ISI parameter pairs used to simulate spike sequence. (B) The Ca²⁺ spike sequences obtained from HEK293 cells, used to fit the ISI distributions in (A). (C) 50 simulated spike sequences from the ISI distributions shown in (A).

sequences with the same properties. This provides a convenient method to generate sizeable datasets of similar spike sequences when only limited real data is available.

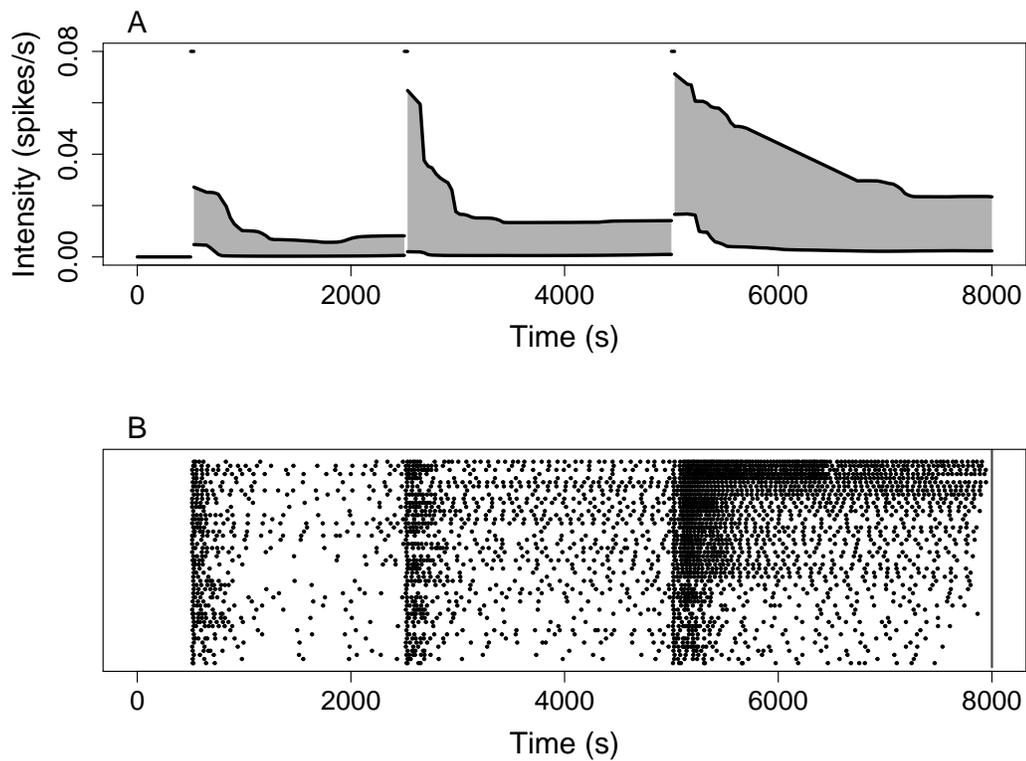


Figure 5.24: Illustration of simulating spike sequences using parameter values from previous or similar experiments. (A) visualisation of the intensity functions required for an experiment where HEK293 cells are challenged with two step changes in stimulus. In particular, cells challenged with $10\mu\text{M}$ carbachol at 500s and the stimulus is exchanged at 2500s and 5000s for $25\mu\text{M}$ and $50\mu\text{M}$, respectively. (B) Raster plot of 50 simulated spike sequences using the intensity functions with the shape shown in (A).

However, for certain cell types or stimulation profiles it is difficult, if not impossible, to record Ca^{2+} spike sequences. For example we may want to record a cell over an extended time region which is experimental infeasible. In this case it would be beneficial to be able to generate surrogate spike sequence that have the expected properties of the cells we want to investigate. In this case, one approach could be to simulate spikes using parameter values found for previous or similar experiments.

For example suppose we require spike sequences that mirror HEK293 cells challenged with multiple step changes in stimulus over 8000s. Specifically, when cells are first challenged with $10\mu\text{M}$ carbachol at 500s, and the stimulus is exchanged at 2500s and 5000s for $25\mu\text{M}$ and $50\mu\text{M}$, respectively. Using the models fit in Section 5.3.1, we see that the intensity function for a step change experiment is consistent with joining multiple intensity functions from constant stimulus experiments. Furthermore, the larger the stimulus the larger the Ca^{2+} response and the longer it takes for the spiking rate to plateau. Putting together this information we create intensity functions as illustrated in Figure 5.24(A). We see that the intensity function can be split into four sections. In $[0\text{s}, 500\text{s}]$ the intensity function is zero because no spikes should occur until the onset of the first stimulus. The remaining three sections $[500\text{s}, 2500\text{s}]$, $[2500\text{s}, 5000\text{s}]$ and $[5000\text{s}, 8000\text{s}]$ correspond to the regions where the stimulus is $10\mu\text{M}$, $25\mu\text{M}$ and $50\mu\text{M}$ carbachol, respectively. We see that each starts with 30s of large intensity (0.08 spikes/s) this corresponds to the onset of the stimulus, where almost always a Ca^{2+} spike occurs. This is followed by an intensity which gradually plateaus over time. Note that the larger the stimulus strength the larger the intensity function and the longer it takes for the intensity function to plateau. Since the Ca^{2+} response varies considerably from cell-to-cell, the intensity function can take numerous paths as illustrated by the shaded regions. For HEK293 cells under a step-change experiment we found the inhomogeneous Weibull ISI distribution performed the best. Therefore, we use the Weibull in this example. Across all step-change experiments the ISI parameter took values in $[1.5, 10]$ across all stimulus strengths with minimal difference across the different experiments. Thus, fitting the 81 ISI parameters from the step-change experiments in Section 5.3.1 to a Gamma distribution produces a $\text{Gamma}(3.64, 0.78)$. Recall that we found a positive relationship between the ISI parameter and the intensity function, where larger intensity functions often correspond to a larger ISI parameter. Therefore, in our ISI models, a large baseline intensity function has a large ISI parameter. For example a function following the top black line in the shaded regions in Figure 5.24(A) would have an ISI parameter close to 10.

Once we have decided on the ISI distribution, intensity functions and ISI parame-

ters we can simulate spikes sequences. We do this by first fixing the ISI distribution to the inhomogeneous Weibull. Then for each simulated spike sequence we sample an ISI parameter and use its value to pick the corresponding intensity function. From the intensity function and ISI parameter we generate the spike sequence. 50 such spike sequences are shown in Figure 5.24(B). We see that the simulated spikes contain the features we expect. Namely, no spikes until the onset of the first stimulus, a larger spiking rate closer to the onset/exchange of stimulus, more spikes for larger stimuli, large cell-to-cell variability.

Constructing intensity functions in the manner described above assumes that the the Ca^{2+} response is invariant to the history of previous stimulation. For example the intensity functions used to mimic the behaviour of HEK293 cells challenged with $50\mu\text{M}$ after 5000s were obtained from experimental Ca^{2+} data where HEK293 cells were challenged at 0s. Therefore, we do not account for any effect of the stimulation between 500s and 5000s, which could potentially result in a diminishing Ca^{2+} spike rate. Future work could investigate whether the assumption is valid, by comparing simulated data to Ca^{2+} spike sequences obtained experimentally.

Therefore, we have seen that our ISI models give a computationally inexpensive method to simulate spike sequences mirroring the Ca^{2+} response of HEK293 cells. Although shown for HEK293 cells, this approach can be used for the Ca^{2+} response of a multitude of cells. This is especially useful to create surrogate datasets for experimentally challenging cells or stimulus profiles. Such datasets could then be fed into 'signalling cascades' to gain insight into Ca^{2+} signalling toolbox.

Clustering Intensity Functions

Previously we have shown how to obtain Ca^{2+} spike sequences from fluorescence time course data of intracellular Ca^{2+} concentration. We have utilised the Ca^{2+} spike sequences to infer the parameters of our model — the intensity function and ISI parameters. Crucially, the inferred intensity function describes the mean spiking rate of the cell. This can be seen in Figure 6.1(A-C) for a HEK293 cell challenged with a step change in carbachol. We threshold the time course data — shown in (A) — to obtain a Ca^{2+} spike sequence — shown in (B). We then fit our ISI model using the Ca^{2+} spike sequence and obtain the posterior distribution for the intensity function. In (C) we show the mean of the posterior intensity function distribution (black line) with the 95% credible interval (grey region). The cell shown belongs to a dataset containing 13 other cells. Thus we can obtain the posterior distribution of the intensity function for each cell individually. In Figure 6.1(D) we plot the mean of the posterior intensity function distribution for each cell in the dataset. This plot shows the variability in Ca^{2+} response for the same cell type challenged with the same stimulus. In this case we see that the mean spiking rate varies between 0 and 0.06 spikes/s. Furthermore, we see that most functions contain a peak in [0s, 1000s]

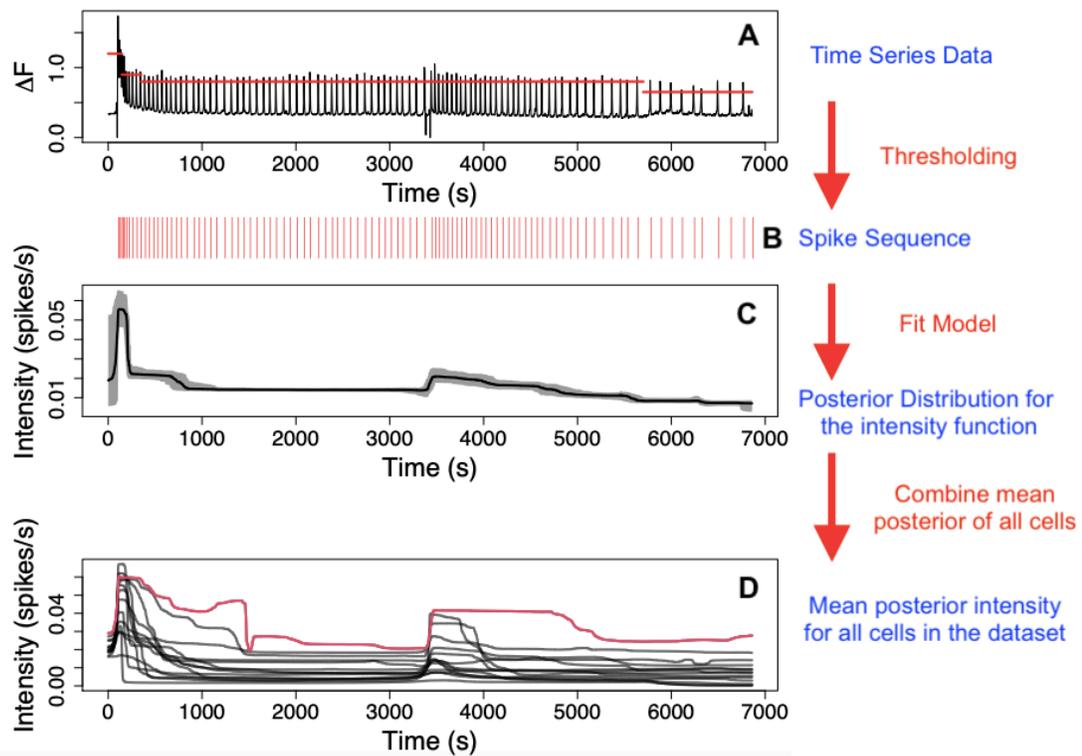


Figure 6.1: An illustration of how we generate intensity functions from experimental data. For a single cell we begin with a Fura-2 fluorescence intensity trace (A) — the changes in the fluorescence of the Ca^{2+} indicator is recorded relative to its basal level (ΔF) — which is thresholded to calculate Ca^{2+} spike times (B). The Ca^{2+} spike times are used as an input into our model which returns the posterior distribution of the intensity function (C). The posterior distribution in (C) shows the mean (black line) with 95% credible interval (grey region). In (D) we show the mean of the posterior intensity function distribution obtained by fitting our model to 14 individual cells.

and another in the region [3400s, 4000s]. However, not all intensity functions have the same behaviour, in particular the red line's 'peaks' appear to last longer and its intensity tends to be higher than the other intensity functions.

In this chapter we investigate how to partition intensity functions — such as those in Figure 6.1(D) — into groups that have similar properties. We want to group the

intensity functions as this will give insight into how cells respond to stimulus. Firstly, consider cells challenged with a constant stimulus. It is often assumed that after some initial transience Ca^{2+} oscillations relax into a stationary regime. This means that the spiking rate converges to a constant. Many models have been developed with this assumption [61, 75]. However our model makes no such assumption on the mean spiking rate. This allows us to consider the Ca^{2+} spike rate over the entire time course (not removing initial transients) and identify trends in this rate. With the inclusion of initial transients we can explore if each cell experiences a similar transient period. For example do we find that the transient period lasts the same length of time for all cells? In addition, what shape does the spiking rate take in the transient period? For example, the spiking rate could gradually decrease or oscillate as it converges to a constant. Moreover, do we find that the spiking rate converges at all? To fit models assuming the Ca^{2+} times are stationary it is sometimes required to remove linear trends in the ISI, which is justified as fluorescent dyes used to record the Ca^{2+} concentration acts as a Ca^{2+} buffer [6]. Thus, we may expect to find that the intensity decreases over time. If this is the case, do all cells show this decrease and is the amount the intensity decrease similar for each cell? If not, this could point to other mechanisms controlling the decreasing intensity such as negative feedback mechanisms found in the cell. Thus, grouping the intensity functions inferred from constant stimulus experiments will test the assumption if a constant spike rate is biologically realistic.

Now consider cells challenged with a time-dependent stimulus. Do we find that a cell's spiking rate mirrors the applied stimulus, and does this happen for all cells? Furthermore, this analysis can be applied to cells challenged with agonist time courses that mimic physiological conditions *in vivo*. Thus, allowing us to investigate cell's spiking rate from a realistic environment.

In the above considerations each cell is challenged with the same stimulus. However, we can also cluster intensity functions where groups of cells are challenged with different stimuli. In this case, if the clusters can distinguish between different stimuli this will give insight into how the stimuli affect the Ca^{2+} concentration. Furthermore,

if we can match cells to their stimulus then we can analysis populations of cells and find which cells in the population receive similar stimuli, which in turn can aid understanding of how signals transmit through populations of cells.

The challenge of grouping together objects is known as clustering. The aim of clustering is to group together objects such that objects in the same group are more similar than objects in other groups [136]. For example points in two dimension are similar if they are close together. As such, the clusters generated depend on the definition of similarity and initial assumptions [137]. Rather than considering the intensity functions let us begin by first simplifying our problem; represent each intensity function by its mean and variance, shown as the points in Figure 6.2. Now our problem involves partitioning these points into groups, depending on the proximity of the points. One method to cluster these points is to use k-means clustering [138]. This involves finding centers (crosses in Figure 6.2) — where each point belongs to their nearest center — optimised such that the distance of all points to their corresponding center is minimised. Thus, Figure 6.2 shows how the points are clustered using this approach with three clusters. One disadvantage of k-means clustering is that the number of clusters is fixed before applying the algorithm. Therefore, k-means clustering cannot be used to automatically detect the optimum number of clusters in the data. Therefore it is often beneficial to use other clustering methods such as: mean shift [139], DSBSCAN [140] or spectral clustering [141]. Each of these methods are easy to apply on two-dimensional data — such as the mean and variance of our intensity functions — but not to functions directly. Therefore, we need to develop an approach suitable for grouping functions, otherwise we will lose vital information such as how the intensity function varies over time.

We mentioned previously that we want to cluster objects that have similar properties, but what does this mean for functions? Consider once again the functions in Figure 6.1(D). In this case we could say functions are similar if their intensities remain close between 0s and 6860s, or if they have the same properties, such as a peak in intensity around 3500s. Recall that the same type of cell challenged with the same stimulus experiences large cell-to-cell variability. For example HEK293 cells

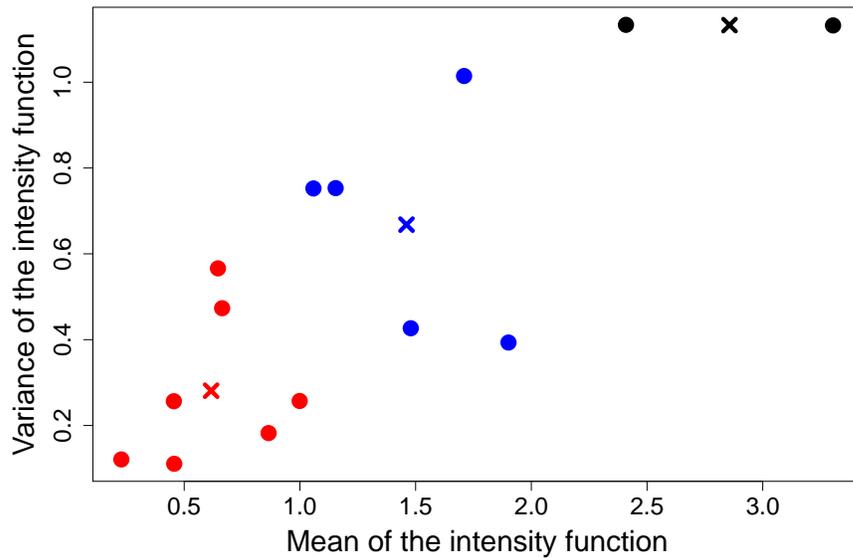


Figure 6.2: Example of clustering using k-means with 3 clusters. Each point on the graph corresponds to a single intensity function. The points are clustered into 3 groups shown by their colour — blue, black and red. The crosses correspond to the center of each cluster.

stimulated with $10\mu\text{M}$ carbachol were found to have between 9 and 87 Ca^{2+} spikes over 3000s. Hence, the mean spiking rate varies widely from cell to cell. With this in mind, we aim to cluster our intensity functions via their features, rather than the absolute rate. By features, we refer to properties such as increasing and decreasing regions and peaks/troughs in the intensity. Any region where a peak occurs — an initial increase in rate followed by a decrease — we shall refer to as a bump. For example, suppose we have 3 intensity functions (black, red and blue lines in Figure 6.3) from an experiment that lasts 3000s with mean spiking rate of 1, 1 and 0.5 respectively. Suppose further that only the first and last functions increase in the region $[0\text{s}, 1000\text{s}]$ and have a bump in the region $[2000\text{s}, 3000\text{s}]$, while the second function is constant. Although the first and third functions (black and blue lines) have significantly different mean rates, we want to cluster these functions in the same

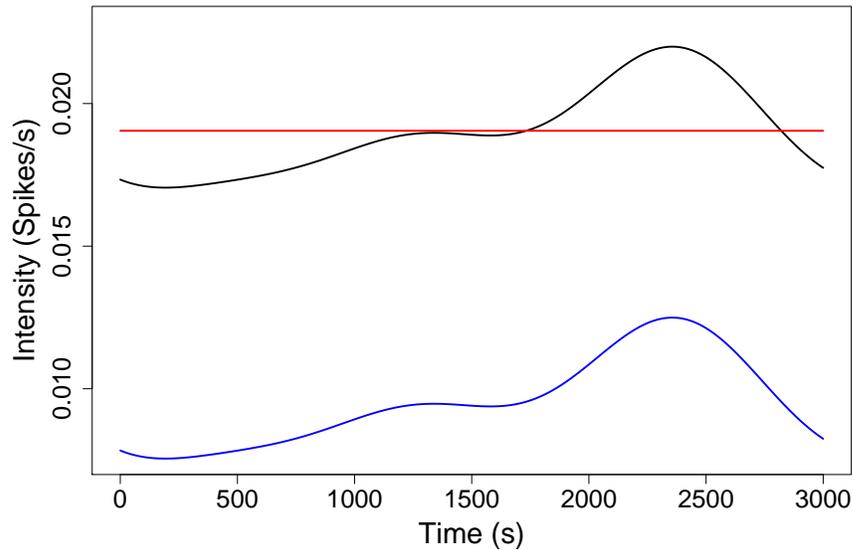


Figure 6.3: Example of clustering methodology applied to three functions. The black and blue lines are grouped together as they have similar shape. Whereas, although the red and black lines have a similar intensity they will not be grouped together as they have different shapes.

group since they have the same shape. Whereas, even though the first two functions (black and red lines) have the same mean firing rate, due to their differing features we do not group these together.

We now require a method that clusters functions with respect to such features. Since functions have infinite dimensions we cannot directly apply the clustering methods mentioned previously. One way to mitigate this challenge is to decompose the functions into a finite number of variables where we can use standard clustering methods. Tilunaite et al. [16] did exactly this to intensity functions calculated from Ca^{2+} oscillations. They decomposed each intensity function into its three leading principal components, using principal component analysis (PCA). They then clustered the components using k-means, mean shift, DBSCAN and spectral clustering. They found that the clustering methods could not detect one unique cluster structure.

Hence, they applied k-means for their analysis due to its speed and simplicity. More recently, Delaigle et al. [142] developed a method specifically designed to cluster functions. Their method consists of applying a weighted k-means on projections of each function onto a space of finite dimension p , where rather than project onto a predetermined basis — such as a principal component basis — the projection is chosen to optimise clustering performance. In practice this is done by first approximating each function via the Haar basis and projecting the coefficients into a space of p dimensions in a manner to optimise clustering. This method has been shown to cluster asymptotically perfectly in some cases. This means if we have two populations in the dataset which differ in their means, then as the sample size increases the proportion of data from one population belonging to the same cluster tends to one.

We tested the methods proposed by Titunaite [16] and Delaigle [142] on surrogate data with attributes similar to those that intensity functions inferred from experimental data contain. We found that both methods perform well in some situations but not others. In particular, both methods fail to cluster a dataset containing constant functions and functions with varying bump placements. In Figure 6.4 we show the results of clustering for this dataset where the two clusters are represented by the colour (red/black) of the lines. Figure 6.4(A) shows that the method proposed by Titunaite clusters functions that have a bump around 1000s, whilst the remaining functions with bumps are grouped with the constant functions. Figure 6.4(B) shows that the method proposed by Delaigle clusters 5 bumps centered around 500s with the constant functions. Thus, neither method managed to group all the functions with bumps together.

Another challenge in clustering is choosing the number of clusters. Some methods automatically choose the number of clusters such as DBSCAN and mean shift. Titunaite et al. [16] set the number of clusters in their case to 5, having noted that differing methods found different number of clusters.

In light of the above, we have developed our own method, specifically designed to decompose functions into their features which we can then cluster. We begin by describing each function by the number and type of features it has. For example

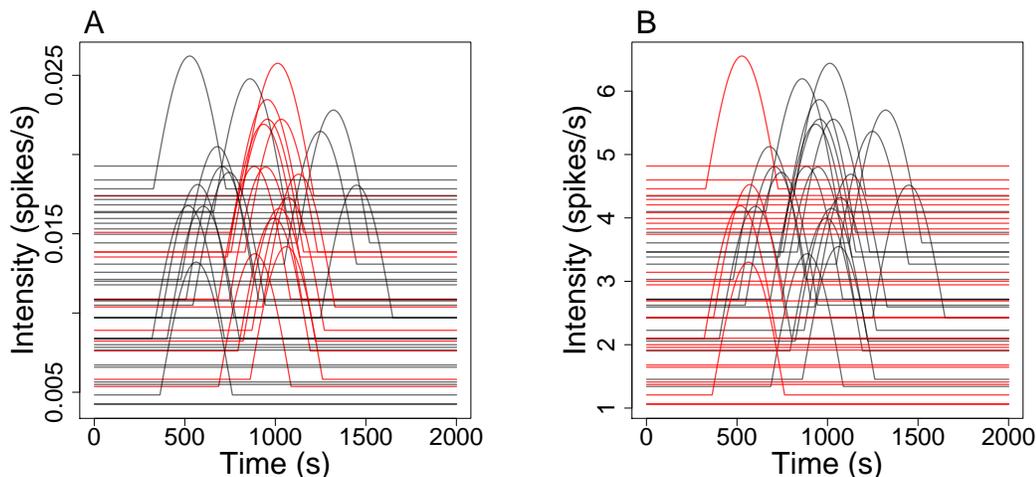


Figure 6.4: Clustering a dataset containing constant functions and functions with a single bump. In (A) we cluster using the method in Tilunaite and (B) Delaigle.

a function could be described by decreasing in $[0s, 100s]$ and a bump in the region $[1000s, 1500s]$. Furthermore, we will provide each feature with a magnitude which describes how large the feature is, which can be used to detect the important features. The benefit of this approach is that it prioritises the shape of the features over the exact value of the function. The clustering is performed by comparing the features of individual functions and is done at the user's discretion, rather than using a clustering algorithm. The advantage of this method is it allows flexibility in the clustering and allows the user to choose the type of clusters. For example we may be interested in the interval $[200s, 500s]$ and we can cluster on the behaviour in this interval. In a different situation we may be interested in the total number of bumps over the entire function, which can then be clustered accordingly.

In the following sections we explain how to extract features of functions by utilising the Haar basis and how to find a feature's magnitude. We then explain how to cluster on an example dataset, before applying our method to data from HEK293 cells stimulated with carbachol.

6.1 The Haar basis

The Haar basis is a sequence of square-like functions that forms an orthonormal basis on $L^2[\mathbb{R}]$ — the space of square-integrable functions on \mathbb{R} . Originally, the basis was put forward by Alfred Haar in 1909 as an example of an orthonormal basis on $[0, 1]$ [143]. The basis was the first known example of a wavelet — a series of mathematical functions that cuts up data into different frequency components — although this concept was only invented in 1982, 73 years after Alfred Haar first constructed the basis [144].

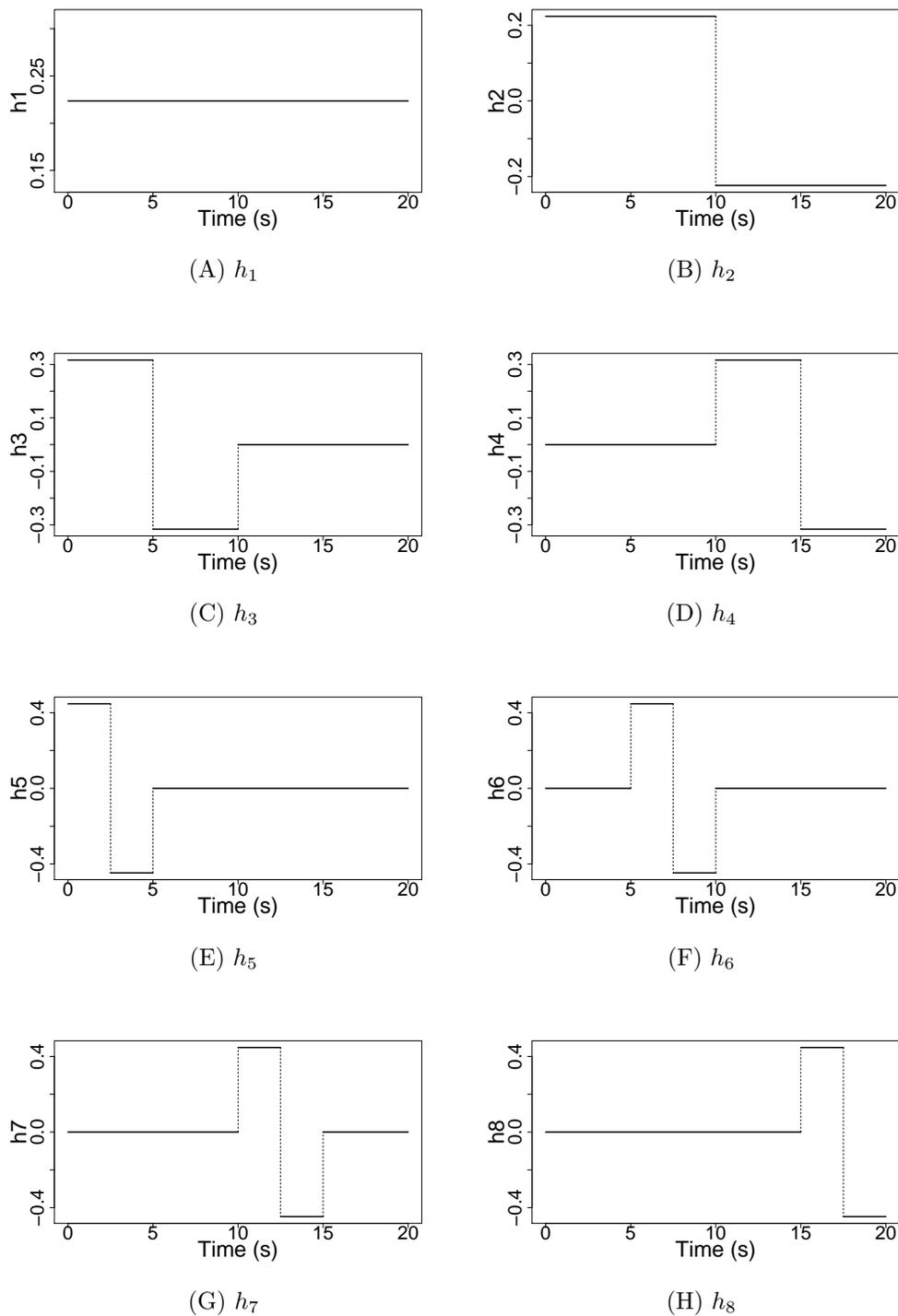
We were motivated to use the Haar basis because it was used by Delaigle et al. [142] in their approach for clustering functions. They found that the Haar basis captures both local and global trends without requiring a large number of bases. We also find that the formation of the basis is well suited to finding locations of features, as described in the next section.

Since we are only considering functions on bounded intervals, we derive the Haar basis on the interval $[0, T]$ equipped with the L^2 -inner product $\langle f, g \rangle = \int_0^T f g \, dt$ for $f, g \in L^2([0, T])$. The basis consists of functions formed by scaling and shifting the Haar mother function,

$$\psi(t) = \begin{cases} 1 & 0 \leq t < 0.5, \\ -1 & 0.5 \leq t < 1, \\ 0 & \text{otherwise,} \end{cases}$$

and a constant function [143]. Since we are forming an orthonormal basis we require the bases to be normalised and orthogonal. This means that for bases h_i and h_j we require $\langle h_i, h_j \rangle = \delta_{i,j}$, where $\delta_{i,j}$ represents the Kronecker delta. The bases will consist of stretching the Haar mother such that each base is zero everywhere except for a region of length $T/2^a$ for some $a \in \mathbb{N}$. Thus, to satisfy the normality condition each function will be scaled by a factor of $\sqrt{2^a/T}$.

We will briefly explain how the first 8 bases are determined. The first base h_1 is defined as the constant function taking value $1/\sqrt{T}$. The second base h_2 is the Haar mother function stretched to take non-zero values on $[0, T]$ and scaled by $1/\sqrt{T}$,

Figure 6.5: The first 8 Haar bases on the interval $[0s, 20s]$.

$h_2 = (1/\sqrt{T})\psi(t/T)$. By definition this is a step function with height $1/\sqrt{T}$ in $[0, T/2]$ and $-1/\sqrt{T}$ in $[T/2, T]$. The next two bases are $h_3 = \sqrt{2/T}\psi(2t/T)$ and $h_4 = \sqrt{2/T}\psi((2t - T)/T)$. These are scaled versions of the Haar mother function taking non-zero values only the intervals $[0, T/2]$ and $[T/2, T]$ respectively. The next 4 bases are created in the same manner by splitting the function into 4 equal regions $h_5 = (2/\sqrt{T})\psi(4t/T)$, $h_6 = (2/\sqrt{T})\psi((4t - T)/T)$, $h_7 = (2/\sqrt{T})\psi((4t - 2T)/T)$ and $h_8 = (2/\sqrt{T})\psi((4t - 3T)/T)$. The graphs of these 8 functions are shown in Figure 6.5, where T was taken to be 20s. From these graphs we can visualise the orthogonality of the bases. Consider two bases h_i and h_j . If the non-zero regions of the two bases do not overlap — for example h_4 and h_5 in Figure 6.5(D,E) — then their product $h_i h_j = 0$. Otherwise, if the bases do overlap the region that is non-zero in their product is just a multiple of the finer base. This can be seen in Figure 6.5 (C,E) for bases h_3 and h_5 , where the product of these two functions is a multiple of h_5 . By construction $\int_0^T h_i dt = 0$ for $i > 1$, hence if h_i and h_j do overlap with h_j finer than h_i we get $\langle h_i, h_j \rangle = \int_0^T h_i h_j dt = \int_0^T C h_j dt = 0$, for some constant C .

From this procedure notice that we generate the bases in levels, starting with $\{h_1\}$, then $\{h_2\}$, $\{h_3, h_4\}$, $\{h_5, h_6, h_7, h_8\}$ and so on. We label these levels 0, 1, 2 and 3 respectively. From Figure 6.5 we see that each level k consists of breaking the space into $l = 2^{k-1}$ equal regions. Consider the k th level, of which we generate l bases g_1, \dots, g_l . Each base g_i only takes non-zero values in a region of length T/l ; for g_1 this region is $[0, T/l]$, for g_2 it is $[T/l, 2T/l]$ and so forth. The non-zero region consists of a scaled version of the Haar mother function, where the first half of this region takes the value $\sqrt{l/T}$ and the second half has value $-\sqrt{l/T}$. More precisely we get

$$g_i = \sqrt{\frac{l}{T}} \psi\left(\frac{lt - (i-1)T}{T}\right).$$

In respect to the Haar basis the bases from the k th level corresponds to the bases h_{l+1}, \dots, h_{2k} . In other words $h_{l+i} = g_i$. Since the sequence of Haar bases $\{h_i\}_{i=1}^{\infty}$

forms an orthonormal basis in $[0, T]$, any function $f(t)$ can be expressed as

$$f(t) = \sum_{i=1}^{\infty} \alpha_i h_i,$$

where α_i is the coefficient of the Haar base h_i , for $i \in \mathbb{N}$.

Naturally, this poses the question: given a function $f(t)$ how do we calculate the Haar coefficients? Since our basis is orthonormal we can calculate the coefficient α_j of the base h_j by considering the inner product of f and the base h_j

$$\begin{aligned} \langle f, h_j \rangle &= \sum_{i=1}^{\infty} \alpha_i \langle h_i, h_j \rangle, \\ &= \sum_{i=1}^{\infty} \alpha_i \delta_{i,j}, \\ &= \alpha_j. \end{aligned}$$

However, in practice the functions we consider are discretised. In particular, we have a time grid $\mathbf{t} = \{t_i\}_{i=0}^R$ where $t_i = iT/R$, where R is the number of equal steps we split $[0, T]$ into. Therefore, we obtain the discretised function of interest $\mathbf{f} = \{f_i\}_{i=0}^R$ and discretised bases $\mathbf{h}_j = \{(h_j)_i\}_{i=0}^R$ for $j \in \mathbb{N}$, where $f_i = f(t_i)$, $(h_j)_i = h_j(t_i)$, and R large. In this case we choose to find the coefficient for the j th base α_j by

$$\begin{aligned} \alpha_j &= \langle f, h_j \rangle, \\ &= \int_0^T f(t) h_j(t) dt, \\ &\approx \sum_{k=0}^N \frac{T}{R} f_k (h_j)_k, \\ &= \frac{T}{R} \sum_{k=0}^N f_k (h_j)_k. \end{aligned}$$

Other numerical integration method exists, for example using a trapezoid rule or Simpson's rule. However, with R large we found our method to be sufficient.

We want to approximate f by calculating the coefficients for the first $q = 2^a$ Haar bases for some $a \in \mathbb{N}$. Rather than calculate each base individually using the formula above, we can calculate the bases by using matrices. To calculate the coefficients

$\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_q)$ we first combine the discretised Haar bases into a basis matrix \mathbf{B} — of dimension $(R + 1) \times q$ — where the i th column of \mathbf{B} is \mathbf{h}_i . We then calculate the Haar coefficients by $\boldsymbol{\alpha} = (T/R)\mathbf{f} \times \mathbf{B}$, which is analogous to the computation above.

We can visual the Haar approximation compared to the original function by computing $\mathbf{f}_{\text{approx}} = \sum_{k=1}^q \alpha_k \mathbf{h}_k = \boldsymbol{\alpha} \times \mathbf{B}^T$.

Let f be a draw from a GP on the interval [0s, 20s] with length scale 3 and step size 0.01s, shown as the black line in Figure 6.6. We then approximate f by decomposing it into 32 Haar bases. The coefficients of the first 16 Haar bases can be seen in Table 6.1.

Base	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8
f	-0.39	-5.54	2.94	-1.14	3.23	-2.76	-0.03	-0.34
Base	h_9	h_{10}	h_{11}	h_{12}	h_{13}	h_{14}	h_{15}	h_{16}
f	0.68	1.39	-0.32	-1.26	-0.13	-0.08	-0.25	-0.03

Table 6.1: The coefficients of the Haar bases for function f drawn from a GP shown in Figure 6.6.

In Figure 6.6, we plot the true function (black line) along with the Haar approximation for $q = 2, 8$ and 32 (red, orange and blue lines respectively). We see that by increasing q the approximation improves. Note that where the Haar approximation is constant, this equals the mean of f in this region.

6.2 Extracting features

We will now explain how to use the Haar basis to extract features from a function. We begin by expressing the function f , defined in the interval $[0, T]$, in terms of a Haar basis with q bases

$$f \approx \alpha_1 h_1 + \dots + \alpha_q h_q,$$

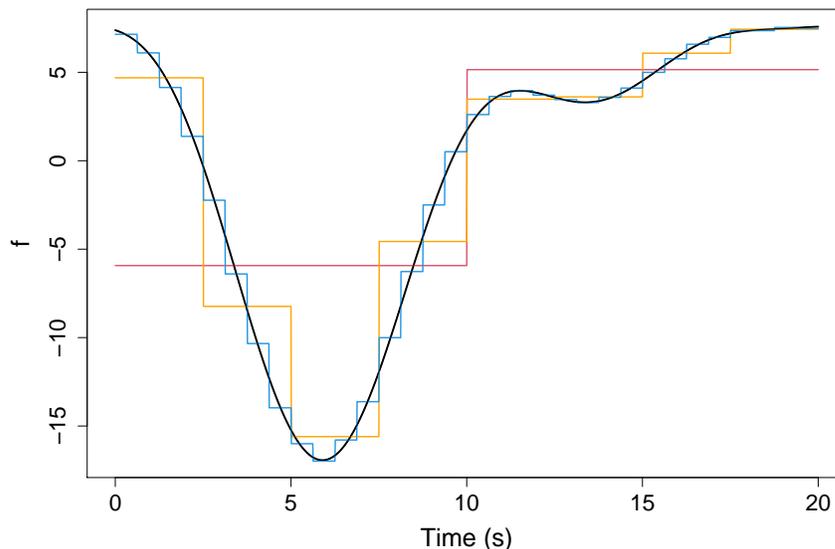


Figure 6.6: Approximating a function via the Haar basis. The function (black line) is approximated by using 2 (red), 8 (orange) and 32 (blue) Haar bases.

where for $i = 1, \dots, q$ the α_i are the coefficients of the Haar bases h_i . The advantage of the Haar basis is that the sign of the coefficient can inform us whether the function is increasing or decreasing in the corresponding interval.

For example consider the base h_7 , which is shown in Figure 6.5(G) for $T = 20$. We see that the base is zero everywhere except for the interval $[T/2, 3T/4]$. In this interval h_7 is positive in $[T/2, 5T/8]$ and negative in $[5T/8, 3T/4]$. Thus, if $\alpha_7 > 0$ then $\alpha_7 h_7$ would have the same shape as h_7 and it appears that the function would be decreasing in $[T/2, 3T/4]$. Whereas, if $\alpha_7 < 0$ then $\alpha_7 h_7$ is similar in shape to the reflection of h_7 in the x-axis and is negative in $[T/2, 5T/8]$ and positive in $[5T/8, 3T/4]$. This implies that the function is increasing in $[T/2, 3T/4]$. It is important to note that this is true only if the function does not vary too much in these intervals. For example consider the approximation with two Haar bases (red line) in Figure 6.6. By the above reasoning we would say the function is increasing in $[0s, 20s]$, which does not capture how the function truly varies in this region. Hence, we will only consider

the bases on the finest intervals because the function will vary less on those intervals.

We build on this idea to calculate the features of functions. The method is explained below in 5 steps.

Step 1: Consider the appropriate bases. We have q bases and each base informs us on the behaviour of the function in a different interval. Recall that q must be a power of 2 and contains η levels. Thus we consider only bases which inform us of details on the smallest intervals, i.e the bases in the η th level. The η th level corresponds to the final $p = q/2$ bases. Thus, set $\mathbb{B} = \{p + 1, \dots, q\}$ the index of the bases to be considered. For example if $q = 16$, we consider the last 8 bases and set $\mathbb{B} = \{9, 10, \dots, 16\}$.

Step 2: Find the important bases. We need to decide which of the appropriate bases informs us on potential features. For a base to contribute to a feature we need the function to be substantially increasing or decreasing in the corresponding interval. An alternative view is that we need to remove the bases whose intervals remain almost flat, as this will not lead to a feature. Thus, we choose a threshold σ_{thres} such that we remove index i from \mathbb{B} if $|\alpha_i| < \sigma_{\text{thres}}$.

Choosing σ_{thres} can be a difficult task. The coefficients of the haar bases scales with the function's range. For example consider a new function $g = 10f$ and approximate g with q Haar bases giving $g \approx \sum_{i=1}^q \beta_i h_i$. Since g is just a scaled version of f we get $\beta_i = 10\alpha_i$ for all i . Hence the threshold value for function f may not be appropriate for g . One approach to avoid this issue would be to scale the functions to a similar range before assessing features. However, this could cause insignificant features to become enhanced. Another issue arises from the choice of q . The larger we take q the smaller the coefficients are due to the factor of 2 in the definition of the bases. With the above taken into account, we find that the threshold should depend on the range Δ and the number of bases q . Thus, we often take $\sigma_{\text{thres}} = \Delta/10q$, which performed well on simulated data, with q either 16 or 32. Note that this would lead to different thresholds been applied to different functions. In some cases, it may be preferable to threshold the functions with the same value for consistency.

Step 3: Check for peaks. In step 2 we removed bases that do not correspond

to a significant increase or decrease. However, a flat interval could be interpreted as a peak of a bump, which is a feature we want to extract. To include a peak the function must be increasing prior to the peak and decreasing afterwards. Thus, to check whether we need to add a base back into \mathbb{B} we require the prior base's coefficient to have negative value (an increase before) and the base after to have positive value (decrease afterwards). For example suppose base i was removed from \mathbb{B} after thresholding. If $\alpha_{i-1} < -\sigma_{\text{thres}}$ and $\alpha_{i+1} > \sigma_{\text{thres}}$ then we add i into the set B .

Step 4: Split \mathbb{B} into groups. Next we split \mathbb{B} into groups that correspond to different features. This is done by partitioning the set \mathbb{B} into sets of consecutive bases, B_1, B_2, \dots, B_n . For example, suppose $\mathbb{B} = \{9, 10, 13, 14, 16\}$, then we split into 3 groups $B_1 = \{9, 10\}$, $B_2 = \{13, 14\}$, and $B_3 = \{16\}$.

Step 5: Determine the features. We now look at each of the groups B_1, B_2, \dots, B_n individually. Consider the group B_i . If the group only contains a single base it may be worth zooming in by considering a finer discretisation in this region. For example suppose $B_i = \{13\}$, this consists of the region $[T/4, 3T/8]$. To zoom in consider the group $\{25, 26\}$ which correspond to the regions $[T/4, 5T/16]$ and $[5T/16, 3T/8]$ respectively. Suppose B_i contains multiple bases. We want to distinguish between increases, decreases and bumps in the group. Furthermore, each group could consist of multiple features that are close together. However, recall that the sign of coefficients corresponds to either increasing or decreasing intervals. Thus we can partition B_i into k either increasing and decreasing sections $B_i = B_{i,1} \cup B_{i,2} \cup \dots \cup B_{i,k}$. We order the sets $B_{i,j}$ such that $B_{i,1}$ contains the smallest indices and $B_{i,k}$ the largest. For example suppose $B_i = \{10, 11, 12, 13, 14\}$ with corresponding coefficients $\{-0.7, -0.2, 0.3, 0.5, 0.8\}$. The resultant partition is then $\{10, 11\} \cup \{12, 13, 14\}$.

We now examine each $B_{i,j}$ to calculate the features in B_i .

- If the corresponding coefficients of $B_{i,1}$ are positive then this group is a decreasing interval.
- If the corresponding coefficients of $B_{i,k}$ are negative then this group is an

increasing interval.

- For n in $\{1, 2, \dots, k - 1\}$ if the coefficients corresponding to group $B_{i,n}$ are negative and $B_{i,n+1}$ positive, join the groups together $B_{i,n*} = B_{i,n} \cup B_{i,n+1}$ and a bump occurs in this region.

To calculate the position of the features we utilise the indices of the Haar bases. Suppose a feature occurs in the group $B_{i,j}$, where a and b are the minimum and maximum indices in $B_{i,j}$ respectively. Then the interval of the feature is $[(a - p - 1)T/p, (b - p)T/p]$.

We now give a worked example of extracting features from three functions shown by the black lines in Figure 6.7, where each function is decomposed into $q = 16$ Haar bases. The functions are drawn from GPs on the interval $[0, 20]$ with length scales 2, 4 and 10 respectively. Our first step is to consider the finest bases, for $q = 16$ this corresponds to the last 8 bases. The coefficients of the last 8 bases are shown in Table 6.2.

Function	α_9	α_{10}	α_{11}	α_{12}	α_{13}	α_{14}	α_{15}	α_{16}
1	-0.19	-0.30	0.62	0.71	0.02	0.06	-2.60	1.09
2	-0.74	-0.19	0.75	0.90	0.23	-0.23	-0.17	-0.29
3	-0.33	-0.35	-0.30	-0.19	-0.05	0.11	0.24	0.33

Table 6.2: The coefficients of bases h_9, \dots, h_{16} for the three functions shown in Figure 6.7.

Next we need to threshold the coefficients to remove bases that correspond to almost flat regions. Here we choose to threshold on $\sigma_{\text{thres}} = \Delta/10q$, where Δ is the range of the function and $q = 16$. Hence, the coefficients are thresholded at 0.18, 0.13 and 0.08 for functions 1, 2 and 3 respectively. Thresholding determines the important bases as $\mathbb{B}_1 = \{9, 10, 11, 12, 15, 16\}$ for the first function, $\mathbb{B}_2 = \{9, \dots, 16\}$ for the second function and $\mathbb{B}_3 = \{9, 10, 11, 12, 14, 15, 16\}$ for the third function. The important bases can be visualised as the highlighted cells in step 2 of Figure 6.8.

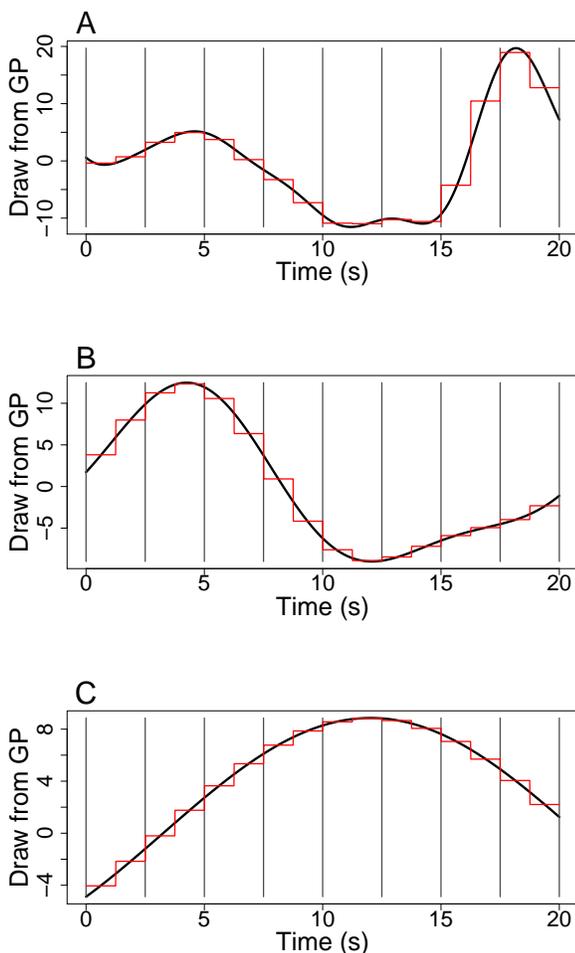


Figure 6.7: Functions (black lines) drawn from a GP on the interval $[0s, 20s]$ with step size equal to $0.01s$ and length scale equal 2 (A), 4 (B) and 10 (C). The red piecewise constant functions show the approximation given by 16 Haar bases. The intervals of the finest 8 Haar bases are visualised by the regions separated by the grey lines.

We need to check whether the removed indices correspond to the peak of a bump. For the first function the indices of the removed bases are 13 and 14. To see if index 13 corresponds to a bump we require $\alpha_{14} > \sigma_{\text{thres}}$ and $\alpha_{12} < -\sigma_{\text{thres}}$. However, $\alpha_{14} = 0.06 < 0.18 = \sigma_{\text{thres}}$ so the index does not correspond to a peak. Similarly, when we check if index 14 corresponds to a peak we find $\alpha_{13} = 0.02 < 0.18$. Hence index 14 is not added to \mathbb{B}_1 . Notice that \mathbb{B}_2 contains all the indices so we do not need

Step 2:	Function	α_9	α_{10}	α_{11}	α_{12}	α_{13}	α_{14}	α_{15}	α_{16}
	1	-0.19	-0.30	0.62	0.71	0.02	0.06	-2.60	1.09
	2	-0.74	-0.19	0.75	0.90	0.23	-0.23	-0.17	-0.29
	3	-0.33	-0.35	-0.30	-0.19	-0.05	0.11	0.24	0.33

Step 3:	Function	α_9	α_{10}	α_{11}	α_{12}	α_{13}	α_{14}	α_{15}	α_{16}
	1	-0.19	-0.30	0.62	0.71	0.02	0.06	-2.60	1.09
	2	-0.74	-0.19	0.75	0.90	0.23	-0.23	-0.17	-0.29
	3	-0.33	-0.35	-0.30	-0.19	-0.05	0.11	0.24	0.33

Step 4:	Function	α_9	α_{10}	α_{11}	α_{12}	α_{13}	α_{14}	α_{15}	α_{16}
	1	-0.19	-0.30	0.62	0.71	0.02	0.06	-2.60	1.09
	2	-0.74	-0.19	0.75	0.90	0.23	-0.23	-0.17	-0.29
	3	-0.33	-0.35	-0.30	-0.19	-0.05	0.11	0.24	0.33

Step 5:	Function	α_9	α_{10}	α_{11}	α_{12}	α_{13}	α_{14}	α_{15}	α_{16}
	1	-0.19	-0.30	0.62	0.71	0.02	0.06	-2.60	1.09
	2	-0.74	-0.19	0.75	0.90	0.23	-0.23	-0.17	-0.29
	3	-0.33	-0.35	-0.30	-0.19	-0.05	0.11	0.24	0.33

Figure 6.8: Visualisation of extracting features for the functions in Figure 6.7. In step 2 we highlight the bases whose coefficient is larger than the threshold — 0.18, 0.13 and 0.08 for functions 1, 2 and 3 respectively. In step 3 the orange box symbolised the base added into the important bases, when checking for a peak. In step 4 we split the bases into consecutive indices, hence splitting the bases for function 1 into two groups, shown in light and dark blue. In step 5 each group from step 4 is partitioned into negative (red) and positive values (yellow). The boxes surrounding the bases shows the different features found by looking at the pattern of sign changes. Function 1 has two features; a bump in bases $\{9, 10, 11, 12\}$ and another bump in bases $\{15, 16\}$. Function 2 also has two features; a bump in bases $\{9, 10, 11, 12, 13\}$ and an increasing region in bases $\{14, 15, 16\}$. Function 3 has one feature; a bump in the bases $\{9, \dots, 16\}$.

to check for missing peaks. For the third function index 13 does not belong to \mathbb{B}_3 , so we need to check if it corresponds to a peak. We find that $\alpha_{12} = -0.19 < -0.08$ and $\alpha_{14} = 0.11 > 0.08$, hence the base does correspond to a peak and we add the base into \mathbb{B}_3 .

We then split the index sets into their consecutive parts. For function 1 we split into $\{9, 10, 11, 12\}$ and $\{15, 16\}$, whereas for functions 2 and 3 we obtain $\{9, \dots, 16\}$.

Now we are ready to determine the features of each function. This involves splitting each set of indices depending on the sign of the corresponding coefficients. Then depending on the pattern of positive and negative coefficients we assign features. This can be visualised by step 5 of Figure 6.8. For each function the partition on sign is shown by the red/yellow cells. The features for each function is shown by black boxes surrounding the coefficients. Below we explain in detail how the features are obtained for each function.

For function 1 we have two groups. Let us first consider $\{9, 10, 11, 12\}$. Partitioning on the sign of corresponding coefficients gives $B_{1,1} = \{9, 10\}$ and $B_{1,2} = \{11, 12\}$, with negative and positive coefficients, respectively. This sign pattern corresponds to a bump across both $B_{1,1}$ and $B_{1,2}$. We join the groups together to form $B_{1,1*} = \{9, 10, 11, 12\}$. The bump occurs in the interval $[20(9 - 9)/8, 20(12 - 8)/8] = [0, 10]$. We consider next the other group $B_2 = \{15, 16\}$. We partition depending on the sign of corresponding coefficients giving $B_{2,1} = \{15\}$ and $B_{2,2} = \{16\}$. Since $\alpha_{15} = -2.6 < 0$ and $\alpha_{16} = 1.09 > 0$ we join the groups together $B_{2,1*} = \{15, 16\}$, giving a bump in the region $[20(15 - 9)/8, 20(16 - 8)/8] = [15, 20]$.

Consider the second function. We partition \mathbb{B}_2 according to the sign of corresponding coefficients. This gives $B_{1,1} = \{9, 10\}$, $B_{1,2} = \{11, 12, 13\}$ and $B_{1,3} = \{14, 15, 16\}$, where $B_{1,1}$ and $B_{1,3}$ corresponds to negative coefficients and $B_{1,2}$ positive coefficients. Thus applying step 5 we find that $B_{1,3}$ leads to an increasing region in $[12.5, 20]$. Additionally, we join the bases $B_{1,1}$ and $B_{1,2}$ together to get $B_{1,1*} = \{9, 10, 11, 12, 13\}$, which describes a bump in the interval $[0, 12.5]$.

Finally, following step 5 for the third function gives $B_{1,1} = \{9, 10, 11, 12, 13\}$ and $B_{1,2} = \{14, 15, 16\}$. The indices in $B_{1,1}$ and $B_{1,2}$ correspond to negative and positive

coefficients, respectively. Thus, we join the two sets together and we find a single bump in the region $[0, 20]$.

In summary, we have found that the first function has two bumps; one in the region $[0, 10]$ and the other in the region $[15, 20]$. For the second function we obtained one bump in the region $[0, 12.5]$ and the function is increasing in $[12.5, 20]$. Finally for the third function we found a single bump in the region $[0, 20]$. Looking at the functions in Figure 6.7 we see that these features accurately describe the features in the functions.

6.2.1 Calculating the magnitude of the features

We have shown above how to extract features and their positions from a function. However, when clustering functions we also want to judge them on the size of their features. For example suppose we are given 10 functions. We extract the features from these functions and find that they all have a bump in the region $[0, 5]$. The first six functions have a bump with range smaller than 3 and the remaining four have range larger than 5. In this case we may want to split the functions into two groups; one with first six functions and the other with the remaining four. Hence, to each feature we attach a value that signifies the magnitude of the feature.

Consider again the function f , which has k features $\xi_1, \xi_2, \dots, \xi_k$. Each feature ξ_i consists of the feature type — bump, increasing or decreasing — and the interval $[a_i, b_i] \subset [0, T]$ to which it belongs. To each feature ξ_i we adjoin a magnitude m_i that is defined as the range of f in the interval $[a_i, b_i]$, namely $m_i = \max(f|_{[a_i, b_i]}) - \min(f|_{[a_i, b_i]})$.

We apply the above to find the magnitudes of the features for the functions in Figure 6.7. For the first function, the magnitude of the bump between $[0, 10]$ is 14.6 and for the bump between $[15, 20]$ is 29.1. Thus the magnitude of the first bump is 14.5 smaller than the second. Looking at Figure 6.7(A) we see that the first bump is indeed smaller by a factor of two. The second function has two features, the bump in $[0, 12.5]$ which has magnitude 21.5 and the increasing interval which has magnitude 7.8. By considering Figure 6.7(B), we find that the bump is larger than the increasing

region. Finally, the third function has a single bump in region $[0, 20]$ which has magnitude 13.7. Hence, the magnitudes accurately distinguish between large and small features.

The method implemented above does not directly use the Haar basis but rather the values the function of interest takes. It is possible to construct a method that calculates the magnitude by using the Haar basis. Namely, for each feature combine the contribution from all bases that are defined on that interval. For example, suppose a feature occurs in the interval $[0, 5]$. If we approximated the function with 16 Haar bases, the bases that take a non-zero value on this interval are: h_1, h_2, h_3, h_5, h_9 and h_{10} . We then calculate the approximated value of the function on each of the finest grid in the interval. The approximated value for the interval $[0, 1.25]$ is $\alpha_1 + \alpha_2 + 2^{0.5}\alpha_3 + 2\alpha_5 + 2^{1.5}\alpha_9$. We have no contribution from h_{10} as it is defined on $[2.5, 5]$. Furthermore, on other intervals we will have a negative contribution from some bases, i.e in the interval $[1.25, 2.5]$ the contribution from h_9 is negative. However, this method has two main limitations. Firstly, we only have a few values to evaluate the magnitude. In the above example we would only have 4 values - corresponding to the intervals $[0, 1.25]$, $[1.25, 2.5]$, $[2.5, 3.75]$ and $[3.75, 5]$ - to decide the magnitude. Secondly, since the Haar approximation only gives a single value for the intervals, if the function varies significantly this value may not accurately represent the magnitude. Hence, we decided to evaluate the magnitude using the values of the function rather than the Haar approximation, as it does not suffer from these pitfalls.

In our method we decided to take a single value to describe the size of features. However, an alternative is to consider how the function changes on smaller intervals. For example consider the function in Figure 6.7(A) which has a bump in $[15, 20]$ with magnitude 29.1. Rather than a single value for the whole region, we now assign 26.4 and -12.4 to the smaller regions $[15, 17.5]$ and $[17.5, 20]$. The values describe the change of the function in these intervals. The advantage of this new approach is that it gives extra details in the shape and size of the feature. Furthermore, this breakdown informs on the rate of change of intensity for the bump or whether the

bump begins and ends at similar intensity values. In the above example, we see that the functions increase more in the region $[15, 17.5]$ than it decreases in $[17.5, 20]$. Also, the intensity is higher at the end of the bump than the beginning. However, with the addition details comes more complex descriptions. If every feature was described more accurately it would be difficult to decide how to cluster the functions on the magnitude. Hence, we decided to use the simpler method to allow for easier clustering at the cost of less detailed features.

6.3 Clustering

We now consider how to cluster a set of functions depending on the features they contain. As mentioned at the beginning of this chapter clustering is not an easy challenge and the type of clusters can vary dramatically depending on what we are interested in. For example, we may only be interested in features that occur in the same time intervals whereas another may be interested in the total number of features. Consequently, we present a representation of the data which can be easily clustered by the user, rather than create a method that clusters the functions without input. This flexibility allows for a greater variety of clusters to be considered.

For our approach we require a method to easily compare the features of multiple functions. We do this by representing the features in terms of a vector over the intervals provided by the Haar basis. Begin by considering a single function f on the interval $[0, T]$. We express f in terms of q Haar bases and find its features. With q bases considered, our method partitions $[0, T]$ into p equal intervals, $[0, T/p], \dots, [(p-1)T/p, T]$, with p as defined above. By construction the region that a feature is defined in comes from these intervals. Thus, we can express the features of f in a vector $\mathbf{f}_{\text{feat}} = (f_{\text{feat},1}, \dots, f_{\text{feat},p})$ of length p , where $f_{\text{feat},i}$ corresponds to the feature in the interval $[(i-1)T/p, iT/p]$. For example, suppose f defined on $[0, 10]$ is decomposed into its features via 8 Haar bases. With 8 bases we split the interval into 4 regions $[0, 2.5], [2.5, 5], [5, 7.5]$ and $[7.5, 10]$. We find that f has a single feature; an increasing interval in $[0, 5]$. Expressing as a vector we obtain

$\mathbf{f}_{\text{feat}} = (\text{increasing}, \text{increasing}, \text{NA}, \text{NA})$. Notice that intervals with no features are expressed as NA. To distinguish between different bumps they will be labelled in the order of appearance, i.e **bump 1**, **bump 2**, etc. This is not required for increasing or decreasing regions because by definition two increases/decreases cannot be connected. To clarify this, suppose we have two increasing regions $[A, B]$ and $[B, C]$ that link to the indices $\{a, b, c\}$ and $\{d, e\}$ respectively. To be increasing the corresponding coefficients must be negative for both groups. Hence, when generating features we would only create a single group $\{a, b, c, d, e\}$ when splitting for positive and negative coefficients. Thus, the method would output a single increasing region in $[A, C]$ rather than two connected intervals. The magnitude of features can be stored in a similar vector $\mathbf{f}_{\text{mag}} = (f_{\text{mag},1}, \dots, f_{\text{mag},p})$, of length p , where $f_{\text{mag},i}$ corresponds to the magnitude of the feature in the interval $[(i-1)T/p, iT/p]$.

For example, consider once more the function given in Figure 6.7(A). The function is defined on the interval $[0, 20]$, and we used 16 Haar bases. We found the function had two bumps, one in the interval $[0, 10]$ with magnitude 14.6 and the other in the interval $[15, 20]$ with magnitude 29.1. With $q = 16$ we partition $[0, 20]$ into 8 equal regions $[0, 2.5], \dots, [17.5, 20]$. Thus, we can express the features and corresponding magnitudes on these intervals, as shown in Table 6.3. Rather than displaying NA if there is no feature we now keep the position blank.

Interval	$[0, 2.5]$	$[2.5, 5]$	$[5, 7.5]$	$[7.5, 10]$
\mathbf{f}_{feat}	bump 1	bump 1	bump 1	bump 1
\mathbf{f}_{mag}	14.61	14.61	14.61	14.61
Interval	$[10, 12.5]$	$[12.5, 15]$	$[15, 17.5]$	$[17.5, 20]$
\mathbf{f}_{feat}			bump 2	bump 2
\mathbf{f}_{mag}			29.07	29.07

Table 6.3: Example of vector representation for the features and magnitudes of the function shown in Figure 6.7(A).

Now consider a dataset with N functions defined on the interval $[0, T]$. For each

function f_j with $j \in \{1, \dots, N\}$ we approximate with q Haar bases and calculate its features and their magnitudes. We use the vector representation of the features $\mathbf{f}_{\text{feat}}^j$ and magnitudes $\mathbf{f}_{\text{mag}}^j$. We create a matrix \mathcal{F} to store the features of all the functions. The matrix has p columns and N rows, where the j th row is taken to be $\mathbf{f}_{\text{feat}}^j$. This matrix gives us a way to easily compare features of multiple functions via visual inspection. To improve the visualisation further we colour the matrix such that each feature has a unique colour and cells without a feature are coloured white. Similarly, we create the matrix \mathcal{M} to contain the magnitudes of the features. The j th row of matrix \mathcal{M} is the vector $\mathbf{f}_{\text{mag}}^j$. The elements of this matrix are coloured on a scale where larger values are yellow and smaller values red.

The advantage of representing the features in this manner is the ease at which one can compare the functions, which is further enhanced by colouring the matrices.

To illustrate the method we generate 30 intensity functions on the interval $[0, 20]$, shown in Figure 6.9(A). Each function has a base intensity drawn uniformly between 1 and 5. Functions 1-10 (black lines) have a bump added in the region $[0, 10]$. Functions 11-20 (red lines) have a bump added in the region $[10, 20]$. And functions 21-30 (blue lines) have a bump added in both regions. The width of the bumps are fixed at 3.1 and 1.9 in the regions $[0, 10]$ and $[10, 20]$, respectively. The height of each bump is drawn uniformly between 0.5 and 5. The onset of the bumps in $[0, 10]$ occur uniformly in the interval $[1, 5.4]$ and the onset on the bumps in $[10, 20]$ occur uniformly in $[11, 15.4]$.

We then apply our methods, with $q = 32$, to find the features of each function and their magnitude. In Figure 6.9(B) we show the matrix of features. As q was taken to be 32, we split $[0, 20]$ into 16 equal regions $[0, 1.25]$, $[1.25, 5]$ to $[18.75, 20]$ —which can be viewed as the number of columns of Figure 6.9(B). Furthermore, each row of Figure 6.9(B) corresponds to a function. The top row corresponds to the first function in the dataset and the bottom row the last function in the dataset. We see the matrix contains 3 colours; white for intervals with no feature, green for the first bump and orange for the second bump. A key is provided on the right hand side. Figure 6.9(C) shows the corresponding magnitudes. Notice that the matrix has

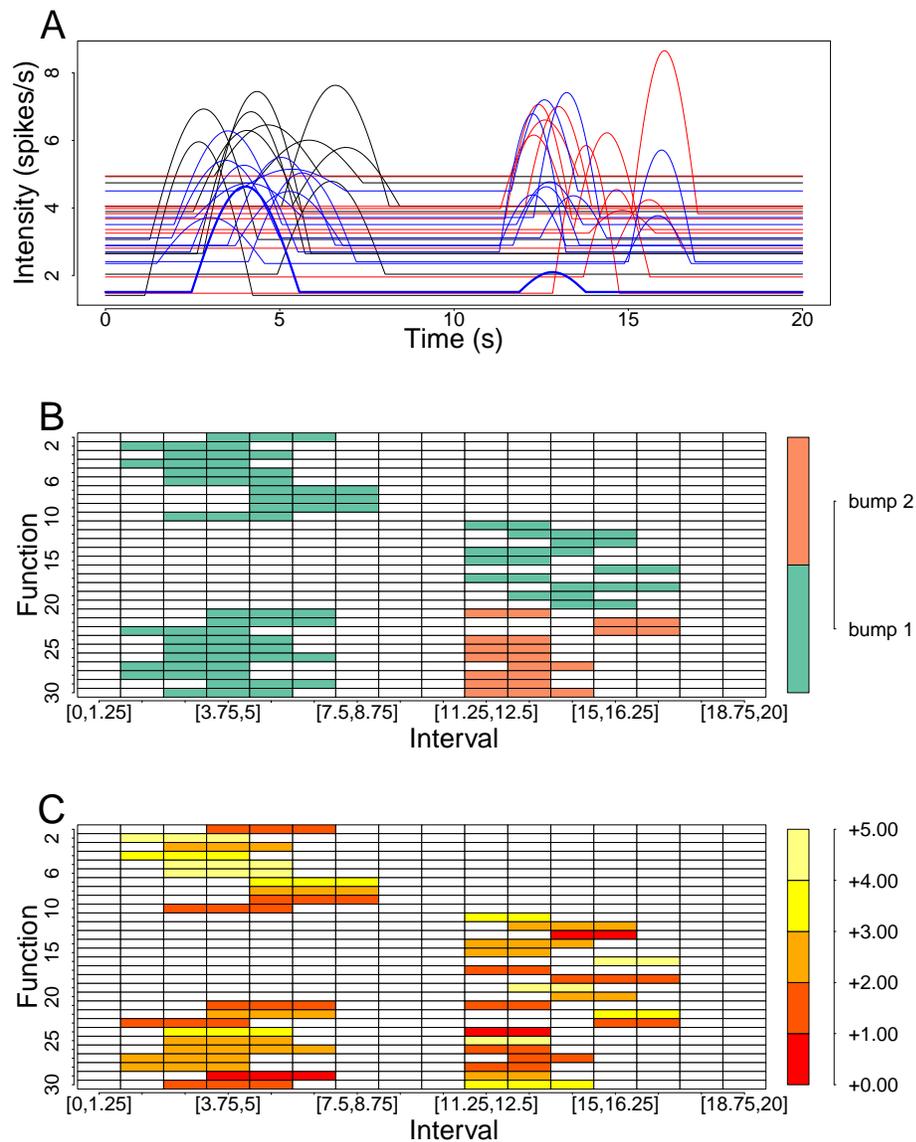


Figure 6.9: (A) shows the functions to be clustered, where the first 10 functions (black lines) have a single bump in $[0, 10]$, functions 11-20 (red lines) have a single bump in $[10, 20]$, and functions 21-30 (blue lines) have a bump in each region. (B) shows the resultant features of each function and (C) the magnitude of the features.

the same dimension as Figure 6.9(B), and the same matrix elements are coloured. However, the colours have changed to show the magnitude. The colours form a scale from red to yellow, where red corresponds to a small magnitude (in this case between

0 and 1) and yellow a large magnitude (between 4 and 5).

For example, consider the 24th function (thick blue line in Figure 6.9(A)). By inspecting the 24th row in Figure 6.9(B) we find two bumps: the first in the interval $[2.5, 6.25]$ (green cells) and the second in the interval $[11.25, 13.75]$ (orange cells). Consider the same row in Figure 6.9(C). We see that the interval for $[2.5, 6.25]$ is coloured bright yellow, thus this feature has magnitude in the region $[3, 4]$. The interval for $[11.25, 13.75]$ is coloured deep red, hence the magnitude is in the region $[0, 1]$. By using the two matrices together we can distinguish the type and size of features.

We now consider clustering the functions. Depending on the criteria used we can find different clusters. First consider only the features of the functions — shown in Figure 6.9(B). We find that functions 1 to 20 have a single bump. The bump lies somewhere in the region $[1.25, 8.75]$ for functions 1 to 10 and $[11.25, 17.5]$ for functions 11 to 20. On the other hand, functions 21 to 30 have two bumps, one in each of the regions above. One option is to cluster on the position and number of bumps. This gives three clusters $\{1, \dots, 10\}$, $\{11, \dots, 20\}$ and $\{21, \dots, 30\}$. If we were only interested in the number of features we could split the functions into those with one bump and those with two. This gives two clusters $\{1, \dots, 20\}$ and $\{21, \dots, 30\}$. If we were only interested in functions that have bumps in a precise location we could also cluster on this, for example functions $\{16, 22, 23\}$ all have a bump in the interval $[15, 17.5]$.

We can also take into consideration the magnitudes of these features from Figure 6.9(C). Suppose we have grouped the functions via their features into three clusters $\{1, \dots, 10\}$, $\{11, \dots, 20\}$ and $\{21, \dots, 30\}$. Notice that all the magnitude of features are contained in the ranges $[0, 1], \dots, [4, 5]$. Hence we can split these clusters depending on the size of their features. For the functions in $\{1, 10\}$, we find that the magnitude of $\{1, 3, 8, 9, 10\}$ lies in $[0, 3]$ and for $\{2, 4, 5, 6, 7\}$ the magnitude lies in $[3, 5]$. Applying this to each of the clusters gives us a new partition, as shown in Table 6.4. For the functions in $\{21, \dots, 30\}$ we took the largest magnitude of both bumps. The clusters can be seen in Figure 6.10.

Cluster	Description	
	Magnitude	Feature
{1, 3, 8, 9, 10}	in [0, 3]	Single bump within [1.25, 8.75]
{2, 4, 5, 6, 7}	in [3, 5]	Single bump within [1.25, 8.75]
{11, 16, 19}	in [3, 5]	Single bump within [11.25, 17.5]
{12, 13, 14, 15, 17, 18, 20}	in [0, 3]	Single bump within [11.25, 17.5]
{22, 24, 25, 30}	in [3, 5]	Two bumps within [1.25, 8.75] and [11.25, 17.5]
{21, 23, 26, 27, 28, 29}	in [0, 3]	Two bumps within [1.25, 8.75] and [11.25, 17.5]

Table 6.4: Six clusters of the example dataset, clustered on feature position and magnitude.

We see by analysing the simulated dataset the flexibility that this method offers. The matrices of features and magnitudes summarises the shape of each function and allows for quick comparison between functions. Furthermore, depending on our interest (e.g. number of bumps, features in $[0, 10]$, etc) the matrices allows us to choose the way in which we cluster the functions.

6.4 Application to HEK293 cells

In this section we apply our clustering method to HEK293 cells stimulated with carbachol. Before clustering takes place, we need to estimate the intensity functions from the Ca^{2+} data. For each cell that presents a sequence of Ca^{2+} oscillations we convert this to a sequence of Ca^{2+} spike times. The Ca^{2+} spike times are used to infer the parameters of our model — the intensity function and ISI parameter. See Chapters 2 and 3 for details of how the model parameters are inferred. Recall, the posterior intensity function is not a single function but a probability density function over the space of functions. Thus, for clustering we will use the mean of the posterior

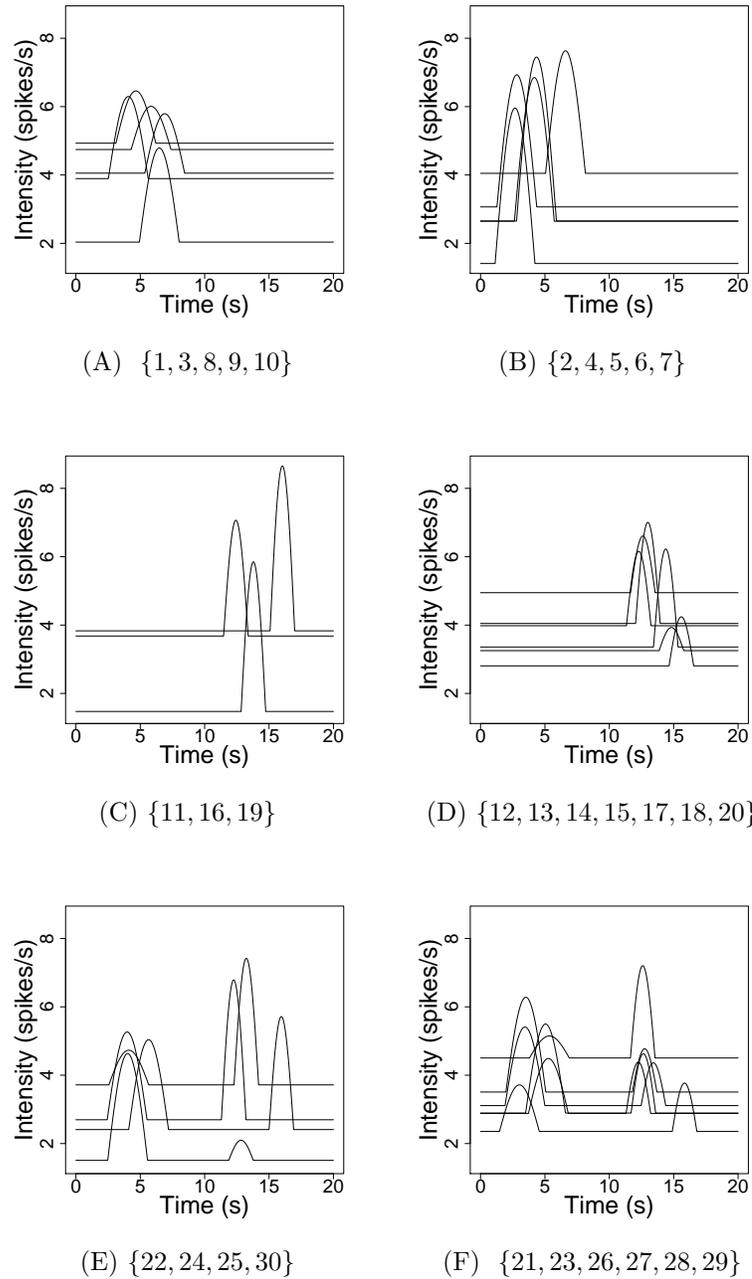


Figure 6.10: Clusters of the simulated dataset, where clustering depends on position and magnitudes of features explained in Table 6.4.

intensity function distribution.

This section is split into 3 parts. The first looks at how the method used to obtain Ca^{2+} spike sequences can change the features of the inferred intensity function. The second clusters intensity functions from constant stimulus experiments and, the third clusters intensity functions from step change experiments.

6.4.1 Effects of altering Ca^{2+} spike times on features

As discussed in the introduction, the Ca^{2+} concentration of a cell is recorded experimentally by using a fluorescent dye. These dyes can act as a Ca^{2+} buffer, which over time reduces the spiking rate of the cell. Thus, it may seem sensible to adjust the Ca^{2+} spike times to counter these effects. Indeed, Thurley et al. [6] remove linear trends (if they exist) in the ISIs when modelling Ca^{2+} oscillations as stationary processes. With this in mind, it is not unreasonable to adjust the Ca^{2+} spike times if we believed they did not mirror the true dynamics in the cell. However, it is important to note that any change to the Ca^{2+} spike times may change the output of our model. In particular, the features of the posterior intensity function could change their locations or be removed, which could alter the clustering.

To highlight this, we consider two HEK293 cells that are stimulated with $50\mu\text{M}$ carbachol over 3240s. The Fura-2 fluorescence intensity traces of the two cells are shown in Figure 6.11. From the traces we extract the Ca^{2+} spike sequences. In both these cases the corresponding ISIs show a positive trend. This is to say that the further into the experiment the longer the time between Ca^{2+} spikes. Thus, we generate new Ca^{2+} spike times where the linear trends are removed. This is done by first fitting a linear model to the ISI times against ISI number. Then we scale the ISIs with respect to the linear fit whilst preserving the mean. See Figure 6.12 to visualise the transformation, where the black dots and line correspond to the original ISI times and their trend. The red dots are the new ISI times with the linear trend removed. Hence, for each cell we have two sets of Ca^{2+} spikes times corresponding to with and without removing trends.

We then fit the inhomogeneous Gamma ISI model to these Ca^{2+} spike sequences

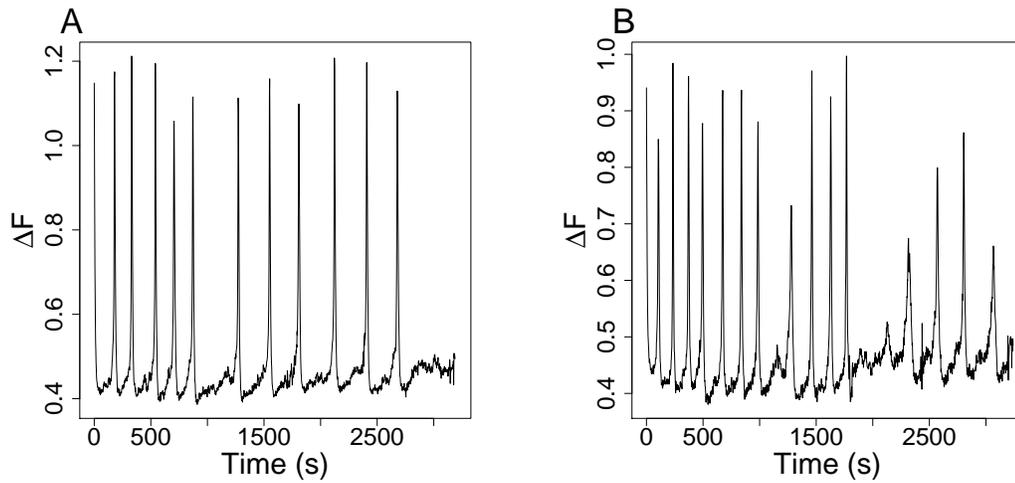


Figure 6.11: Fura-2 fluorescence intensity traces of two HEK293 cells stimulated with $50\mu\text{M}$ carbachol. Changes in the fluorescence of the Ca^{2+} indicator is recorded relative to its basal level (ΔF).

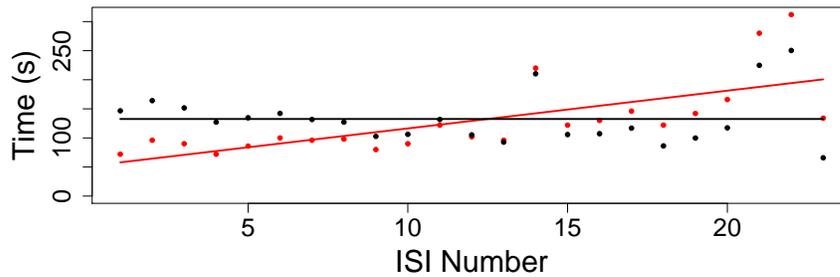


Figure 6.12: Example of removing the trend from ISIs, where the ISIs with trends are shown by black dots and the ISIs with linear trends removed are red. The solid lines show the linear trends in the respective ISIs.

where the intensity function has a PWC prior. In Figure 6.13 we see how the posterior intensity function distribution changes depending on whether we remove linear trends. The red lines and regions represent the mean and 95% credible region of the posterior

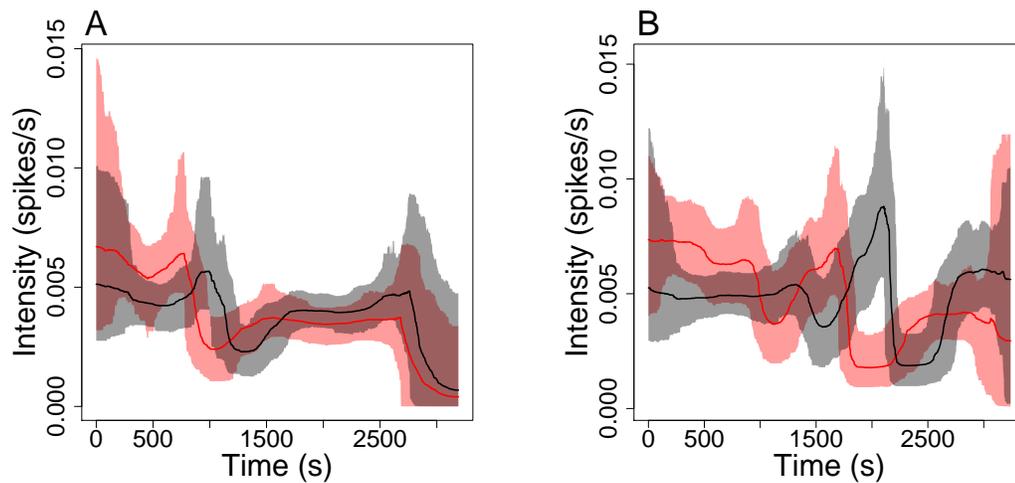


Figure 6.13: Illustration of the posterior intensity distribution when the model is fitted to an individual Ca^{2+} spike sequence with (black) and without (red) linear trends in the spike times removed. This is shown for two cells in (A) and (B), respectively. The solid lines and shaded regions correspond to the mean of the posterior intensity function distribution and 95% credible region.

intensity function distribution fitted with the original Ca^{2+} spike sequence, whereas the black lines and regions corresponds to the transformed Ca^{2+} spike sequence with linear trends removed. Recall from above that both original ISIs contained a positive trend. A positive trend in the time between Ca^{2+} spikes corresponds to a negative trend in the spiking rate. This can be seen in both Figure 6.13(A) and (B) where the red lines tend to decrease over time. In other words, if we fitted a line of best fit through the mean of the posterior intensity function distribution it would have a negative gradient, whereas this trend cannot be found in the intensity distribution fitted with the Ca^{2+} spike sequence with linear trends removed (black lines). Thus, the posterior intensity function distribution can detect whether we removed the linear trends in the ISIs or not.

We notice that the functions maintain the majority of their overall shape. In Figure 6.13(A) both the red and black lines begin by decreasing, then a bump before

plateauing and decreasing at the end. However, the intervals over which these events occur are different. We see that the bump for the red line occurs in the interval [500s, 1500s] whereas for the black line it occurs in [750s, 1750s]. Thus, removing the linear trends in the ISIs has affected the position of features in the intensity function. When we shift our focus onto Figure 6.13(B), we see again that the functions maintain most of their shape, but not all. In the initial 1000s of the function we see the red line is decreasing but the black remains mostly constant. Thus, removing trends may also remove increasing or decreasing regions found in the intensity function.

We apply our feature extracting method with 32 Haar bases to the inferred intensities in Figure 6.13(A). The features are described in 16 intervals [0s, 202s], [202s, 405s], . . . , [3038s, 3240s]. The features are shown in the table below, where the intervals are labelled 1 to 16, and decreasing and increasing regions are represented as `dec` and `inc` respectively.

Interval	1	2	3	4	5	6	7	8
Raw		<code>dec</code>	<code>bump 1</code>	<code>bump 1</code>	<code>bump 1</code>	<code>inc</code>	<code>inc</code>	<code>inc</code>
No trend		<code>dec</code>		<code>bump 1</code>	<code>bump 1</code>	<code>bump 1</code>		<code>inc</code>
Interval	9	10	11	12	13	14	15	16
Raw						<code>dec</code>	<code>dec</code>	
No trend	<code>inc</code>						<code>dec</code>	<code>dec</code>

Table 6.5: Features of intensity functions generated from the same cell, shown in Figure 6.13. Raw and no trend corresponds to the intensity function when the model was fitted from the original Ca^{2+} spike sequence and the spike sequence with linear trends removed, respectively.

We see that that the same types of features occur in both functions but the times where they occur has changed. Therefore, we have shown how altering — in particular removing linear trends — Ca^{2+} spike times can affect the features of an inferred intensity function. From this point onwards we do not alter the Ca^{2+} spike times prior to fitting the intensity function.

6.4.2 Constant stimulus experiments

We will apply our clustering method to 60 HEK293 cells challenged with $10\mu\text{M}$ carbachol for 3002s. For each Ca^{2+} spike sequence we fit the inhomogeneous Gamma ISI distribution with a PWC prior for the intensity function. The means of the posterior intensity function distributions are shown in Figure 6.14. We see that the intensity functions are generally between 0 and 0.03 Ca^{2+} spikes per second over the whole experiment, although there does appear to be an initial peak in intensity in the region $[0\text{s}, 300\text{s}]$. Furthermore, it appears that most functions remain flat in large regions although, some functions vary in intensity after the initial peak.

We now decompose each of these functions into their features, using $q = 32$ Haar bases. The functions are partitioned into 16 intervals where features can occur. As

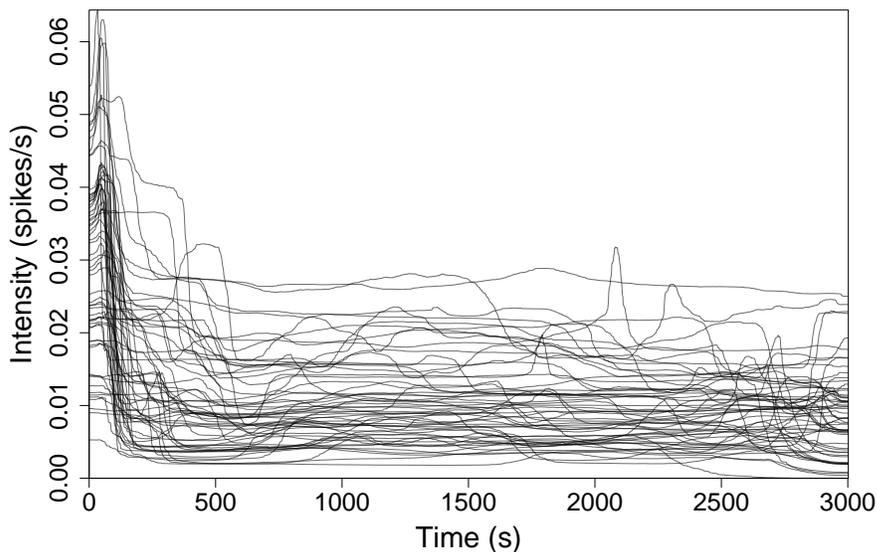


Figure 6.14: The mean of the posterior intensity function distribution obtained from fitting individual Ca^{2+} spike sequences to the inhomogeneous Gamma ISI distribution. The Ca^{2+} spike sequences were obtained from HEK293 cells challenged with constant stimulus.

the intensity for each functions is small we multiple each by 100 to emphasise the difference between flat regions and features in the Haar bases. The threshold used for most functions was $\Delta/10q$. The exceptions are functions 2, 9, 28, 41, 56 where we thresholded at 0.002, 0.008, 0.01, 0.002, 0.003, 0.005, respectively. For these functions the threshold $\Delta/10q$ did not accurately find all features. Either the threshold was too small (created features that did not exist) or too large (missed features). The resultant matrices are shown in Figures 6.15 and 6.16 for the features and magnitude, respectively.

Consider first only the features. We can see via the key that the functions contain increasing and decreasing regions (`inc` and `dec`) and up to three individual bumps. In fact, only function 31 contains three bumps. The majority of functions either have no bumps (22 functions), or a single bump (38 functions), with only eight functions having two bumps. Observing the first two columns of the matrix we see that the majority of functions have a feature here. Only function 42 does not have a feature occurring in these two intervals. All of these features are either decreasing regions or bumps, with the majority (54/59) containing decreasing intervals. This corresponds to the initial transience in spiking rate, where once stimulated the spiking rate begins high but quickly degrades. We see both bumps and decreasing regions because in some cases the spiking rate initially increases before a large drop in intensity. After the initial decrease we see that some cells recover and an increase in spike rate occurs — the lime green boxes in the third to eighth intervals. This does not occur in all functions and is a property we could cluster on. It is rare to find a decrease in intensity in this region. Consider the last eight columns. We see varied behaviour, with both bumps and decreasing and increasing regions. However, we see these columns contain far more decreasing intervals then columns 3-8. Concentrating on the bumps, we see their positions do vary over the entire function. The bumps length varies from only one column to nine, with the majority covering 2-4 columns. Thus, we have found that initially the features cells present is common throughout most cells, then as time progresses the type and timing of features is harder to predict. This can be seen by looking at the first 3 columns, where the majority of cells contain a decrease (purple).



Figure 6.15: Matrix containing the extracted features for the constant stimulus experiments.

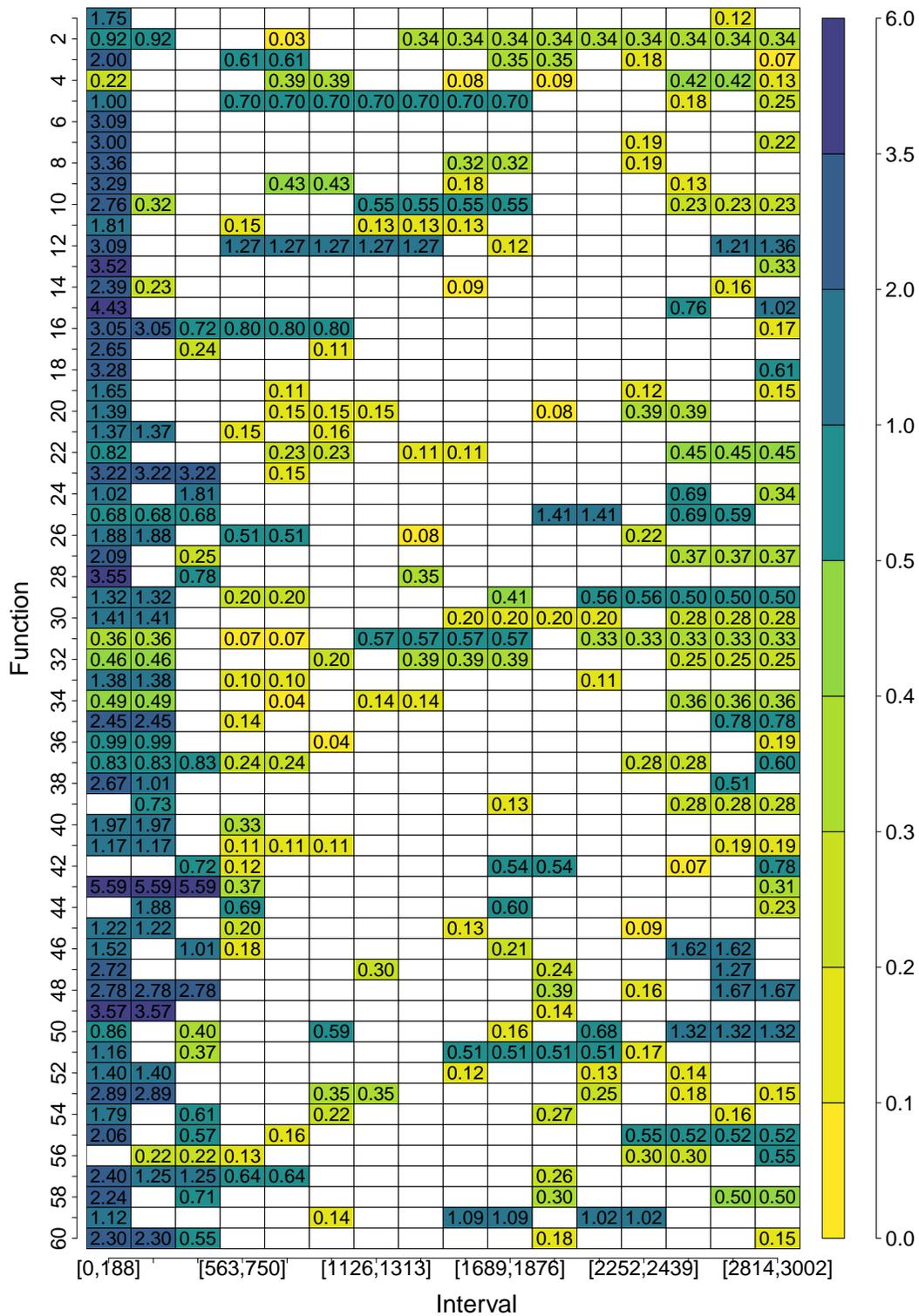


Figure 6.16: Matrix containing the magnitudes of the features for the constant stimulus experiments.

Then the following columns contain an increase (lime). After this the features vary, and we see a mixture of all colours in columns 7-16. This could imply temporal heterogeneity.

Consider the corresponding magnitudes in Figure 6.16. We decided the scale used to colour the magnitudes by considering the histogram of all magnitudes, shown in Figure 6.17. We see that the majority of features have magnitude in $[0, 0.5]$ (red bar), and there are less features with larger magnitudes. Hence, we want to break this region into smaller sections. The scale we used ranges from yellow to lime green for magnitudes in $[0, 0.5]$. Then for magnitudes in $[0.5, 1]$, $[1, 2]$, $[2, 3.5]$ and $[3.5, 6]$ we colour from a dark green/blue to purple. This is shown in the key of Figure 6.16. We find that the large magnitudes are dominated at the start of the function corresponding to the initial decrease. The amount a function decreases does vary considerably between functions — from 5.59 to 0.36. We see that the majority of features away from first to third columns are far smaller in size with only a few being larger than one. The magnitudes clearly indicate the difference between features at the beginning of the function compared to the rest of the function. However, it is

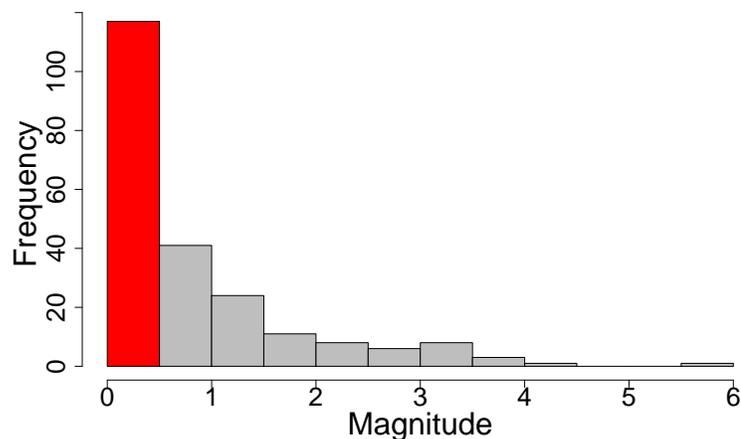


Figure 6.17: Histogram of the magnitudes of all the features found from the intensity functions inferred from the constant stimulus experiments.

unclear whether the magnitude can easily be used to differentiate between features in the middle and end of the intensity functions.

We now aim to cluster the functions. We use the information provided above to help us choose what characteristics to partition the functions on. We decide to cluster the functions that contain a bump outside the initial two columns (to remove bumps corresponding to the initial decrease). This partitions the functions into two groups; the 28 functions containing no bumps and the 32 functions containing bumps. In Figure 6.18(A-B) we show the clusters, where (A) contains bumps and (B) does not. In the Figure we see the intensity varies more for the functions with bumps. We

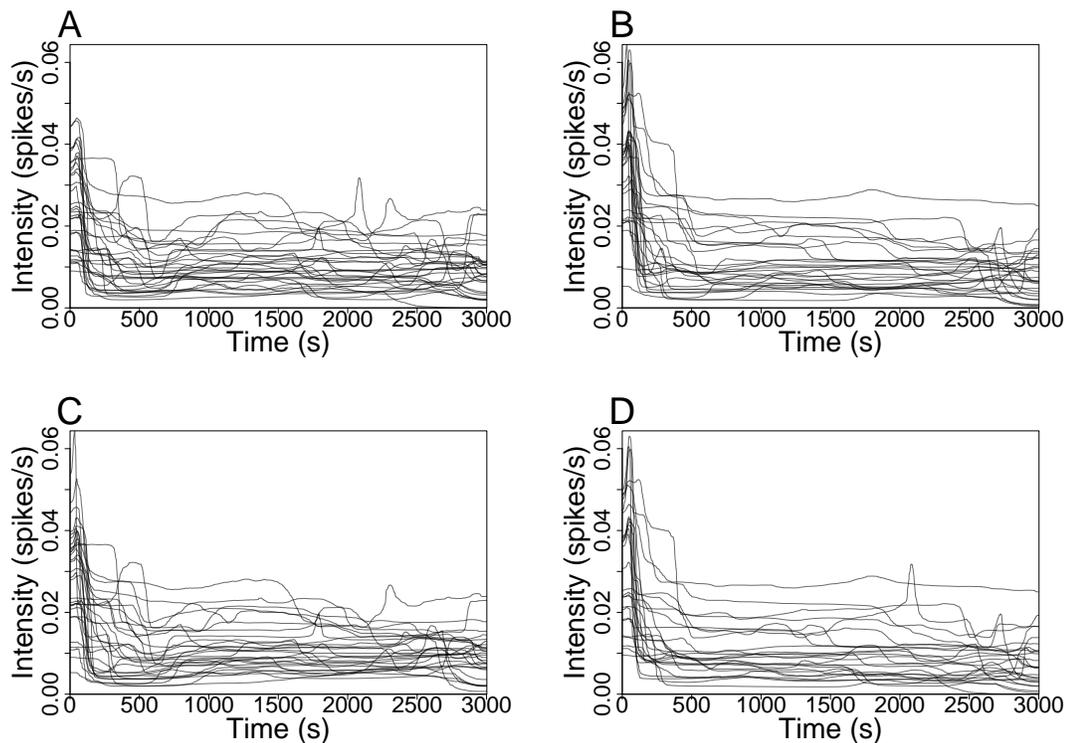


Figure 6.18: Clusters of the intensity functions inferred from a constant stimulus experiment. (A) and (B) corresponds to clusters with and without bumps respectively. Whereas, (C) and (D) corresponds to clustering on functions that do and do not rebound after the initial decrease in intensity respectively.

can also cluster on those functions that have an increase after the initial decrease. This splits the functions into two groups, both containing 30 functions. The clusters can be seen in Figure 6.18(C-D), where (C) corresponds to the functions with the increase and (D) without. The difference can be viewed by concentrating on the region [500s, 1200s], where in (C) we see that most functions increase in the region.

Our analysis shows strong cell-to-cell variability to HEK293 cells challenged with constant stimulus. In almost all cells we see an initial decrease in spiking rate with varying magnitude. From the clustering we have found that some cells experience a rebound in intensity after the initial decrease. We also find that some cells experience a bump in intensity, whereas others remaining mostly constant after the initial transience.

6.4.3 Step change experiments

We now apply the clustering method to intensity functions inferred from HEK293 cells challenged with a step-change in carbachol for 6860s. The cells were first stimulated with 10 μ M carbachol and then with 50 μ M carbachol, where the solution was exchanged at 3430s. For each Ca²⁺ spike sequence we fit the inhomogeneous Weibull ISI distribution with a PWC prior for the intensity function. The means of the posterior intensity function distributions are shown in Figure 6.19(A). We see that the intensity begins high and decreases within the first 1500s, then the intensity appears relatively flat until the intensity spikes when the stimulus is exchanged.

We now decompose each of these functions into their features, where we use $q = 32$ Haar bases. Thus the functions are partitioned into 16 intervals [0s, 429s], [429s, 858s], \dots , [6431s, 6860s] where features can occur. Due to the small values the intensity takes we multiply each function by 100 to more easily recognise features in the Haar bases. The threshold used for each function was $\Delta/10q$. The resultant matrices are shown in Figure 6.19(B,C) for the features and magnitude, respectively.

Consider first only the features. We can see via the key that the functions contain increasing and decreasing regions and up to three individual bumps. However, only the tenth function contains increasing regions. In fact, the tenth function may be an

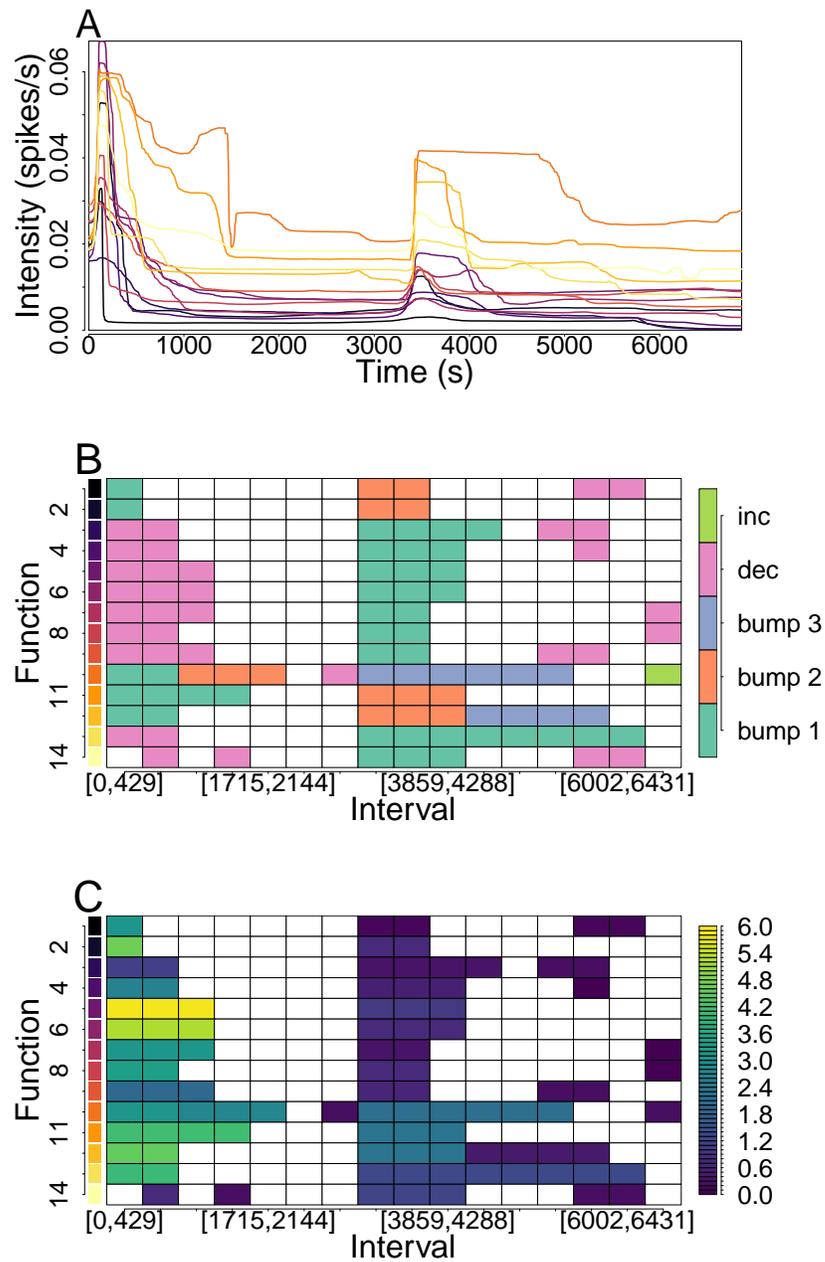


Figure 6.19: (A) shows the intensity functions inferred from the step change experiment. (B) shows the extracted features of each function and (C) the feature's magnitude. The functions in (A) are coloured corresponding to the row that describes their features in (B) and (C).

outlier, as it is the only function to have two features at the start of the function and the only function to contain an increasing region. All functions contain a feature in the first 900s, with either a bump or a decreasing interval. All functions also have a bump when the stimulus is exchanged, however the length of the bump varies. The only other common feature is that some of the functions contain a decreasing interval between the stimulus change and the end of the experiment.

We now consider the magnitudes of the features in Figure 6.19(C). We see that the features in [0s, 1286s] tend to have the largest magnitude for a single function. However, the magnitude varies significantly between functions. The bumps that occur from the exchange of stimulus [3000s, 3500s] have magnitudes ranging from 2.36 to 0.11 and in all cases they are smaller than the feature at the beginning of the experiment. We also find that the decreasing regions in intensity after the stimulus is exchanged have small magnitudes between 0.26 and 0.10.

After taking into account the features and their magnitudes, we choose to cluster on the magnitude of the bump in the region [3000s, 4000s]. We find that the functions {10, 11, 12} have larger bumps compared to the other functions. After accounting for this, we also split the functions into two whether they have a decreasing region after this bump. We find that the functions {1, 3, 4, 7, 8, 9, 14} have a decreasing interval and {2, 5, 6, 12} do not. The clusters can be viewed in Figure 6.20, where (A),(B) and (C) correspond to the clusters {10, 11, 12}, {1, 3, 4, 7, 8, 9, 14} and {2, 5, 6, 13} respectively.

In the step change experiment we find that all functions initially begin with a high spiking rate which reduces significantly over the first 1000s. Once the stimulus is exchanged all cells experience an increase in spiking intensity that reduces after 1000-1300s. Hence, we find that most cells experience a strong change in spiking rate at the initial exposure to stimulus and again when stimulus is changed. In between these two features the change in intensity is smaller. However, decreases in intensity is found for some of the functions after the second peak in intensity.

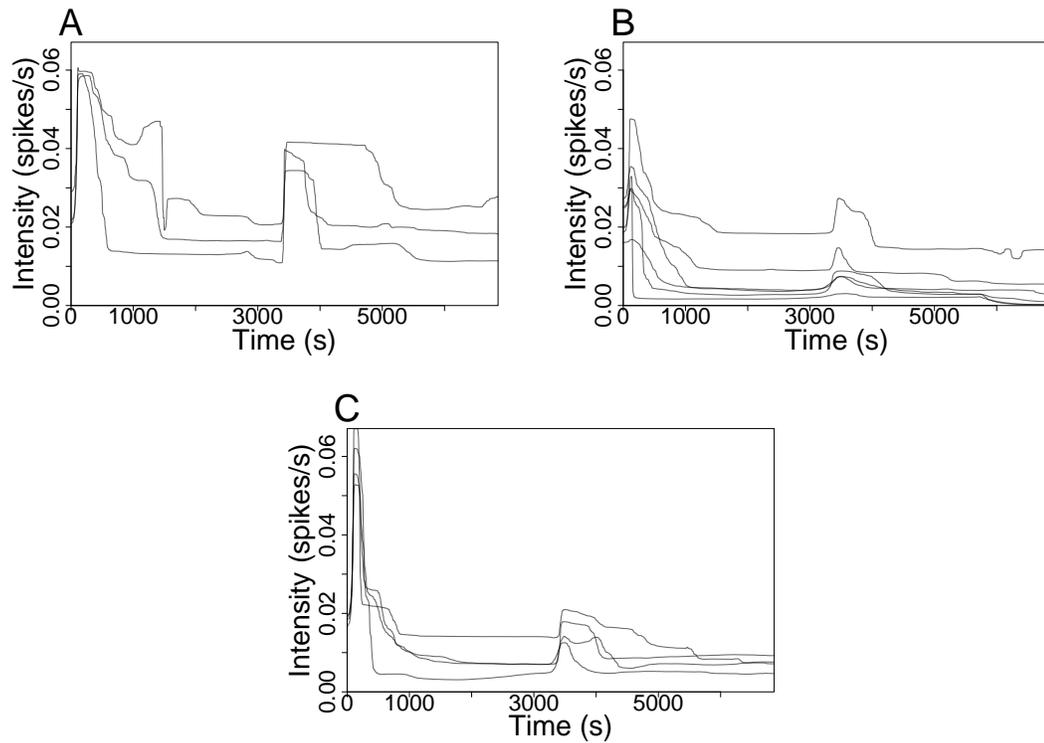


Figure 6.20: Clusters of the intensity functions obtained from a step change experiment. We cluster into 3 groups; (A) functions with a large bump, (B) functions that have a decrease after the bump in $[3000\text{s}, 4000\text{s}]$ and (C) those that do not have a decrease afterwards.

6.5 Summary

Intensity functions inferred from HEK293 cells applied with the same stimulus vary in shape. The shape of each intensity function contains information about how a single cell responds to the stimulus. Thus, by analysing these functions we can find common responses to the stimulus. To do this we suggested clustering the intensity functions according to features that each function contains. Our initial investigation found that recent methods — proposed by Tilunaite and Delaigle — for clustering functions did not perform well with our clustering ideology. In particular, neither method could accurately partition a set of constant functions and a set of functions

with a bump. In light of this we have created a new method to cluster functions, where we first decompose each function into its features before clustering.

Our method takes the intensity functions as its input. Then by utilising the Haar basis each function is decomposed into its features, where each feature has a corresponding magnitude. The features and magnitudes are outputted into matrices, which allows for easy comparisons between functions. We then use the matrices to cluster the functions. The benefit of this method is that we can cluster the functions according to the features that interest us.

Our method for finding the features relies entirely on the structure of the Haar basis, and we have shown that this basis can accurately find features given a good threshold. However, we have not investigated to what extent other bases can also find features. In future work it may be worth exploring if other bases — such as Daubechies wavelet — are better suited to calculating features. Moreover, we have only empirically found that $\Delta/10q$ acts as a good threshold in most cases. Further investigation into finding an optimal threshold would help decomposing functions into their features.

Currently all clustering is performed by examining the matrices of features and magnitudes. This method allows for direct input from the user in choosing the clusters, which enables the choice of particular clusters; for example clustering on the number of bumps. However, if our dataset contained hundreds of intensity functions this method could become difficult to implement. It may be beneficial to add a method which directly compares the features and magnitudes. For example a method based on the Jaccard distance — a measure of how dissimilar two sets are — could potentially be used to compare the features of two functions.

We applied our methods to HEK293 cells challenged with constant stimulus and a step change in stimulus. From the constant stimulus data we found that all cells have a peak in spiking rate after the stimulus is applied, which quickly decreases. When using $q = 32$ bases we found that most transient period occurs within the first 12.5% of the experiment's length. However, the amount the intensity changes varies considerably with magnitudes from 0.22 to 5.59. Thus, the decrease in intensity

occurs for a consistent amount of time at the beginning of the experiment, but the magnitude of the decrease varies considerable across the cells. We also found that half the cells experience a recovery where the intensity increases after the initial drop in intensity at the beginning of the experiment. This recovery always has smaller magnitude than the decrease prior. This could indicate that the quick Ca^{2+} spiking after the cell is challenged with stimulus induces a negative feedback mechanism, which causes the spiking rate to decrease beyond its stationary value. Furthermore, this might suggest that the initial transiencies lasts longer than originally thought, where the transient period also includes this increase in intensity. In addition, we found that over half the cells contain a bump in spiking rate away from the start of the experiment. This implies that the intensity function is not constant. Thus, the assumption that the spiking rate is stationary away from the beginning of the experiment may not be accurate, and requires further investigation. Finally, we discovered that as time progressed the features become less common throughout the cells, which could imply temporal heterogeneity.

We found that HEK293 cells in step change experiments have two peaks in intensity at the times when initially stimulated and when the stimulus is exchanged. Furthermore, the first peak always has larger magnitude than the second, which is counter intuitive given that the second stimulus is stronger than the first. We find that the inferred intensity does partially mirror the stimulus, as a peak in intensity can be found at the times that the stimulus is applied/exchanged.

CHAPTER 7

Web applications of our methods

To supplement the methods described in this thesis we have created an R package that implements them. The package can be found at <https://github.com/JPNotts/CaliumSpikes>. This allows for our methods to be openly available for anyone's use. We have also created three applications (apps) to accompany the package. The apps are used for: converting raw data into spike times, simulating spike sequences and decomposing intensity functions into their features. These apps provide a straightforward, easy to use interface for the end user, without specialised knowledge of the computer language R. Moreover, the apps for raw data and decomposing intensity functions allows for downloading and uploading inputs for the ease of checking previous work and alterations.

We have not included an app to fit our ISI models to Ca^{2+} spike sequences due to the large computational and memory demands. However, to make model fitting as easy as possible we have included a vignette explaining how to use the package to fit ISI models. Moreover, we explain how to run multiple processes simultaneously reducing the overall time to fit multiple ISI models.

In the following sections we shall explore each app and give an example of how

they can be used.

7.1 Web application for viewing/analysing raw data

We have created an app to view the raw time series data of Ca^{2+} concentration with a user-friendly method of thresholding the data to calculate Ca^{2+} spike times. Depending on the experiment the raw data for each cell is either the fluorescent ratio or the Ca^{2+} concentration against time. The only difference between the two is the scale used. Therefore, both can be used with this app. However, to simplify the app we always regard the raw data to be the Ca^{2+} concentration over time. The app can be found at <https://shiny.maths.nottingham.ac.uk/pmxjp8/ViewData/>.

This bespoke app has been created with Ca^{2+} data in mind. For example, it is often required to remove transient regions from the experiment and remove linear trends in the ISIs [6]. Both of these features are implemented in the app allowing the user to view how transients and linear trends effect the generated Ca^{2+} spike sequence. Moreover, it is often difficult to decide the concentration to threshold the raw data. For example the magnitude of the Ca^{2+} spikes may vary over the experiment. Therefore, the app allows for varied thresholding to capture all spikes using the user's discretion. We can also zoom into regions of the time series data to give a clearer picture of the concentration at that time. The Ca^{2+} concentration is often noisy, which can lead to false spikes being detected. For example, a spike time may be found when the noise causes a small local increase in Ca^{2+} concentration. To negate this we include a closeness parameter which merges spikes together if they are too close. Finally, the app allows for the uploading and downloading of threshold parameters, allowing users to save and return to analysing the raw data at a later date.

7.1.1 Example of using the app

To assist the user, a step-by-step guide of instructions is provided on loading the app. We now give an example where we use the app to obtain spike sequences from raw

data.

First we must upload raw data to the app. To do this we **press the ‘Input Data’ button**, and a pop up window appears. On the popup we are presented with two upload boxes, the first for the raw data and the second for loading threshold details. We **upload the raw data into the input box** from our local machine, where the file is formatted such that the first column corresponds to the experiment time and subsequent columns are the Ca^{2+} concentrations for each cell. On **pressing the ‘Ok’ button** the popup disappears and we are presented with the concentration plot — Ca^{2+} concentration against time — for the first cell in the dataset. We want to obtain spike times by thresholding the raw data. The options to threshold are shown in the ‘Thresholding’ section of the sidebar. After considering the plot we **enter 0.7 into the ‘Thresholding at?’ box**. This causes a red threshold line to appear on the concentration plot along with red ticks on the time axis representing the spike times and the experiment’s length. Furthermore, a new plot showing the ISI times against the order they arise appears underneath the concentration plot. The ISI plot contains a linear fit, showing any trends in the ISIs. Beneath the ISI plot are written the obtained spike times together with the end time of the experiment. This is shown in Figure 7.1.

To aid in the choice of threshold we can vary the time and concentration (conc) axes in the concentration plot. This is especially useful to zoom into regions where the spikes occur rapidly proportional to the experiment’s length. Moreover, if the concentration is far larger in one region — often at the onset of stimulus — compared to the rest of the experiment we can scale the concentration axis to make spikes distinguishable. By default the conc axis is between the maximum and minimum values for the concentration data for the current cell and the time axis is the length of the experiment. In the sidebar we find a section called ‘Change concentration plot axes?’, where we have two checkboxes for the time and conc axes, respectively. To change the time axis simply **click and drag a box on the concentration plot** for the region you want to zoom into. On releasing the click the concentration plot has zoomed the time axis into the region of the drawn box, whilst the conc axis remains

View Cell Data

TOGGLE HELP SCREEN

Click the button below to load data

INPUT DATA

Cell number

1

Change concentration plot axes?

Default conc axis?

Default time axis?

Thresholding

Threshold at?

0.7

Transient period

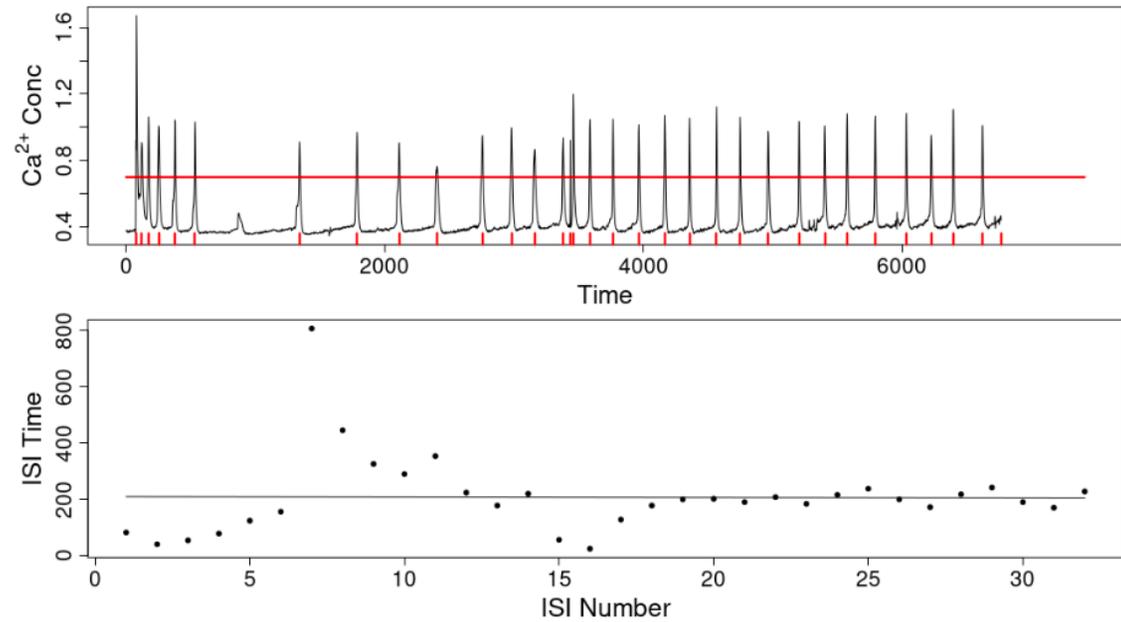
Remove linear trends?

Limit closeness of spikes?

DOWNLOAD

Click the download button if you wish to save your spikes!

You have selected cell 1, maximum cell number is 24.



The spike times are: 82, 122, 176, 254, 378, 534, 1342, 1788, 2114, 2404, 2758, 2982, 3160, 3380, 3436, 3460, 3588, 3766, 3966, 4168, 4358, 4566, 4750, 4966, 5204, 5404, 5576, 5794, 6036, 6226, 6396, 6624, 6766.

Figure 7.1: Screenshot of the app for viewing/analysing raw data when we threshold at 0.7 .

unchanged. The ‘Default time axis?’ checkbox has become unchecked revealing inputs for the maximum and minimum times for the plot window. If we want to rescale the conc axis **uncheck the default conc axis checkbox**. This rescales the conc axis between the minimum and maximum concentration found in the selected region. The user can change the axes limits by changing the values in the provided maximum and minimum input boxes for the time and conc axes. By **rechecking the default time axis and default conc axis checkboxes** the axes returns to their default setting.

Between 800s and 1000s we see a small spike in the Ca^{2+} concentration that we may want to include as a spike time. However, if we lowered the threshold time we may lose some of the other spikes — i.e the first three spike times. So, to include this spike we can use multiple thresholds by **entering ‘0.7,0.45,0.7,800,1000’ into the ‘Threshold at?’ box**. This tells the app that between 0s and 800s you want to threshold at 0.7, between 800s and 1000s we threshold at 0.45 and for times greater than 1000s threshold at 0.7. In figure 7.2(A) we see the red line has updated for our new threshold and a new red tick is produced on the time axis representing the new spike time. Moreover, we find that the large ISI time of around 800s has been split into two new ISIs of approximately 320s and 480s in the ISI plot.

Note that the Ca^{2+} 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’ checkbox** in the sidebar we are prompted to enter the transient regions at either the start, middle or end of the experiment. We **enter the transient periods to be between 0s and 200s and between 3300s and 3500s**. In figure 7.2(B) we see the affect of adding the transient periods. In the concentration plot, red boxes appear over the transient periods. The first transient period alters the start time of the calculated spike times, which now begins at the time of the first spike after 200s, namely at 254s. We use this approach such that we only need to choose the transient period between two spikes, i.e. any transient period between the spikes at 177s and 254s gives rise to the same spike sequence. For example, without transient regions the first 6 spike times are given by 82s, 122s, 176s,

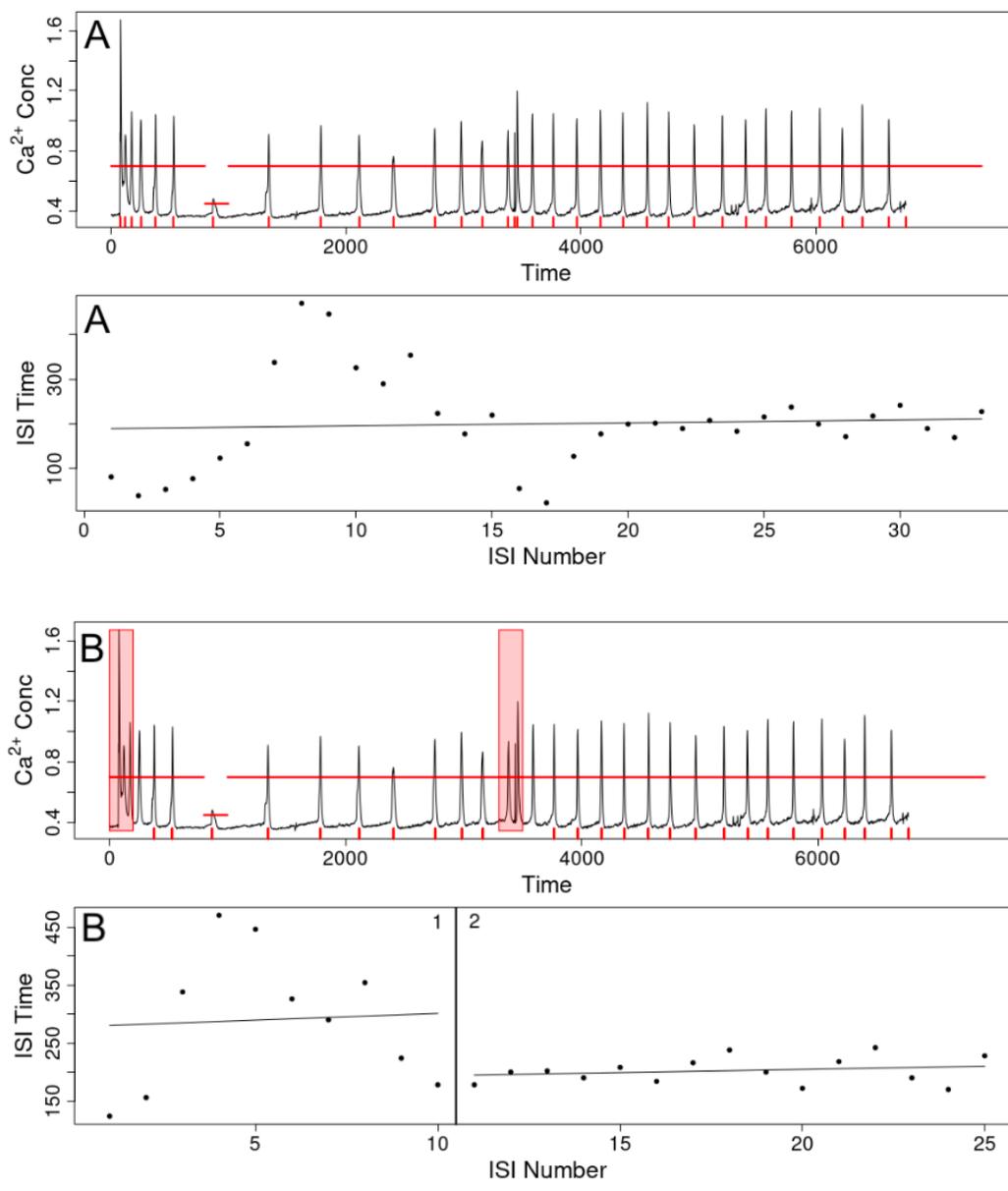


Figure 7.2: The concentration and ISI plots shown in the app when we (A) change the threshold to $'0.7, 0.45, 0.7, 800, 1000'$ and (B) add two transient regions.

254s, 378s, 534s. By adding the transient period [0s, 200s] we cut the first 254s off the experiment time — the time of the first spike after the transient period. Thus, the new spike times are given by 134s, 146s, etc. This maintains the ISI times, e.g. $378s - 254s = 134s$. This is shown in the concentration plot by removing the red ticks for the spikes up to 254s. Similarly for the transient period in [3300s, 3500s], we remove the time between the last spike before 3300s to the first spike after 3500s. This is shown by removing the red ticks of the spikes under the transient region and the red tick corresponding to the first spike afterwards. Since we use a transient period in the middle of the experiment the ISI plot is now split into two windows. The first window corresponds to the spikes before the middle transient period and the second window for the spike times afterwards. Both subsets of spikes have their own linear fit line. We choose to split the ISIs in this manner because a middle transient period usually arises due to a change of stimulus. Therefore, the spikes after the transient region do not share the same trend as the spikes before it.

If we wish to remove linear trends in the ISIs we **press the ‘Remove linear trends?’ checkbox**. This alters the spike times by removing the linear trends found in the ISI times. If the ISIs are split into two windows this is performed for both subsets of ISIs. Visually, this has no affect on the concentration plot. However, the ISI times change and the lines of best fit are now flat, showing there are no linear trends in the adjusted ISIs.

We then change to the second cell by **entering 2 into the ‘Cell number’ box**. We threshold the cell by **inputting 0.85 into the ‘Threshold at?’ box** — the concentration plot, in the region [400s, 800s], is shown in Figure 7.3(A). We see that using this threshold leads to false spike times. For example, we get two spikes in [480s, 500s], which is caused by noise in the Ca^{2+} concentration. To solve this problem we **check the ‘Limit closeness of spikes?’ checkbox** and an input box appears in the sidebar for ‘closeness’. We **enter 10 into the ‘closeness’ input**, which merges spikes that are within 10s of each other. This results in the false spikes been removed, as shown in Figure 7.3(B).

We can repeat this process for all the others cells in the dataset. At any stage

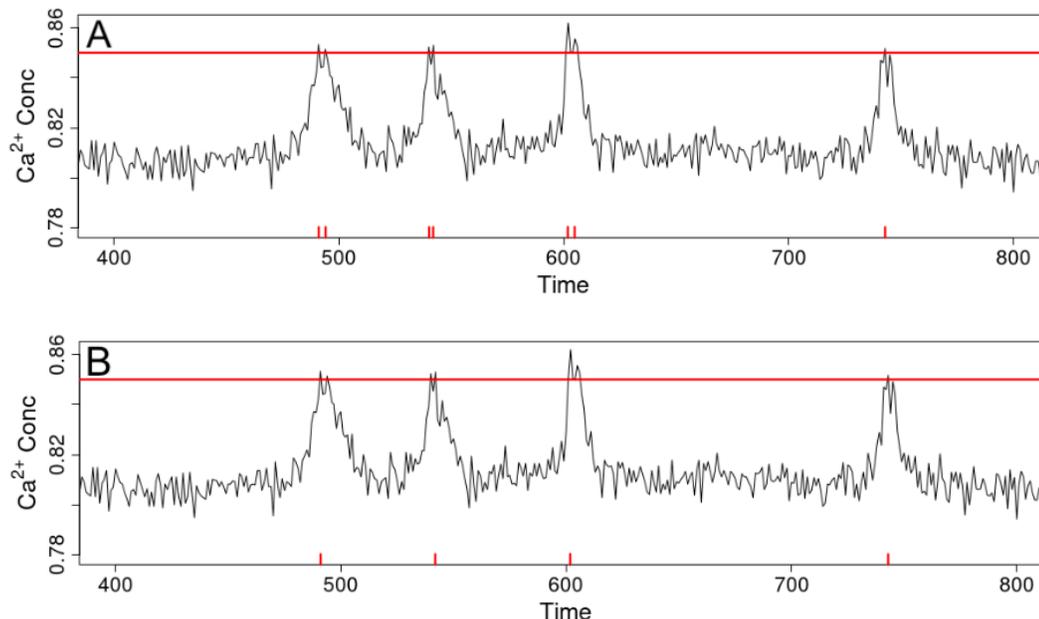


Figure 7.3: Screenshots of the concentration plot for the second cell using a threshold of 0.85. The closeness parameter is set to 0 and 10 in (A) and (B), respectively.

we can download the spikes and threshold details for later use. **Pressing the ‘Download’ button** will prompt the user to select a location to download a folder on their local machine. In this folder there are three files. The first file ‘*Details.txt*’ informs the user of the file name of the raw data used. The second file ‘*spikes.csv*’ stores the calculated spike sequences as columns in a table, where the i th column contains the spikes for the i th cell. Cells which have not been thresholded are denoted by a column of NAs. The third file ‘*store.csv*’, contains the threshold details for each cell in the dataset. This file can be uploaded into the app with the raw data allowing changes to the thresholds and checking how spike times were created.

7.2 Web application for simulating spikes

We have created an app to generate surrogate spike sequences using the methods described in Section 2.3. The app can be found at <https://shiny.maths.nottingham.ac.uk/pmxjp8/SimSpikes/>.

This app supplies an accessible user interface for simulating spike sequences. It

also provides graphical techniques to allow the user to visually decide whether the spikes sequences match their needs. The created spike sequences can be used as surrogate data to Ca^{2+} models when experimental data is difficult to obtain.

In this app we use the sidebar to enter the ISI distribution we want to generate spike sequences from and the number of sequences we require. Once inputted a plot of the intensity function and a raster plot of the spike times are presented. The user may then download the spike sequences which can easily be used as an input in other applications.

7.2.1 Example of using the app

On opening the app the user is presented with a guide explaining how to use it and a sidebar containing inputs — the ISI distribution and the number of spike sequences we want to generate. We shall give two examples of using the app. Firstly, we choose the following inputs:

- *ISI distribution = 'Gamma'*. A choice between an inhomogeneous Exponential, Gamma, inverse Gaussian, log-normal or Weibull ISI distributions.
- *ISI parameter = 32*. The parameter of the chosen ISI distribution.
- *Refractory period = 0*. The time after a spike when the next spike cannot occur.
- *Intensity function = $\cos(t/2.3) + 1.2$* . A positive function defined in terms of the variable t over the experiment's length.
- *End time = 40*. The length of time we simulate spikes for.
- *Number of sequences = 10*. How many spike sequences we want to generate.

Once the desired inputs have been entered we **press the 'Get spikes' button**. The main panel is updated with a figure containing the intensity function and the raster plot of the generated spike sequences — with a maximum of 20 sequences shown. Furthermore, underneath the figure the first spike sequence is written. This is shown

Simulate spike sequences

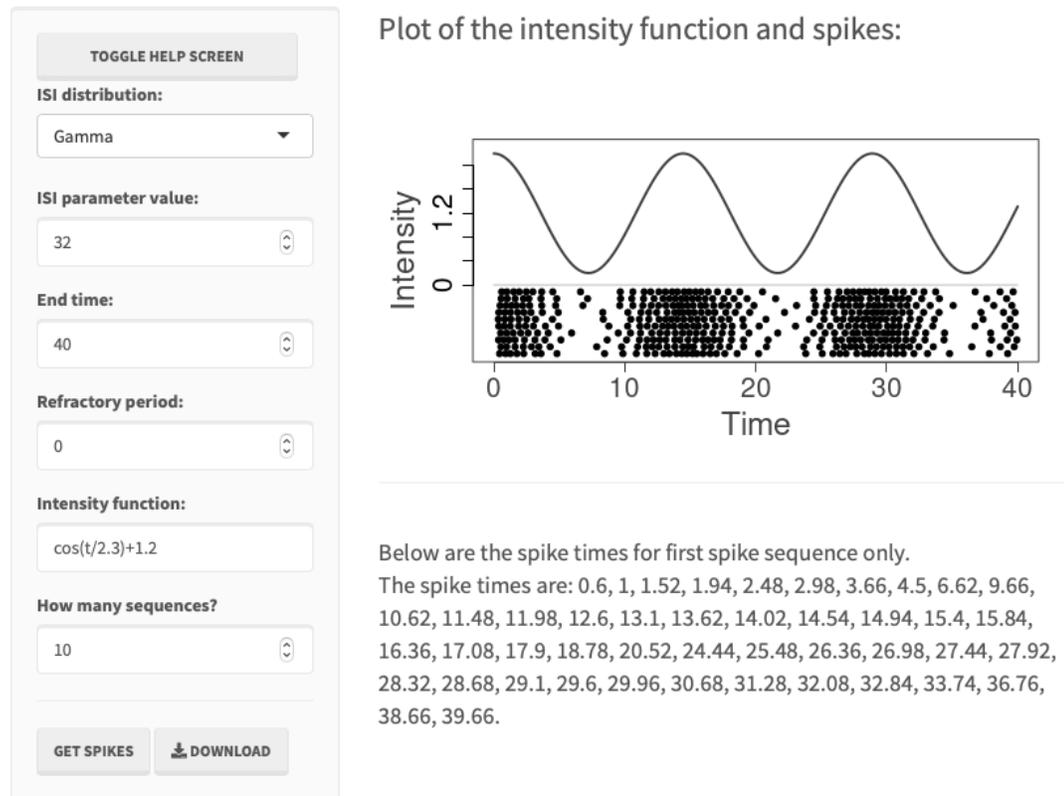


Figure 7.4: Screenshot of the simulating spikes app.

in Figure 7.4. If we only require a single spike sequence it can easily be copied from the app. If we require more than one spike sequence we can **press the ‘Download’ button**. This creates a zipped folder on the user’s local machine containing two files. The first file ‘*spikes.csv*’ contains each spike sequence as a column of a table, which can easily be used in other applications. The second file ‘*details.txt*’ contains the parameters used to create the spike sequences.

In the second example we will consider a step function as the intensity function. The inputs are set to:

- *ISI distribution* = Weibull
- *Refractory period* = 0
- *End time* = 40
- *Number of sequences* = 7
- *ISI parameter* = 7

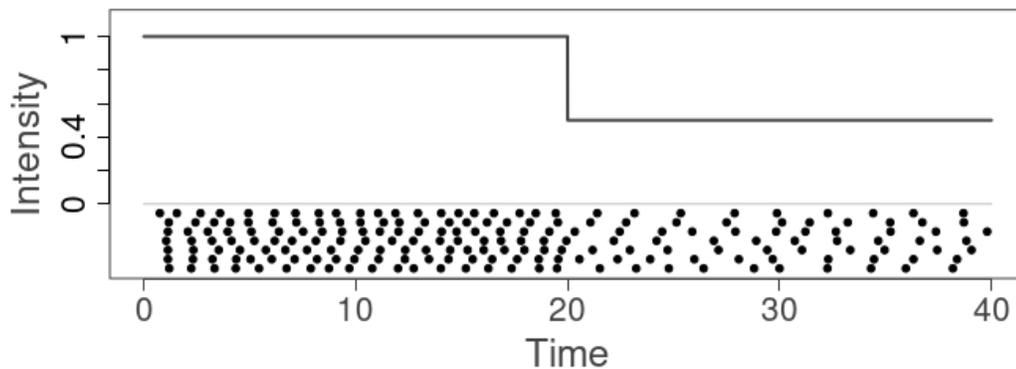


Figure 7.5: Screenshot of the results panel for the second example where the intensity function was $c(t[t < 20] * 0 + 1, t[t > 20] * 0 + 0.5)$.

- *Intensity function* = $c(t[t < 20] * 0 + 1, t[t > 20] * 0 + 0.5)$

The input for the intensity function may look confusing. However, this is the format required for the app to understand the step function. The intensity function is generated by applying the input provided to the vector t which partitions the experiment's length into 8000 equal pieces. Thus, in the example of the step function we split t into two parts by $t[t < 20]$ for the first 20s and $t[t \geq 20]$ for the last 20s. Then to get the constant intensity we multiply each part by 0 and add the required intensity. In our case we set the intensity to 1 in $[0s, 20s]$ and 0.5 in $[20s, 40s]$. Then we join the two parts together by using $c()$ — the app's method to concatenate vectors. The benefit of this approach is that the user can create complex intensity functions using the app.

Pressing the 'Get spikes' button we are presented with the intensity function and spike times as shown in Figure 7.5. We see that the inputted intensity function matches the previously described step function.

7.3 Web application for finding features in intensity functions

This app is used for the decomposing intensity functions into their features, which can be used to cluster them. The app can be found at

https://shiny.maths.nottingham.ac.uk/pmxjp8/Cluster_features/.

This app provides an interface for generating the features — and their corresponding magnitudes — of intensity functions. After uploading intensity functions the app will automatically find the features. The user may check the features of any intensity function and change them if required. To search for common features between intensity functions the feature or magnitude tables can be sorted by the region of interest. Finally, the app allows the user to download the features and magnitude tables for later use.

7.3.1 Example of using the app

On opening the app the user is greeted with a panel where they can input the intensity functions and a link to the step-by-step guide for the app, see Figure 7.6. We shall now go through an example of uploading data to the app, calculating features and looking for clusters from the features.

We first upload the intensity functions into the app. This is done by **uploading the intensity functions via the ‘Load functions’ input**. The data must be contained in a *.csv* file where the first row consists of the time indexing and each subsequent row is an intensity function. After the upload is completed, we need to choose how to decompose the intensity functions into their features. We have two choices; either we upload features from previous analysis or use a Haar decomposition. We **select a Haar decomposition with 16 regions**. To finalise our decision we **press the ‘OK’ button**, this causes a popup to appear in the bottom right hand corner with a loading bar. The popup informs on the process of creating the feature and magnitude tables. Once completed, the app takes us to the *Overview tab* where we are presented with the feature and magnitude tables, as shown in Figure 7.7. The

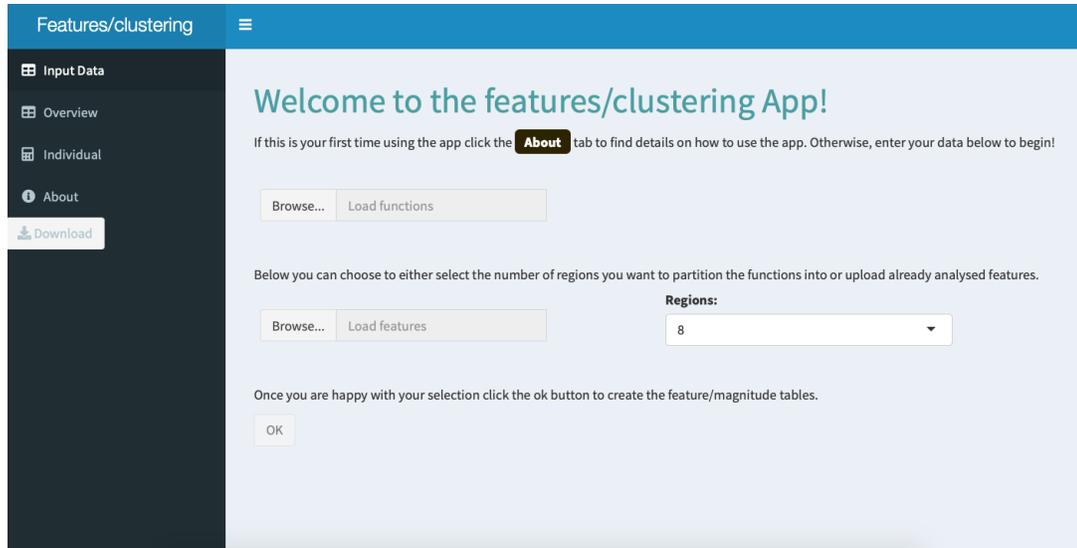


Figure 7.6: Screenshot of the opening panel of the app.

feature table shows the features for the first 10 intensity functions and we can shift between functions by using the *previous* or *next* buttons in the bottom right corner of the table. Depending on the number of regions selected we may need to scroll left and right to see all the regions. The features are coloured to make similarities easier to distinguish — for example a decreasing region is coloured pink. Similarly, in the magnitude table the darker the red the larger the magnitude of the corresponding feature.

To check that the automated features give an accurate representation of the intensity functions we can **switch to the *Individual tab* in the sidebar**, this is shown in Figure 7.8. In this section we can view single intensity functions and alter their features. The main panel is split into three sections via horizontal lines. The top section contains three options. The first option *‘Show Haar?’* is a checkbox that when selected overlays the Haar representation of the intensity function (in red) in the intensity function plot. The second option *‘Threshold level’* gives the threshold used to decide where the features are. This can be altered to change the features found for the intensity function. For more information on the threshold see Section 6.2. The third option *‘Functions:’* allows the user to switch between the

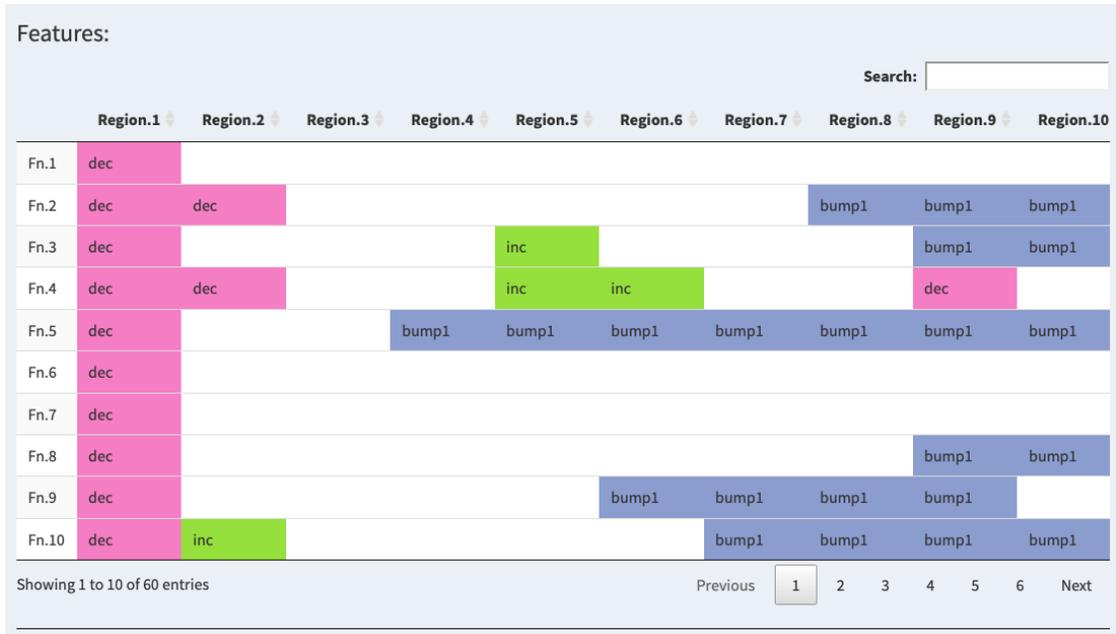


Figure 7.7: Screenshots of the feature (top) and magnitude (bottom) tables taken from the app.

intensity functions found in the dataset. The middle section contains the plot of the intensity function, which has been split into 16 regions vertically. This allows the user to easily compare the features and the intensity function. For example, we clearly see that region 1 is decreasing. The bottom section outlines the features

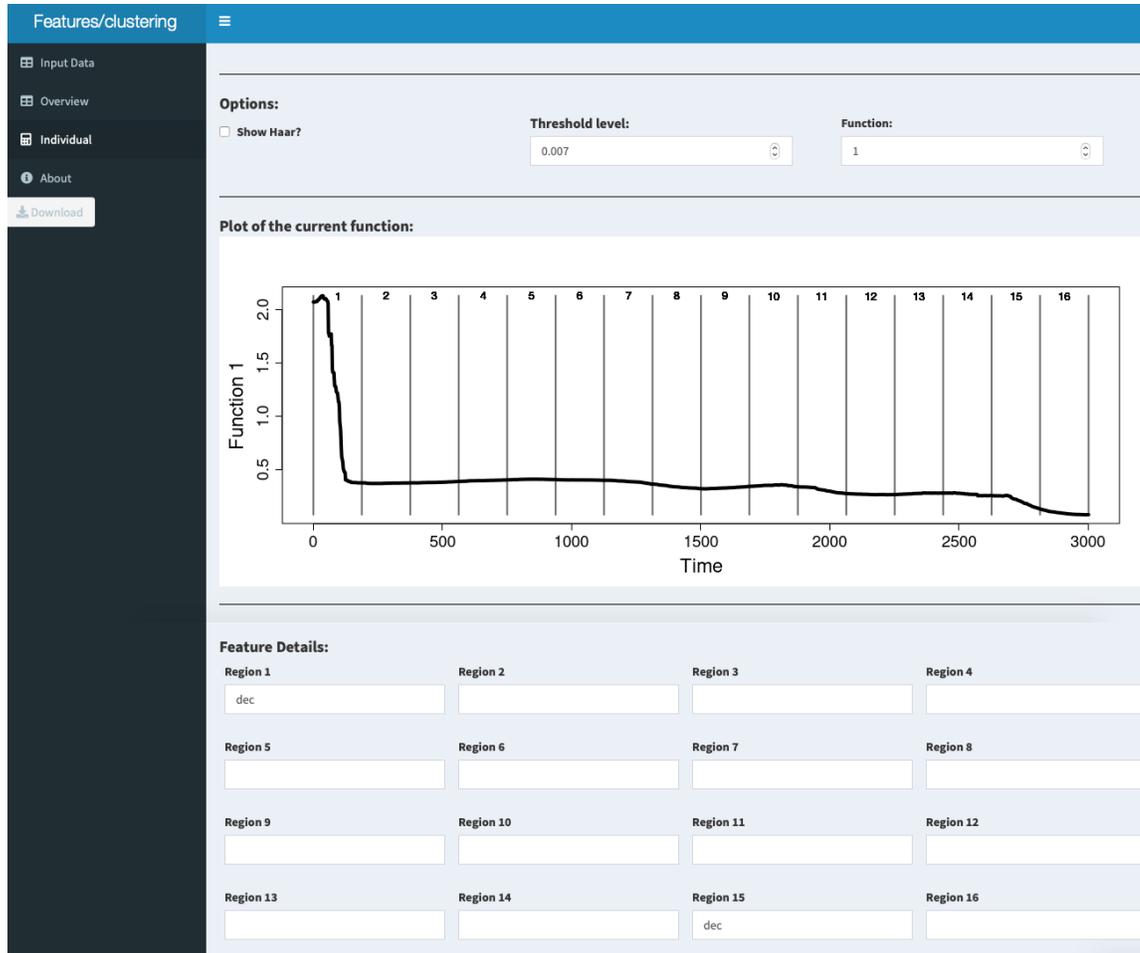


Figure 7.8: Screenshot of the *Individual tab* taken from the app.

for the current intensity function. These can be updated by the user to match the features they observe in the intensity plot. Any changes made in the *Individual tab* are automatically updated in the *Overview tab*.

We can use the feature and magnitude table to aid us in clustering the functions. By **clicking ‘Region.1’ on the feature table** this sorts the table by the features found in the first region. This is helpful to differentiate functions that start with a decreasing region and those that do not. This can be applied to any region on both the feature and magnitude tables. Moreover, if we are only interested in intensity functions that contain two bumps we can **enter ‘bump2’ into the search bar** and only functions that contain at least two bumps will remain in the feature table. Therefore, the sorting and searching of the magnitude and feature tables simplifies

the clustering process.

Once we are happy with the feature table we can download it, along with the magnitude table, by **pressing the ‘Download’ button provided on the sidebar**. This creates a folder on the user’s local machine containing three files. The first *‘fns.csv’* is the dataset of intensity functions used to create the feature and magnitude tables. The other two files *‘features.csv’* and *‘magnitudes.csv’* contain the feature and magnitude tables, respectively. This allows the features and magnitudes to be examined externally as well as storing all the files collectively in one location reducing the risk of file misplacement. Furthermore, *‘fns.csv’* and *‘features.csv’* can be uploaded into the app allowing the user to revisit the data.

8.1 Overview

In this thesis, we have developed a statistical model of Ca^{2+} oscillations that captures the intrinsic heterogeneity of Ca^{2+} responses and the dynamic stimulation that cells are challenged with in vivo. We applied our approach to Ca^{2+} oscillations from HEK293 cells and astrocytes and investigated their common features, and clustered cells that have similar responses.

We began with the modelling framework for Ca^{2+} oscillations first developed by Tilunaite et al. [16]. This consisted of representing Ca^{2+} oscillations as a stochastic point process, which is defined by its ISI distribution. To account for heterogeneity in the Ca^{2+} oscillations and time-dependence of the stimulus challenging cells Tilunaite et al. utilised an intensity function in the model to capture how the ISIs vary over time. Therefore, their model was defined by the type of ISI distribution used — Exponential, Gamma or inverse Gaussian — together with the intensity function and ISI parameters. Tilunaite et al. utilised a Bayesian framework to fit their model to Ca^{2+} spike sequences, where each model parameter has a prior belief which is

then updated to get the posterior belief after observing the data. In their work they took a non-parametric approach for the intensity function, assigning it a GP prior distribution. To fit their model they approximated the posterior distribution by using a Laplace's approximation.

In this thesis, we extended Tilunaite's work in three critical ways. Firstly, we increased the number of ISI distributions used to describe the ISI dynamics. In particular, we explained how to create an inhomogeneous ISI distribution using any continuous probability distribution with support on the positive real numbers. Our approach imbued meaning to the intensity function to describe the average spiking rate irrespective of the spiking history. We found that the inhomogeneous inverse Gaussian ISI distribution used by Tilunaite et al. does not agree with our formulation of the intensity function — it does not describe the average spiking rate. Therefore, we provided a new parameterisation for the inhomogeneous inverse Gaussian ISI distribution. We also included the log-normal and Weibull distributions as candidates for the ISI distribution.

Secondly, we considered additional priors for the intensity function. In addition to the GP prior used by Tilunaite et al, we added the constant and PWC priors for the intensity function. This allowed us to explore different functional forms for the intensity function. Moreover, constant and PWC priors are computationally cheaper than GPs. Therefore, the additional priors may give rise to a computationally cheaper method to generate similar posterior intensity functions as the GP.

Thirdly, Tilunaite et al. approximated the posterior distribution, which can lead to approximation errors. For example such an approach struggles to capture if the posterior distribution is multimodal. Therefore, in this thesis we changed the model fitting philosophy and use exact inference. To do this we developed MCMC algorithms to sample from the posterior distribution. In general, we use Metropolis-within-Gibbs sampling over the parameter space, where we sample the intensity function and ISI parameter individually. For the ISI parameter we used

a Metropolis-Hastings algorithm. For the intensity function we developed efficient MCMC algorithms tailored to each prior. For the constant prior we used a Metropolis-Hastings algorithm. For the PWC prior we utilised a RJMCMC algorithm. The advantage of this algorithm is that it can change the dimension of the search space by birth/death/split/merge moves. In our case, it allowed for moves between PWC functions with varying number of pieces. For the Gaussian process prior we used an under-relaxed algorithm. The main advantage of this approach is that in each iteration a new intensity function is proposed over the entire domain, which is preferable compared to proposing a change at each time index individually. The MCMC algorithms are described in Chapter 2 for the inhomogeneous Gamma ISI distribution.

Although sampling from the posterior is advantageous it comes with computational challenges. These are described and resolved in Chapter 3. We summarise the three challenges below.

One obstacle is sampling the posterior distribution when the intensity function has a GP prior because it is slow, often due to the requirement of repeated sampling from the GP prior. We resolve this issue using two approaches. Firstly, if we fix the length scale of the GP prior, we used the spectral representation of a Gaussian process to improve the speed we could sample the GP prior distribution. Secondly, when we infer the length scale we used an approximation method called Mean Projection Approximation. This fits the GP prior distribution over a smaller time indexing set and then projects this function onto the full time index. Both of these approaches drastically decrease the time taken to sample from the posterior distribution.

Another obstacle is understanding when we can infer the length scale of the GP prior directly. As we often fit our models to a single Ca^{2+} spike sequence there is only limited information about the value and shape of the intensity function. For example between two Ca^{2+} spikes we do not know if the spiking rate is zero in this region or we were just unlucky to find no spikes in this region. Thus, we do not know if the intensity function remains flat in this region or varies quickly. This makes inferring the length scale of the Gaussian process very difficult. We mitigated this issue by

developing an approach that uses the inferred intensity function from the PWC prior to inform the length scale of the Gaussian process.

The final issue we encountered was that sampling from the conditional distribution of the intensity function can be very challenging for the GP prior. In particular, the intensity function took artificially large values before the first spike time and after the last spike. To alleviate this problem we developed a new sampling approach to be used in tandem with the under-relaxed method to reduce the intensity in these problematic regions.

In Chapter 4, we expanded the model to include the refractory period as a parameter. We found that it was computationally difficult to decouple the refractory period from the intensity function and ISI parameter. This was due to the models without the refractory period being able to have near zero spiking rate in the region of the refractory period. Therefore, with the exception of the inhomogeneous Exponential distribution, it was not possible to extend the inhomogeneous distributions to include the refractory period explicitly. However, even with this setback simulated Ca^{2+} spike sequences still implicitly contain the refractory period.

In Chapter 5, we turned our attention to real data sets. We analysed Ca^{2+} spike sequences obtained from HEK293 cells and astrocytes. We first considered Ca^{2+} spike sequences of HEK293 cells and astrocytes challenged with a constant stimulus. For both cell types our model captured the heterogeneity of the Ca^{2+} response. In particular, the intensity function demonstrates two time scales in the Ca^{2+} response, where initially there is a burst of Ca^{2+} spikes at the onset of the stimulus and more regular Ca^{2+} spiking further into the experiment. We also found distinct differences in the Ca^{2+} response between the two cell types where astrocytes often have an extended ISI after the first Ca^{2+} spike, whereas HEK293 cells do not. Furthermore, we found on average astrocytes challenged with $100\mu\text{M}$ glutamate express fewer Ca^{2+} spikes than those challenged with $10\mu\text{M}$ glutamate but the Ca^{2+} spikes occur more regularly. In comparison, HEK293 cells exhibit a larger number of Ca^{2+} spikes when

the stimulus strength was increased — from $10\mu\text{M}$ to $30\mu\text{M}$ carbachol.

We then analysed Ca^{2+} spike sequences obtained from HEK293 cells challenged with carbachol delivered via time-dependent protocols. In particular, step-changes, waves and pulses of carbachol. HEK293 cells challenged with a step-change in stimulus are comparable with joining together two constant stimulus experiments. This was captured in our model by the intensity function which closely mirrored joining two intensity function from constant stimulation experiments. For cells challenged with waves or pulses of stimulus we found that the inferred intensity function followed a similar shape as the applied stimulation protocol.

Therefore, for all stimulation protocols we found that our model was able to capture the heterogeneity in the Ca^{2+} . One advantage of our framework is that given model parameters we can cheaply simulate surrogate spike sequences with similar properties to the Ca^{2+} spike sequences used to fit the model. Since our approach only uses Ca^{2+} spike times, we can create realistic spike sequences for HEK293 cells and astrocytes challenged with a variety of stimulation profiles without building individual mechanistic models for each situation. This allows us to cheaply generate realistic spike sequences for several cell types which can be used in signalling cascades to gain insight into the Ca^{2+} signalling toolbox.

For each dataset of Ca^{2+} spike sequences we fit our model for a selection of inhomogeneous ISI distributions, namely the Exponential with or without the refractory period, Gamma, inverse Gaussian, log-normal and Weibull. For each we fit the model with three different choices for the prior of the intensity function: constant, PWC and GP. For all datasets considered we found that the constant prior for the intensity function was too restrictive and cannot capture the underlying Ca^{2+} response. Similarly, we found that the inhomogeneous Exponential ISI distribution with or without the refractory period cannot accurately describe the Ca^{2+} spiking of any cell type or stimulation protocol. We often found that the models with a PWC prior for the intensity function leads to a similar or better performance than the GP prior. This is because both priors allow intensity functions that have similar shapes. The PWC prior occasionally outperforms the GP prior due to the discrete jumps in intensity it

allows, which allows sharp changes in the Ca^{2+} spiking rate to be captured in the intensity function.

In Tilunaite et al.'s work [16] they found that the inhomogeneous Gamma ISI distribution outperformed the inhomogeneous Exponential and inverse Gaussian ISI distributions. In comparison, for all the cell types and stimulation protocols considered we found that the best ISI distribution is between the inhomogeneous Gamma ISI distribution, the inhomogeneous inverse Gaussian ISI distribution or the inhomogeneous Weibull ISI distribution, where the inverse Gaussian ISI distribution has been reparameterised. Therefore, by extending the modelling framework we have found two more ISI distributions along with the Gamma ISI distribution that are consistent with Ca^{2+} spikes. Our analysis further confirmed that the inhomogeneous Exponential ISI model does not capture the Ca^{2+} response even with the addition of the refractory period as a model parameter.

As expressed by Tilunaite et al. the inhomogeneous Gamma ISI distribution was an appealing candidate for Ca^{2+} spikes because the Gamma probability distribution represents a combination of events occurring for the first time, such as Ca^{2+} puffs leading to a Ca^{2+} spike. We have shown that the inhomogeneous Gamma ISI distribution performs well and this interpretation is still valid. However, we have also found that the inhomogeneous Weibull and inverse Gaussian perform similarly. This leads to other potential interpretations that could drive Ca^{2+} spikes. For example Ca^{2+} spikes have been described as first-passage events and the inverse Gaussian distribution has been used to model such problems.

From our analysis it appears that the Ca^{2+} response of cells challenged with a complex stimulation protocol is comparable to joining the Ca^{2+} response from simpler experiments. For example a single step-change experiment leads to a similar Ca^{2+} response to two constant stimulus experiments. Therefore, it might be possible to generate realistic spike sequences for complex stimulation protocols by using knowledge from simpler experiments. This would be particularly exciting to create surrogate spike sequences for stimulation protocols that are difficult to record experimentally. We give an illustration of how this may be achieved in principle in Chapter 5.

In Chapter 6, we investigated how to cluster intensity functions. In particular, we wanted to cluster them depending on their shape over time. We took this approach because the shape contains important information about the underlying dynamics controlling the Ca^{2+} response. We began by considering recent methods used by Tilunaite et al. [16] for intensity functions and Delaigle et al. [142] for functional data. We found that neither approach clustered the intensity functions by their shape. Therefore we developed a new approach that decomposes intensity functions into their features, utilising the Haar basis, which are used to perform clustering. The features of the intensity functions are visualised by matrices containing the shapes and the magnitudes of the features. The approach successfully allows intensity functions to be clustered via their shape. We applied our method to intensity functions inferred from HEK293 cells challenged with constant stimulus and a step-change in stimulus. By decomposing the intensity functions into their features we found that most intensity functions share a similar shape. For example in constant stimulus experiments most intensity functions begin with a large decrease before plateauing. As expected we found the shape remains similar across all cells irrespective of the strength of the Ca^{2+} response. This illustrates that the shape of the intensity function is alike across the same cell type and stimulation protocol. However, we do find some differences that allow for clustering. For example some cells experience a prolonged region of low intensity after the initial peak when the stimulus is applied before recovering later in the experiment to 'plateaued' level of intensity, whereas other cells converge to the plateau directly after the stimulus is applied. Therefore, by clustering we have found differences in the cellular response in HEK293 cells. It is intriguing whether the differences occur randomly or arise from difference in the Ca^{2+} signalling toolbox expressed in the same cell type. For example one justification could be that the applied stimulus depletes cellular Ca^{2+} resources for some of the cells, which leads to the extended period of low intensity.

Finally, in Chapter 7, we outlined the online applications we created to be used alongside our work. In particular, we created applications to simulate Ca^{2+} spike

sequences, threshold Ca^{2+} concentration data into Ca^{2+} spike sequences and cluster intensity functions. These applications allow for an easy-to-use interface for some of the approaches used in this thesis.

8.2 Future work

Within our framework we took a non-parametric approach to model the intensity function. In particular, we used a GP prior for the intensity function. With this prior, we implemented an under-relaxed algorithm to sample from the conditional distribution of the intensity function, as this updates the intensity function as a block. However, computationally more complex methods — such as Hamiltonian Monte Carlo or Stochastic Gradient MCMC — may be more efficient at sampling from the conditional distribution. Therefore, in future developing such methods within our framework would be advantageous.

In our model the intensity function is by definition positive, where regions of near zero intensity correspond to regions of near zero Ca^{2+} spiking rate. Often the Ca^{2+} response of cells contain regions with no spikes — such as before the stimulus is applied. Therefore, the inferred intensity functions often contain near zero regions. Within the GP prior to constrain the intensity function to be positive we consider its logarithm. However, on the logarithmic scale regions where the intensity function is close to zero gets stretched by the transformation. This may lead to difficulty in inferring the length scale, as a small change in intensity function may lead to large changes in its logarithm. Therefore, future work looking for alternatives to the logarithm may improve the implementation of the GP prior for the intensity function.

One difficulty faced when fitting inhomogeneous ISI distributions is the balance between single cell heterogeneity and informative parameter values. We have shown that models fitted to datasets of Ca^{2+} spike sequences describe the mean spiking behaviour which does not encapsulate the cell-to-cell variability shown across all Ca^{2+}

spike sequences. However, a single Ca^{2+} spike sequence only contains limited information and therefore parameter estimates can contain a large amount of uncertainty. This could be one of the reasons it is difficult to tease apart which ISI distribution best describes the Ca^{2+} spike sequences. In future, it may be beneficial to create a procedure to group similar Ca^{2+} spike sequences together to be fitted simultaneously. It would be interesting whether such an approach would lead to parameter estimates with less uncertainty whilst capturing the cell-to-cell variability.

As illustrated in Chapter 5, our approach has the potential to generate realistic surrogate Ca^{2+} spike sequences from experimentally difficult stimulation protocols. A starting point to achieving this goal is to apply the methods provided to a wide range of Ca^{2+} spike sequences obtained from a variety of cell types and stimulation protocols, to explore if it is possible to classify cell types or stimulation protocols by their representative model parameters.

In Chapter 6 clustering is performed by examining the feature and magnitude matrices. This can be viewed as a positive as it forces the practitioner to consider how to cluster rather than trusting an automatic clustering scheme, which can lead to clusters that are difficult to understand. However, with large amounts of data it may preferred to use an automatic clustering mechanism. Therefore, in future work it would be beneficial to create such a tool for this approach, and investigate how automatic clusters compare to those chosen by the practitioner.

Throughout our goal has been to develop a statistical framework to analyse single cell Ca^{2+} spike sequences. We have shown that our models can accurately capture the Ca^{2+} spiking behaviour of astrocytes and HEK293 cells challenged with a variety of stimulation protocols. As an input our modelling framework only require Ca^{2+} spike sequences. Therefore, our framework can also be applied to any data that corresponds to a sequence of events over time. In particular, our methods could also be used to explore other aspects of Ca^{2+} signalling. For example the Ca^{2+} oscillations

of individual Ca^{2+} channels or Ca^{2+} oscillations in tissue could be used as an input. It would be interesting to apply our methods to such data and explore patterns that occur across different biological levels. So far, we have only considered Ca^{2+} oscillations from single cells in isolation. Subsequent work could investigate how single cells interact in tissue. For example are clustered cells equally distributed throughout the tissue or do we find regions of the tissue where cells respond similarly? Moreover, it would be exciting to find out whether intensity functions can track a stimulus's path through the tissue. For example this could be represented by the first peak in the intensity function, which varies across the population of cells. Finally, the methods herein are not restricted only to Ca^{2+} oscillations, it would be intriguing to see in which other fields such approaches could be used.

Appendices

APPENDIX A

Non-identifiable distributions

In this appendix we show the non-identifiability of the 3 parameter inhomogeneous inverse Gaussian, log-normal and Weibull ISI distributions.

• **Inverse Gaussian:** Consider the inhomogeneous inverse Gaussian distribution with parameters $(ax, a\lambda, a\mu)$ with $a > 0$ a constant below we show this is identical to the inverse Gaussian with parameters (x, λ, μ) .

$$\begin{aligned} p(t, s|ax, a\lambda, a\mu) &= ax(t) \left(\frac{a\lambda}{2\pi (aX(s, t))^3} \right)^{0.5} \exp \left\{ \frac{a\lambda (aX(s, t) - a\mu)^2}{2(a\mu)^2 aX(s, t)} \right\}, \\ &= ax(t) \left(\frac{\lambda}{a^2 2\pi X(s, t)} \right)^{0.5} \exp \left\{ \frac{a^3 \lambda (X(s, t) - \mu)^2}{2a^3 \mu^2 X(s, t)} \right\}, \\ &= x(t) \left(\frac{\lambda}{2\pi X(s, t)^3} \right)^{0.5} \exp \left\{ \frac{\lambda (X(s, t) - \mu)^2}{2\mu^2 X(s, t)} \right\}, \\ &= p(t, s|x, \lambda, \mu). \end{aligned}$$

• **Log-normal:** Consider the inhomogeneous log-normal distribution with parameters $(ax, \mu + \log a, \sigma)$ with $a > 0$ a constant below we show this is identical to

the log-normal with parameters (x, μ, σ) .

$$\begin{aligned}
 p(t, s|ax, \mu + \log a, \sigma) &= \frac{ax(t)}{aX(s, t)\sigma\sqrt{2\pi}} \exp \left\{ -\frac{(\log aX(s, t) - (\mu + \log a))^2}{2\sigma^2} \right\}, \\
 &= \frac{x(t)}{X(s, t)\sigma\sqrt{2\pi}} \exp \left\{ -\frac{(\log X(s, t) - \mu + \log a - \log a)^2}{2\sigma^2} \right\}, \\
 &= \frac{x(t)}{X(s, t)\sigma\sqrt{2\pi}} \exp \left\{ -\frac{(\log X(s, t) - \mu)^2}{2\sigma^2} \right\}, \\
 &= p(t, s|x, \mu, \sigma).
 \end{aligned}$$

• **Weibull:** Consider the inhomogeneous Weibull distribution with parameters $(ax, k, a\lambda)$ with $a > 0$ a constant below we show this is identical to the Weibull with parameters (x, k, λ) .

$$\begin{aligned}
 p(t, s|ax, k, a\lambda) &= \frac{ax(t)k}{a\lambda} \left(\frac{aX(s, t)}{a\lambda} \right)^{k-1} \exp \left\{ -\left(\frac{aX(s, t)}{a\lambda} \right)^k \right\}, \\
 &= \frac{x(t)k}{\lambda} \left(\frac{X(s, t)}{\lambda} \right)^{k-1} \exp \left\{ -\left(\frac{X(s, t)}{\lambda} \right)^k \right\}, \\
 &= p(t, s|x, k, \lambda).
 \end{aligned}$$

Complementary figures for application to real data

3

B.1 Astrocytes challenged glutamate

In this section, we provide rescaled ISI occurrence plots for four Ca^{2+} spike sequences; two obtained from astrocytes challenged with $10\mu\text{M}$ glutamate and two from astrocytes challenged with $100\mu\text{M}$ glutamate. This is shown in Figure B.1, B.2, B.3 and B.4 for the constant prior of the intensity function only.

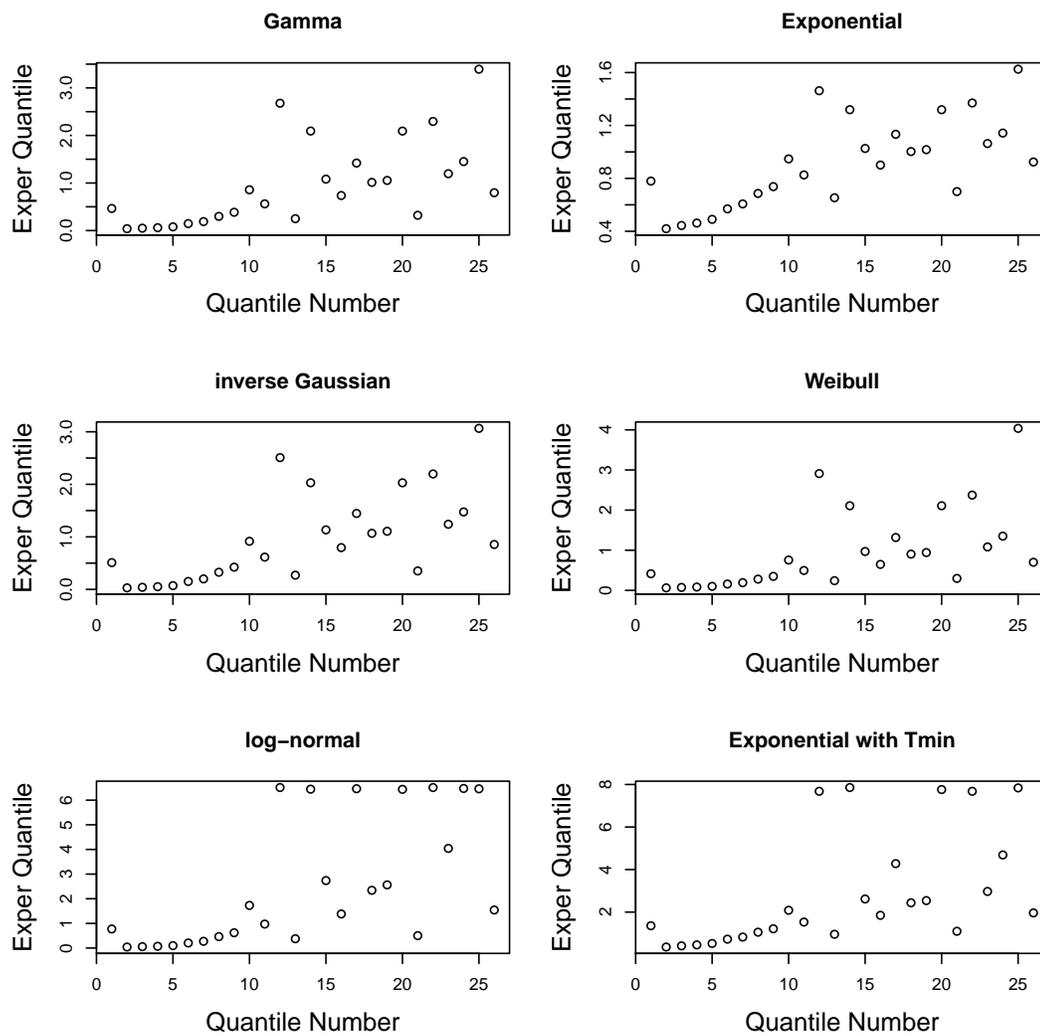


Figure B.1: Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from an astrocyte challenged with $10\mu\text{M}$ glutamate.

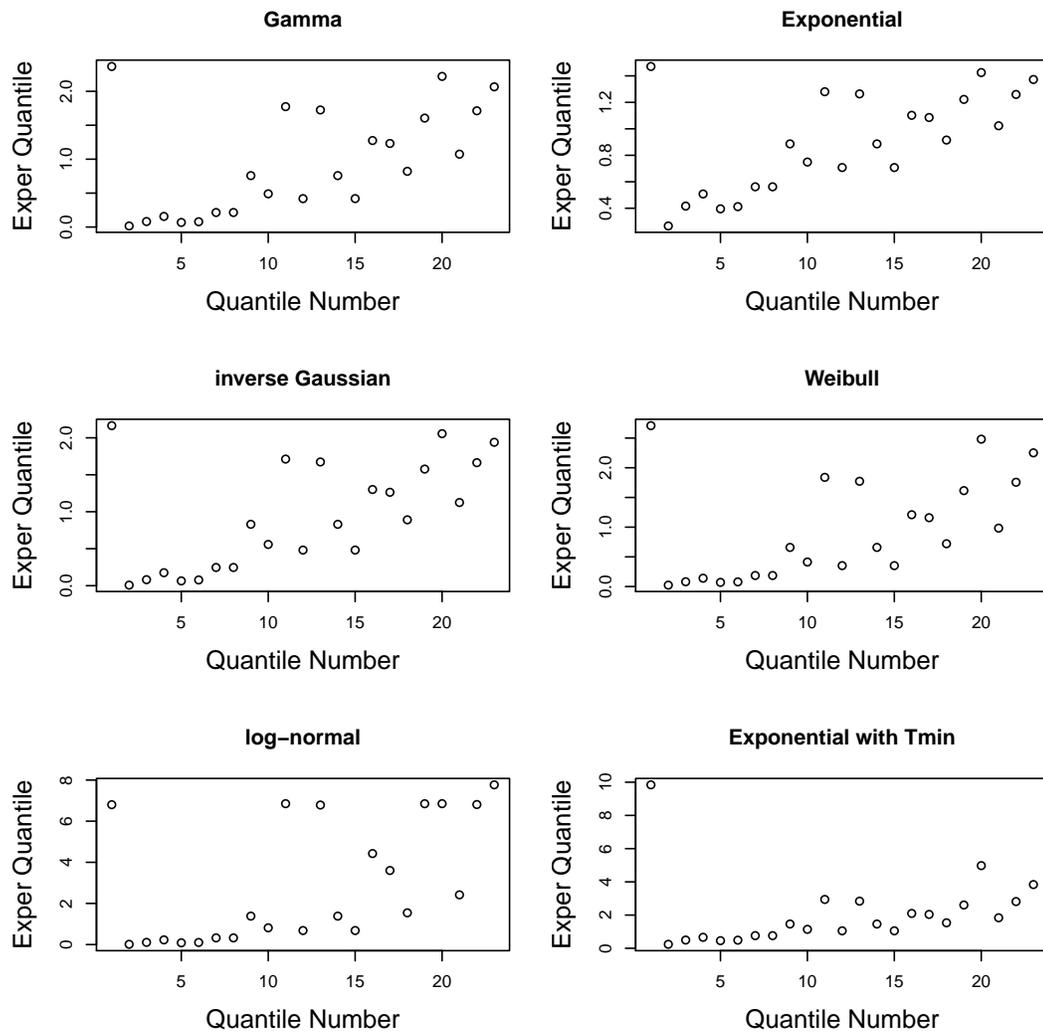


Figure B.2: Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from an astrocyte challenged with $10\mu\text{M}$ glutamate. The cell used differs from the one used in Figure B.1.

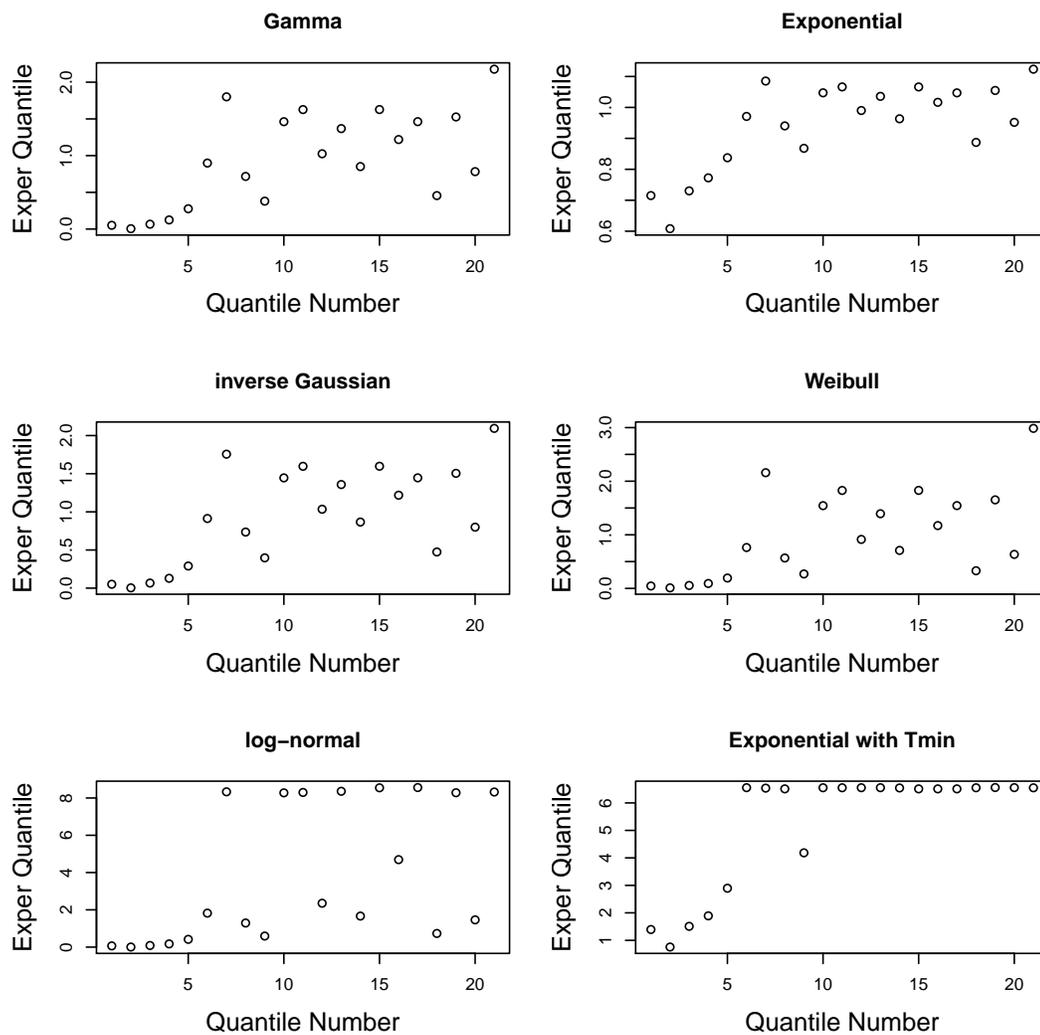


Figure B.3: Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from an astrocyte challenged with $100\mu\text{M}$ glutamate.

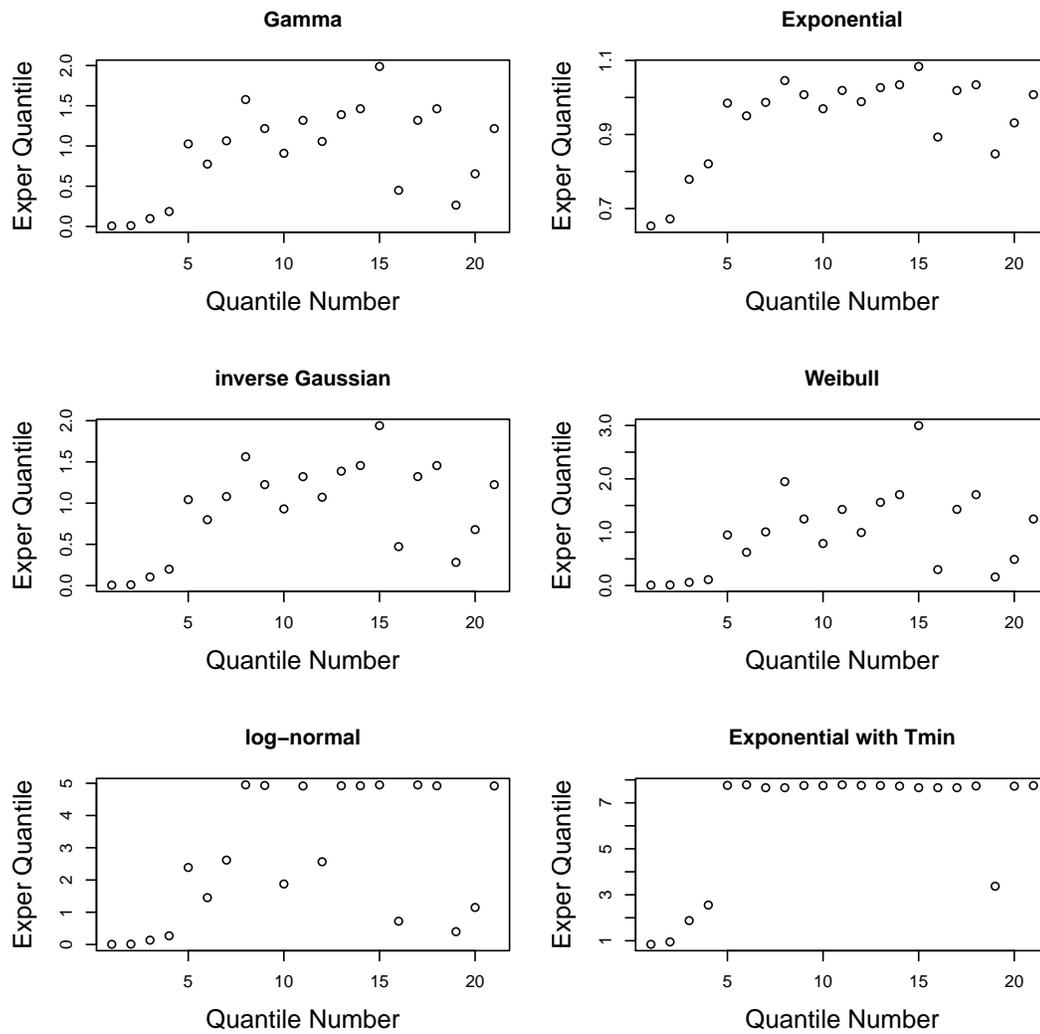


Figure B.4: Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from an astrocyte challenged with $100\mu\text{M}$ glutamate. The cell used differs from the one used in Figure B.3.

B.2 HEK293 cells challenged with constant carbachol

In figure B.5 we compare simulated spike sequences using the posterior of the inverse Gaussian ISI model with a PWC prior fitted to HEK293 cells challenged with 30 μ M carbachol, with the Ca²⁺ spikes used to fit the model.

The slopes of the Q-Q and K-S plot when the applied stimulus was 10 μ M carbachol is shown in Figure B.6.

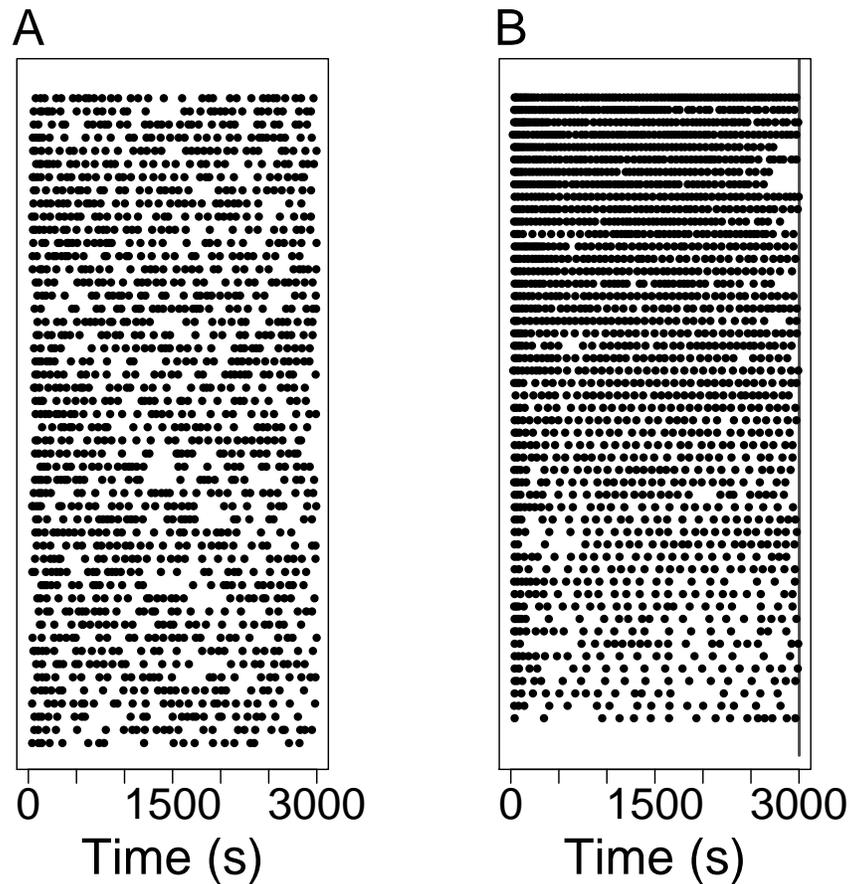


Figure B.5: (A) 50 simulated spike sequences from the posterior inverse Gaussian ISI distribution with a PWC prior fitted from the datasets where the applied stimulus $30\mu\text{M}$ carbachol. (B) The dataset of Ca^{2+} spike sequences used to fit the model used to simulate spikes in (A). In both the spike sequences are ordered such that the sequences with the most spikes occur at the top of the plots.

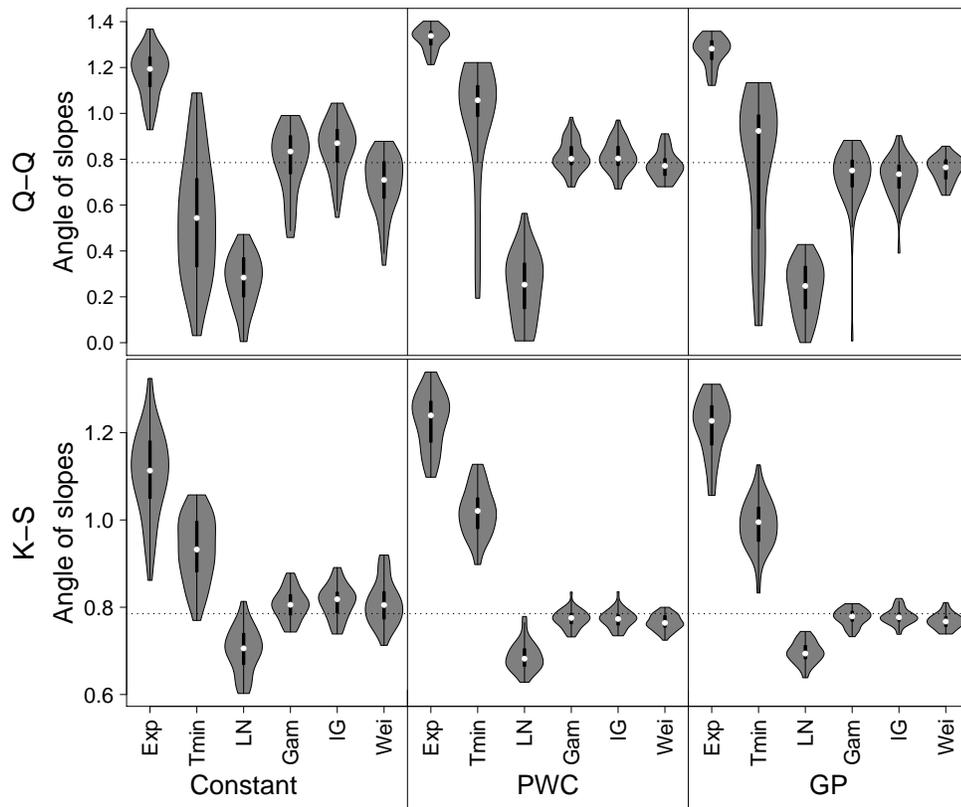


Figure B.6: Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with $10\mu\text{M}$ carbachol.

B.3 HEK293 cells challenged with a step change in carbachol

In this section, we first provide rescaled ISI occurrence plots for two Ca^{2+} spike sequences obtained from HEK293 cells challenged with a step change of carbachol of strength $10\mu\text{M} \rightarrow 50\mu\text{M}$. This is shown in Figure B.7 and Figure B.8 for the constant prior of the intensity function only.

We also produce the slopes of the Q-Q and K-S plots for HEK293 cells challenged with a step-change in stimulus, shown in Figures B.9, B.10, B.11 and B.12.

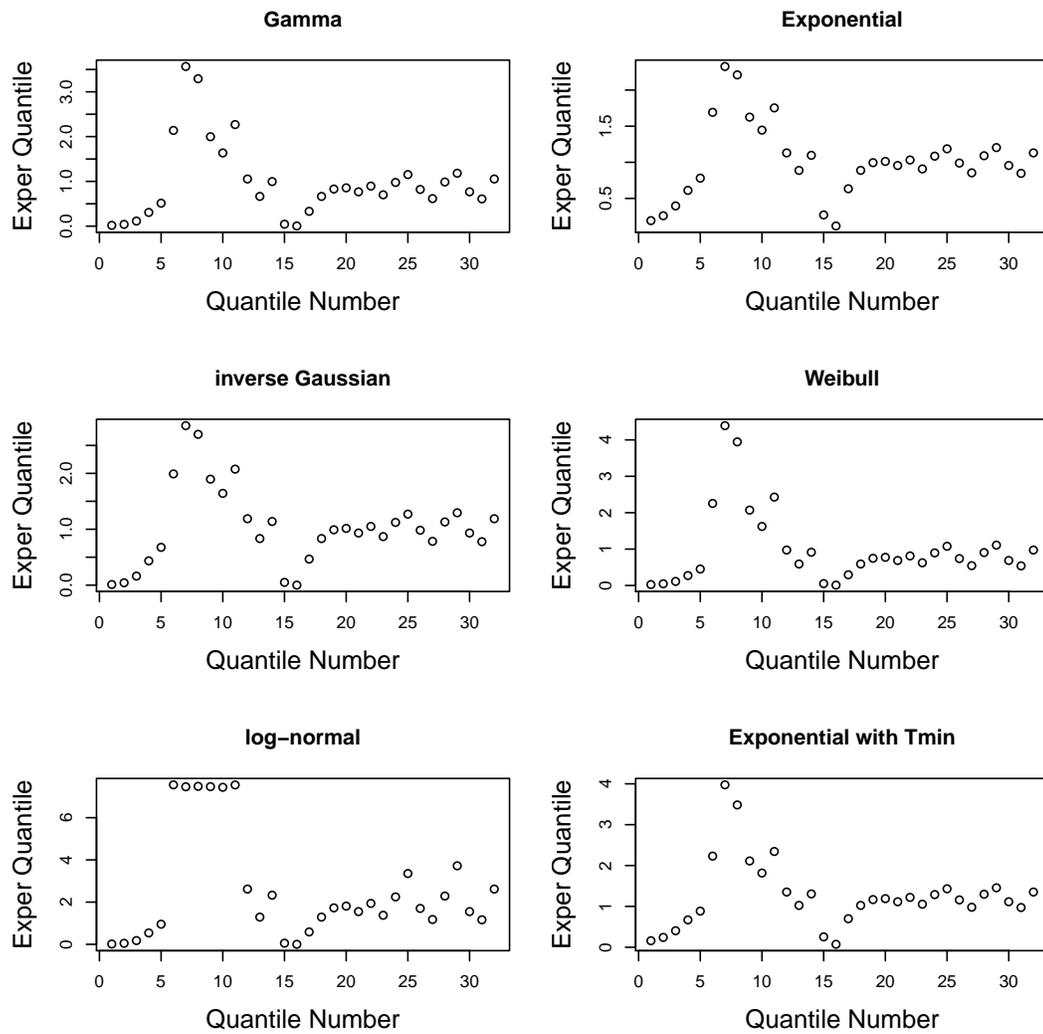


Figure B.7: Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from HEK293 cells challenged with a step change of carbachol of strength $10\mu\text{M} \rightarrow 50\mu\text{M}$.

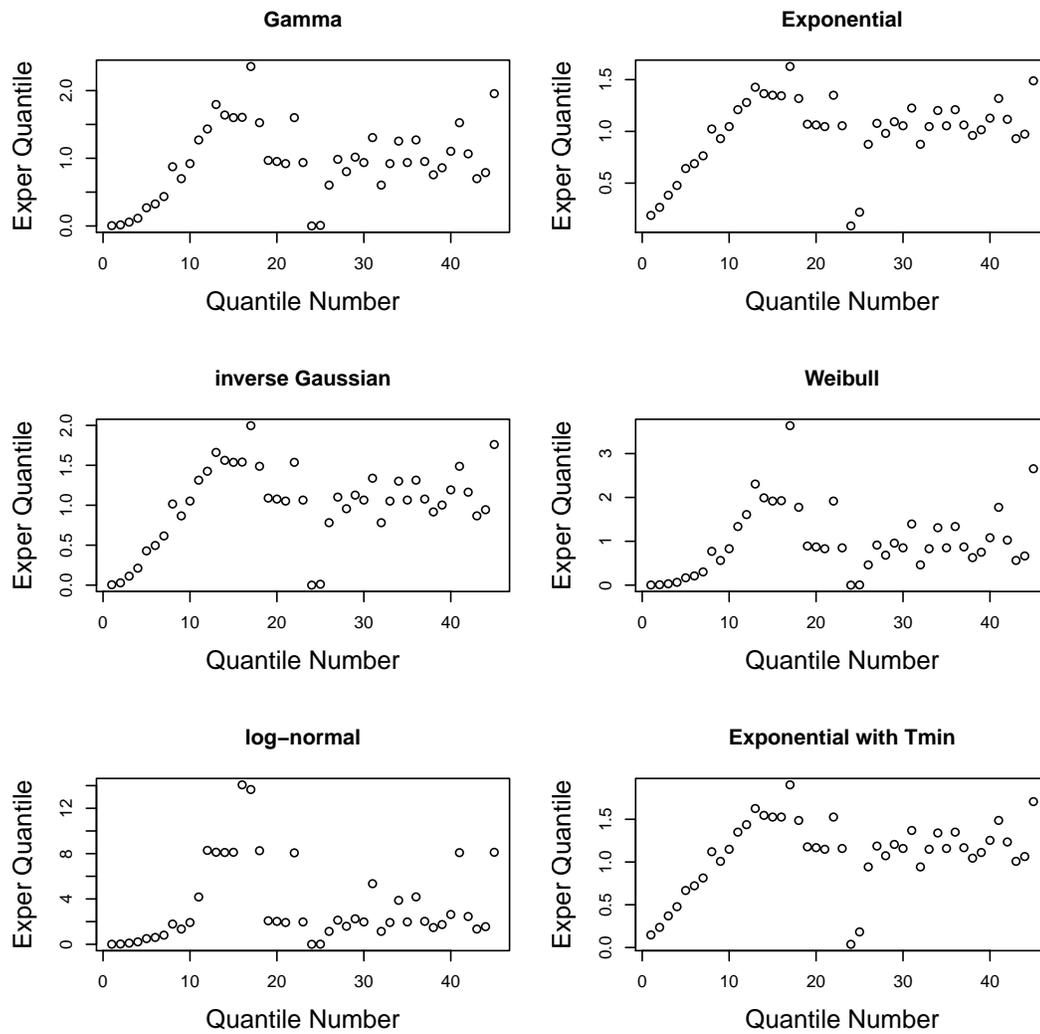


Figure B.8: Rescaled ISI occurrence plots for the ISI models fitted with a constant prior for the intensity function. The models were fitted to a single Ca^{2+} spike sequence obtained from HEK293 cells challenged with a step change of carbachol of strength $10\mu\text{M} \rightarrow 50\mu\text{M}$. The cell used was different from that used in Figure B.7.

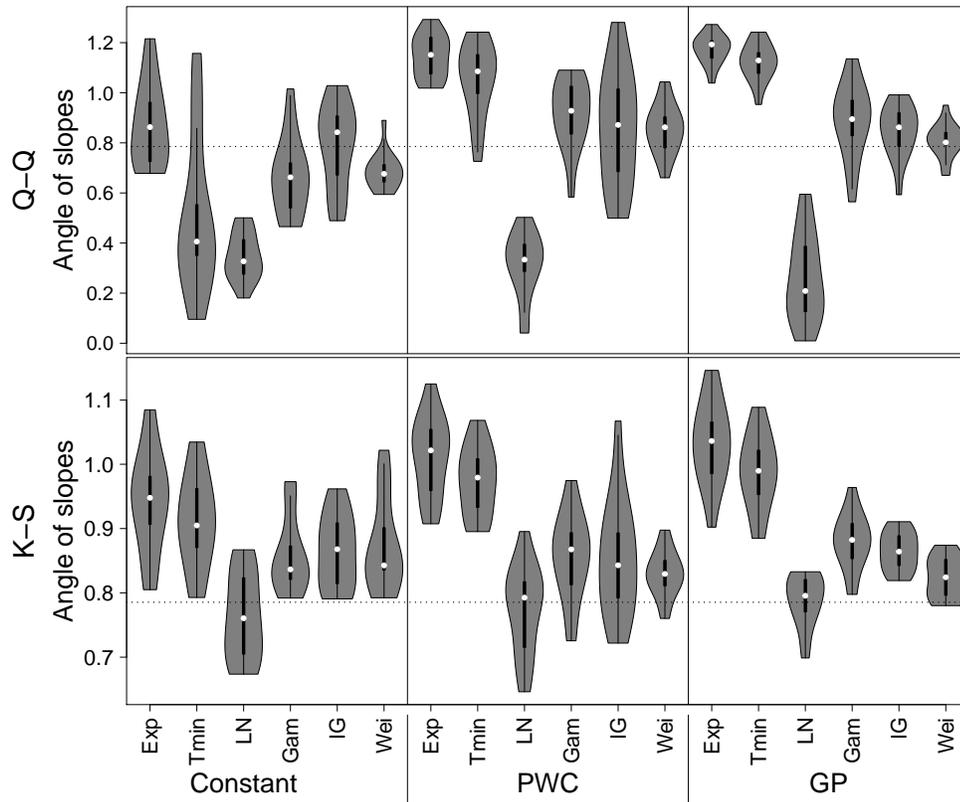


Figure B.9: Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with a step change of carbachol of strength $10\mu\text{M} \rightarrow 50\mu\text{M}$.

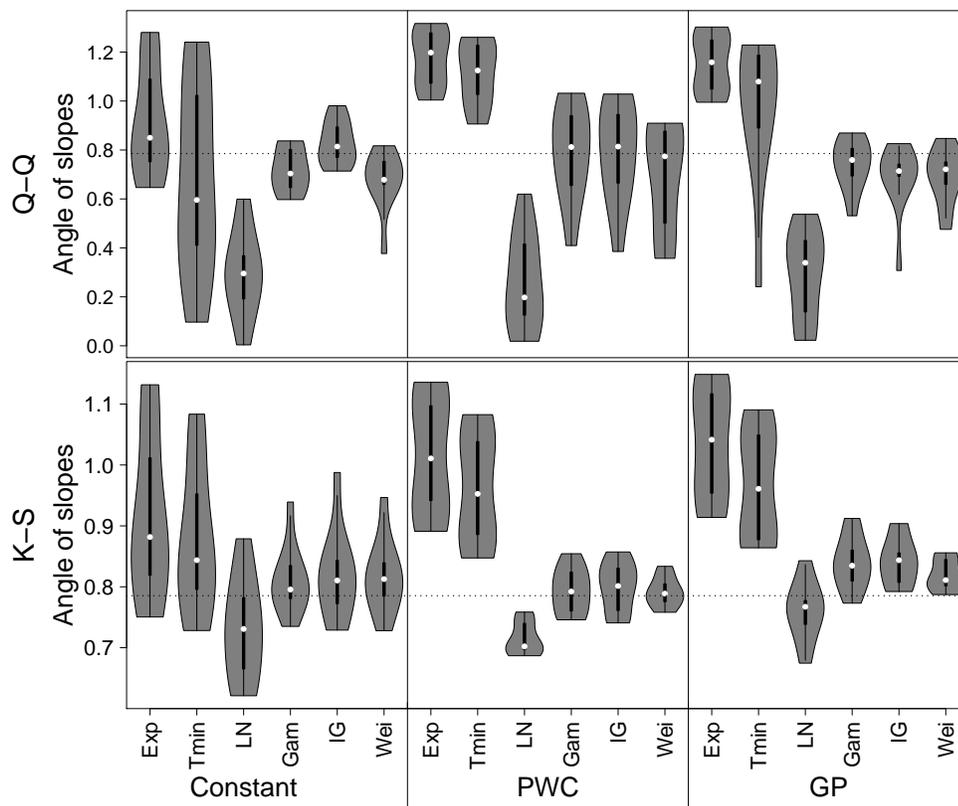


Figure B.10: Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with a step change of carbachol of strength $20\mu\text{M} \rightarrow 50\mu\text{M}$.

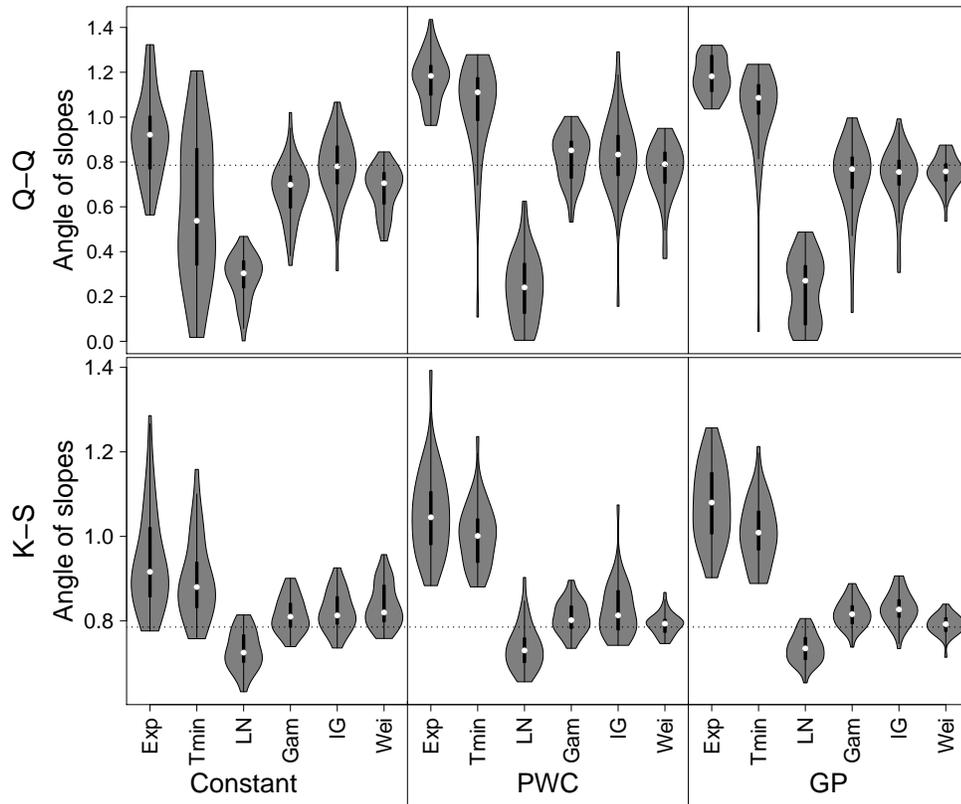


Figure B.11: Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with a step change of carbachol of strength $20\mu\text{M} \rightarrow 100\mu\text{M}$.

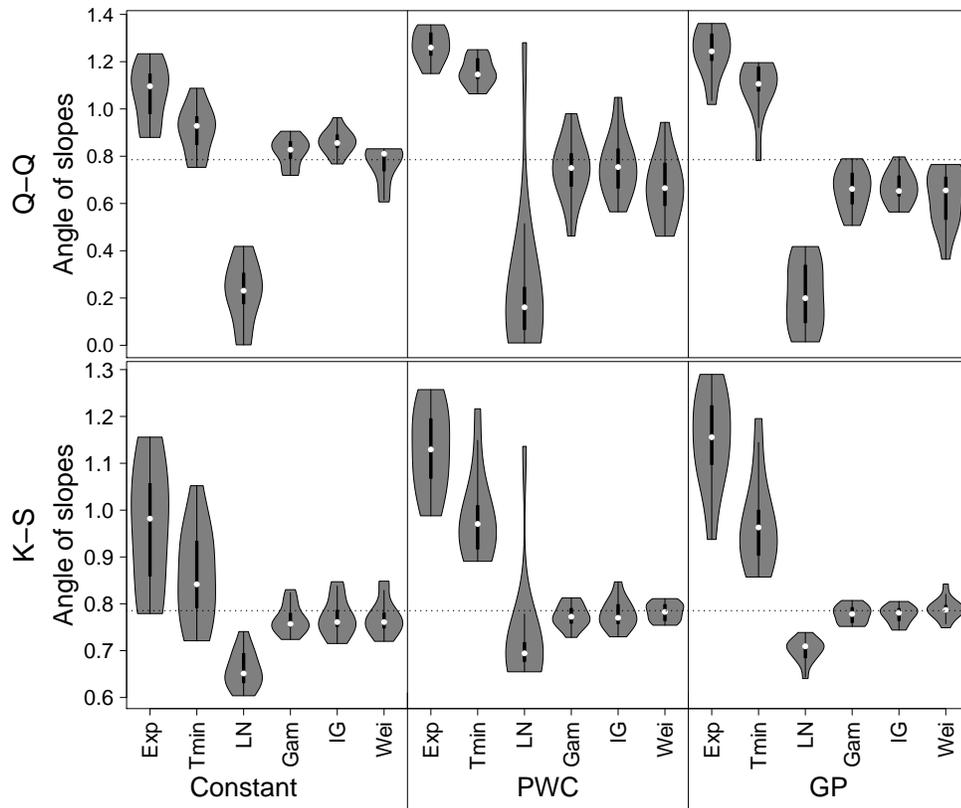


Figure B.12: Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with a step change of carbachol of strength $50\mu\text{M} \rightarrow 100\mu\text{M}$.

B.4 HEK293 cells challenged with three waves of carbachol

The slopes of the Q-Q and K-S plots for HEK293 cells challenged with three waves of carbachol.

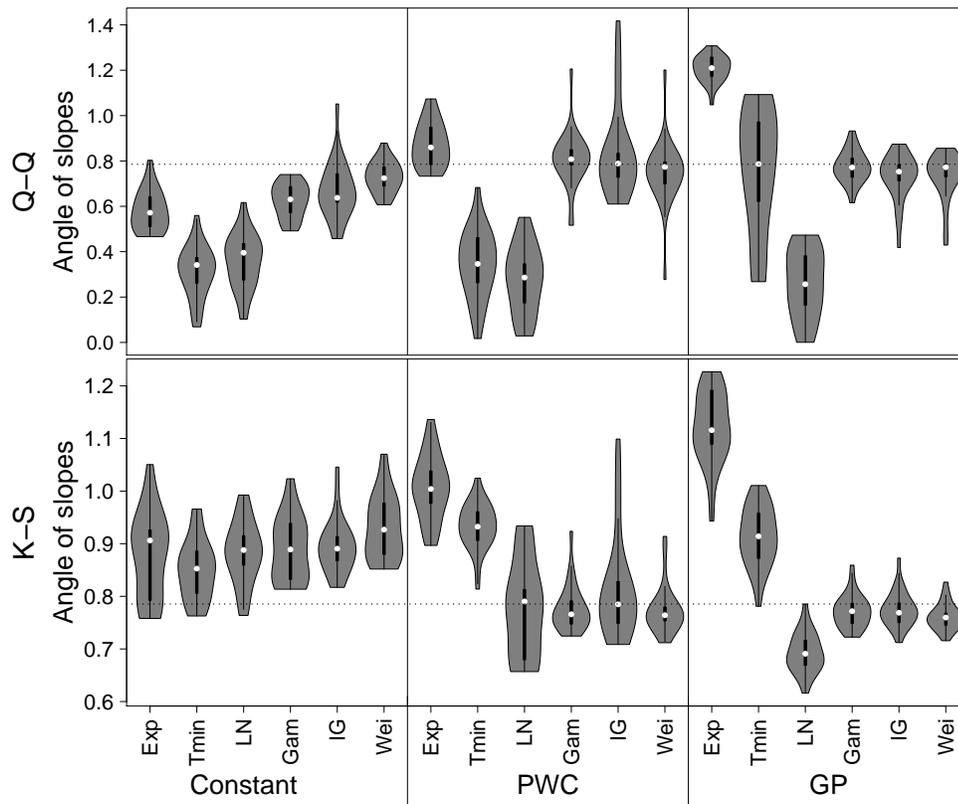


Figure B.13: Slope summaries of Q-Q and K-S plots of the inhomogeneous Exponential, Exponential with refractory period, log-normal, Gamma, inverse Gaussian and Weibull models for HEK293 cells challenged with three waves of carbachol.

Bibliography

- [1] M. D. Bootman and G. Bultynck, “Fundamentals of cellular calcium signaling: A primer,” *Cold Spring Harbor Perspectives in Biology*, vol. 12, p. a038802, aug 2019.
- [2] Y. Takeda, T. Shimayoshi, G. G. Holz, and A. Noma, “Modeling analysis of inositol 1,4,5-trisphosphate receptor-mediated Ca_2^+ mobilization under the control of glucagon-like peptide-1 in mouse pancreatic β -cells,” *American Journal of Physiology-Cell Physiology*, vol. 310, pp. C337–C347, mar 2016.
- [3] G. Fan, M. L. Baker, Z. Wang, M. R. Baker, P. A. Sinyagovskiy, W. Chiu, S. J. Ludtke, and I. I. Serysheva, “Gating machinery of InsP_3R channels revealed by electron cryomicroscopy,” *Nature*, vol. 527, pp. 336–341, oct 2015.
- [4] J. Parkash and K. Asotra, “Calcium wave signaling in cancer cells,” *Life Sciences*, vol. 87, pp. 587–595, nov 2010.
- [5] Y. Zhou, M. A. David, X. Chen, L. Q. Wan, R. L. Duncan, L. Wang, and X. L. Lu, “Effects of osmolarity on the spontaneous calcium signaling of in situ juvenile and adult articular chondrocytes,” *Annals of Biomedical Engineering*, vol. 44, pp. 1138–1147, jul 2015.

- [6] K. Thurley, S. C. Tovey, G. Moenke, V. L. Prince, A. Meena, A. P. Thomas, A. Skupin, C. W. Taylor, and M. Falcke, “Reliable encoding of stimulus intensities within random sequences of intracellular Ca^{2+} spikes,” *Science Signaling*, vol. 7, jun 2014.
- [7] E. Tosti, “Calcium ion currents mediating oocyte maturation events,” *Reproductive Biology and Endocrinology*, vol. 4, may 2006.
- [8] G. van Haasteren, S. Li, M. Muda, S. Susini, and W. Schlegel, “Calcium signalling and gene expression,” *Journal of Receptors and Signal Transduction*, vol. 19, pp. 481–492, jan 1999.
- [9] L. Santella, D. Lim, and F. Moccia, “Calcium and fertilization: the beginning of life,” *Trends in Biochemical Sciences*, vol. 29, pp. 400–408, Aug 2004.
- [10] M. Whitaker, “Calcium at fertilization and in early development,” *Physiological Reviews*, vol. 86, pp. 25–88, Jan 2006.
- [11] P. Denninger, A. Bleckmann, A. Lausser, F. Vogler, T. Ott, D. W. Ehrhardt, W. B. Frommer, S. Sprunck, T. Dresselhaus, and G. Grossmann, “Male–female communication triggers calcium signatures during fertilization in arabidopsis,” *Nature Communications*, vol. 5, Aug 2014.
- [12] M. P. Mattson and S. L. Chan, “Calcium orchestrates apoptosis,” *Nature Cell Biology*, vol. 5, pp. 1041–1043, dec 2003.
- [13] S. Orrenius, B. Zhivotovsky, and P. Nicotera, “Regulation of cell death: the calcium–apoptosis link,” *Nature Reviews Molecular Cell Biology*, vol. 4, pp. 552–565, jul 2003.
- [14] M. J. Berridge, “The inositol trisphosphate/calcium signaling pathway in health and disease,” *Physiological Reviews*, vol. 96, pp. 1261–1296, oct 2016.
- [15] M. J. Berridge, P. Lipp, and M. D. Bootman, “The versatility and universality of calcium signalling,” *Nature Reviews Molecular Cell Biology*, vol. 1, pp. 11–21, oct 2000.

- [16] A. Tilūnaitė, W. Croft, N. Russell, T. C. Bellamy, and R. Thul, “A bayesian approach to modelling heterogeneous calcium responses in cell populations,” *PLOS Computational Biology*, vol. 13, p. e1005794, oct 2017.
- [17] S. Ringer, “A further contribution regarding the influence of the different constituents of the blood on the contraction of the heart,” *Journal of Physiology*, vol. 4, no. 1, pp. 29–42, 1883.
- [18] S. Ringer, “Concerning the influence exerted by each of the constituents of the blood on the contraction of the ventricle,” *Journal of Physiology*, vol. 3, no. 5-6, pp. 380–393, 1882.
- [19] A. K. Campbell, “Intracellular calcium, its universal role as regulator,” *Wiley*, pp. 13–17, 1983.
- [20] Institute of Medicine, *Dietary Reference Intakes for Calcium and Vitamin D*. Washington, DC: The National Academies Press, 2011.
- [21] Z. Song, Y. Wang, F. Zhang, F. Yao, and C. Sun, “Calcium signaling pathways: Key pathways in the regulation of obesity,” *International Journal of Molecular Sciences*, vol. 20, p. 2768, jun 2019.
- [22] G. W. D. Young and J. Keizer, “A single-pool inositol 1,4,5-trisphosphate-receptor-based model for agonist-stimulated oscillations in Ca^{2+} concentration.,” *Proceedings of the National Academy of Sciences*, vol. 89, pp. 9895–9899, oct 1992.
- [23] J. Sneyd and M. Falcke, “Models of the inositol trisphosphate receptor,” *Progress in Biophysics and Molecular Biology*, vol. 89, no. 3, pp. 207–245, 2005.
- [24] G. Dupont, M. Falcke, V. Kirk, and J. Sneyd, *Models of Calcium Signalling*. Springer International Publishing, 2016.
- [25] S. Swillens, G. Dupont, L. Combettes, and P. Champeil, “From calcium blips to calcium puffs: Theoretical analysis of the requirements for interchannel

- communication,” *PNAS; Proceedings of the National Academy of Sciences*, vol. 96, no. 24, pp. 13750–13755, 1999.
- [26] D. L. Prole and C. W. Taylor, “Structure and function of IP₃ receptors,” *Cold Spring Harbor Perspectives in Biology*, vol. 11, p. a035063, feb 2019.
- [27] J. Sneyd and R. Bertram, *Tutorials in Mathematical Biosciences II Mathematical modelling of calcium dynamics and signal transduction*, vol. 1867. Springer Science and Business Media, 2005.
- [28] H. Lodish, A. Berk, P. Matsudaira, and C. A. Kaiser, *Molecular Cell Biology*. W.H. Freeman, 2004.
- [29] M. J. Berridge, “Calcium signalling, a spatiotemporal phenomenon,” in *Calcium* (J. Krebs and M. Michalak, eds.), vol. 41 of *New Comprehensive Biochemistry*, pp. 485–502, Elsevier, 2007.
- [30] B. Schwaller, “Cytosolic Ca²⁺ buffers,” *Cold Spring Harbor Perspectives in Biology*, vol. 2, pp. a004051–a004051, oct 2010.
- [31] M. J. Berridge, M. D. Bootman, and H. L. Roderick, “Calcium signalling: dynamics, homeostasis and remodelling,” *Nature Reviews Molecular Cell Biology*, vol. 4, pp. 517–529, jul 2003.
- [32] C. Fewtrell, “Ca²⁺ oscillations in non-excitabile cells,” *Annual Review of Physiology*, vol. 55, pp. 427–454, oct 1993.
- [33] J. A. Gilabert, “Cytoplasmic calcium buffering,” in *Advances in Experimental Medicine and Biology*, pp. 483–498, Springer Netherlands, 2012.
- [34] N. H. Evans, M. R. McAinsh, and A. M. Hetherington, “Calcium oscillations in higher plants,” *Current Opinion in Plant Biology*, vol. 4, pp. 415–420, oct 2001.
- [35] M. J. Boulware and J. S. Marchant, “Timing in cellular Ca²⁺ signaling,” *Current Biology*, vol. 18, pp. R769–R776, sep 2008.

- [36] U. Kummer, L. Olsen, C. Dixon, A. Green, E. Bornberg-Bauer, and G. Baier, “Switching from simple to complex oscillations in calcium signaling,” *Biophysical Journal*, vol. 79, no. 3, pp. 1188–1195, 2000.
- [37] W. Croft, K. Reusch, A. Tilunait, N. A. Russell, R. Thul, and T. C. Bellamy, “Probabilistic encoding of stimulus strength in astrocyte global calcium signals,” *Glia*, vol. 64, pp. 537–552, dec 2015.
- [38] C. C. Petersen, E. C. Toescu, and O. H. Petersen, “Different patterns of receptor-activated cytoplasmic Ca^{2+} oscillations in single pancreatic acinar cells: dependence on receptor type, agonist concentration and intracellular Ca^{2+} buffering,” *The EMBO Journal*, vol. 10, pp. 527–533, mar 1991.
- [39] R. E. Dolmetsch, K. Xu, and R. S. Lewis, “Calcium oscillations increase the efficiency and specificity of gene expression,” *Nature*, vol. 392, pp. 933–936, apr 1998.
- [40] A. Politi, L. D. Gaspers, A. P. Thomas, and T. Höfer, “Models of IP_3 and Ca^{2+} oscillations: Frequency encoding and identification of underlying feedbacks,” *Biophysical Journal*, vol. 90, pp. 3120–3133, may 2006.
- [41] P. Uhlén and N. Fritz, “Biochemistry of calcium oscillations,” *Biochemical and Biophysical Research Communications*, vol. 396, pp. 28–32, may 2010.
- [42] M. J. Berridge, M. D. Bootman, and P. Lipp, “Calcium - a life and death signal,” *Nature*, vol. 395, pp. 645–648, oct 1998.
- [43] E. Smedler and P. Uhlén, “Frequency decoding of calcium oscillations,” *Biochimica et Biophysica Acta (BBA) - General Subjects*, vol. 1840, no. 3, pp. 964–969, 2014.
- [44] M. J. Berridge and A. Galione, “Cytosolic calcium oscillators,” *The FASEB Journal*, vol. 2, pp. 3074–3082, dec 1988.

- [45] T. A. Rooney, E. J. Sass, and A. P. Thomas, “Characterization of cytosolic calcium oscillations induced by phenylephrine and vasopressin in single fura-2-loaded hepatocytes,” *Journal of Biological Chemistry*, vol. 264, no. 29, pp. 17131–17141, 1989.
- [46] A. H. Cornell-Bell, S. M. Finkbeiner, M. S. Cooper, and S. J. Smith, “Glutamate induces calcium waves in cultured astrocytes: Long-range glial signaling,” *Science*, vol. 247, pp. 470–473, jan 1990.
- [47] A. Z. Larsen, L. F. Olsen, and U. Kummer, “On the encoding and decoding of calcium signals in hepatocytes,” *Biophysical Chemistry*, vol. 107, pp. 83–99, jan 2004.
- [48] M. J. Berridge, “The AM and FM of calcium signalling,” *Nature*, vol. 386, pp. 759–760, apr 1997.
- [49] H. Qi, X. Li, Z. Jin, T. Simmen, and J. Shuai, “The oscillation amplitude, not the frequency of cytosolic calcium, regulates apoptosis induction,” *iScience*, vol. 23, p. 101671, nov 2020.
- [50] M. T. Nelson, H. Cheng, M. Rubart, L. F. Santana, A. D. Bonev, H. J. Knot, and W. J. Lederer, “Relaxation of arterial smooth muscle by calcium sparks,” *Science*, vol. 270, pp. 633–637, oct 1995.
- [51] V. A. Porter, A. D. Bonev, H. J. Knot, T. J. Heppner, A. S. Stevenson, T. Kleppisch, W. J. Lederer, and M. T. Nelson, “Frequency modulation of Ca^{2+} sparks is involved in regulation of arterial diameter by cyclic nucleotides,” *American Journal of Physiology-Cell Physiology*, vol. 274, pp. C1346–C1355, may 1998.
- [52] A. B. Parekh, “Decoding cytosolic Ca^{2+} oscillations,” *Trends in Biochemical Sciences*, vol. 36, pp. 78–87, feb 2011.
- [53] S. Song, J. Li, L. Zhu, L. Cai, Q. Xu, C. Ling, Y. Su, and Q. Hu, “Irregular Ca^{2+} oscillations regulate transcription via cumulative spike duration and spike

- amplitude,” *Journal of Biological Chemistry*, vol. 287, pp. 40246–40255, nov 2012.
- [54] M. D. Pittà, V. Volman, H. Levine, G. Pioggia, D. D. Rossi, and E. Ben-Jacob, “Coexistence of amplitude and frequency modulations in intracellular calcium dynamics,” *Physical Review E*, vol. 77, mar 2008.
- [55] J. Yao, A. Pilko, and R. Wollman, “Distinct cellular states determine calcium signaling response,” *Molecular Systems Biology*, vol. 12, p. 894, dec 2016.
- [56] K. Thurley and M. Falcke, “Derivation of Ca^{2+} signals from puff properties reveals that pathway function is robust against cell variability but sensitive for control,” *Proceedings of the National Academy of Sciences*, vol. 108, pp. 427–432, dec 2010.
- [57] T. Wada, K. ichi Hironaka, and S. Kuroda, “Cell-to-cell variability serves as information not noise,” *Current Opinion in Systems Biology*, vol. 27, p. 100339, 2021.
- [58] M. Falcke, “Reading the patterns in living cells —the physics of Ca^{2+} signaling,” *Advances in Physics*, vol. 53, no. 3, pp. 255–440, 2004.
- [59] G. Dupont, L. Combettes, G. S. Bird, and J. W. Putney, “Calcium oscillations,” *Cold Spring Harbor Perspectives in Biology*, vol. 3, pp. a004226–a004226, dec 2010.
- [60] R. Thul, T. C. Bellamy, H. L. Roderick, M. D. Bootman, and S. Coombes, “Calcium oscillations,” in *Advances in Experimental Medicine and Biology*, pp. 1–27, Springer New York, 2008.
- [61] A. Skupin, H. Kettenmann, U. Winkler, M. Wartenberg, H. Sauer, S. C. Tovey, C. W. Taylor, and M. Falcke, “How does intracellular Ca^{2+} oscillate: By chance or by the clock?,” *Biophysical Journal*, vol. 94, pp. 2404–2411, mar 2008.

- [62] J. Powell, M. Falcke, A. Skupin, T. C. Bellamy, T. Kypraios, and R. Thul, “A statistical view on calcium oscillations,” in *Advances in Experimental Medicine and Biology*, pp. 799–826, Springer International Publishing, oct 2019.
- [63] Y.-X. Li and J. Rinzel, “Equations for InsP_3 receptor-mediated $[\text{Ca}^{2+}]_i$ oscillations derived from a detailed kinetic model: A Hodgkin-Huxley like formalism,” *Journal of Theoretical Biology*, vol. 166, pp. 461–473, feb 1994.
- [64] A. Atri, J. Amundson, D. Clapham, and J. Sneyd, “A single-pool model for intracellular calcium oscillations and waves in the *Xenopus laevis* oocyte,” *Biophysical Journal*, vol. 65, pp. 1727–1739, oct 1993.
- [65] Y. Tang, J. Stephenson, and H. Othmer, “Simplification and analysis of models of calcium dynamics based on IP_3 -sensitive calcium channel kinetics,” *Biophysical Journal*, vol. 70, pp. 246–263, jan 1996.
- [66] N. Mikolajewicz, D. Smith, S. V. Komarova, and A. Khadra, “High-affinity $\text{p}2\text{y}2$ and low-affinity $\text{p}2\text{x}7$ receptor interaction modulates ATP-mediated calcium signaling in murine osteoblasts,” *PLOS Computational Biology*, vol. 17, p. e1008872, jun 2021.
- [67] M. Falcke, L. Tsimring, and H. Levine, “Stochastic spreading of intracellular Ca^{2+} release,” *Physical Review E*, vol. 62, pp. 2636–2643, aug 2000.
- [68] M. Falcke, “On the role of stochastic channel behavior in intracellular Ca^{2+} dynamics,” *Biophysical Journal*, vol. 84, pp. 42–56, jan 2003.
- [69] G. Ullah and P. Jung, “Modeling the statistics of elementary calcium release events,” *Biophysical Journal*, vol. 90, no. 10, pp. 3485–3495, 2006.
- [70] J. R. Groff and G. D. Smith, “Calcium-dependent inactivation and the dynamics of calcium puffs and sparks,” *Journal of Theoretical Biology*, vol. 253, pp. 483–499, aug 2008.

- [71] R. Thul and M. Falcke, “Building oscillations bottom up: Elemental time scales of intracellular calcium dynamics,” in *Analysis and Control of Complex Nonlinear Processes in Physics, Chemistry and Biology*, pp. 293–324, WORLD SCIENTIFIC, jan 2007.
- [72] A. Skupin and K. Thurley, “Calcium signaling: From single channels to pathways,” in *Advances in Experimental Medicine and Biology*, pp. 531–551, Springer Netherlands, 2012.
- [73] M. Perc, A. K. Green, C. J. Dixon, and M. Marhl, “Establishing the stochastic nature of intracellular calcium oscillations from experimental data,” *Biophysical Chemistry*, vol. 132, pp. 33–38, jan 2008.
- [74] J. Keizer and G. D. Smith, “Spark-to-wave transition: saltatory transmission of calcium waves in cardiac myocytes,” *Biophysical Chemistry*, vol. 72, pp. 87–100, may 1998.
- [75] A. Skupin and M. Falcke, “Statistical analysis of calcium oscillations,” *The European Physical Journal Special Topics*, vol. 187, pp. 231–240, sep 2010.
- [76] K. Thurley, A. Skupin, R. Thul, and M. Falcke, “Fundamental properties of Ca^{2+} signals,” *Biochimica et Biophysica Acta (BBA) - General Subjects*, vol. 1820, pp. 1185–1194, aug 2012.
- [77] S. Rüdiger, “Stochastic models of intracellular calcium signals,” *Physics Reports*, vol. 534, pp. 39–87, Jan. 2014.
- [78] P. Cao, G. Donovan, M. Falcke, and J. Sneyd, “A stochastic model of calcium puffs based on single-channel data,” *Biophysical Journal*, vol. 105, pp. 1133–1142, sep 2013.
- [79] M. R. Maurya and S. Subramaniam, “A kinetic model for calcium dynamics in RAW 264.7 cells: 1. mechanisms, parameters, and subpopulational variability,” *Biophysical Journal*, vol. 93, pp. 709–728, aug 2007.

- [80] A. Skupin and M. Falcke, “From puffs to global Ca^{2+} signals: How molecular properties shape global signals,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 19, p. 037111, sep 2009.
- [81] R. Y. Tsien, “New calcium indicators and buffers with high selectivity against magnesium and protons: design, synthesis, and properties of prototype structures,” *Biochemistry*, vol. 19, pp. 2396–2404, may 1980.
- [82] R. M. Paredes, J. C. Etzler, L. T. Watts, W. Zheng, and J. D. Lechleiter, “Chemical calcium indicators,” *Methods*, vol. 46, pp. 143–151, nov 2008.
- [83] J. Oh, C. Lee, and B.-K. Kaang, “Imaging and analysis of genetically encoded calcium indicators linking neural circuits and behaviors,” *The Korean Journal of Physiology & Pharmacology*, vol. 23, no. 4, p. 237, 2019.
- [84] M. D. Bootman, K. Rietdorf, T. Collins, S. Walker, and M. Sanderson, “ Ca^{2+} -sensitive fluorescent dyes and intracellular Ca^{2+} imaging,” *Cold Spring Harbor Protocols*, vol. 2013, p. pdb.top066050, feb 2013.
- [85] M. D. Bootman, K. Rietdorf, T. Collins, S. Walker, and M. Sanderson, “Converting fluorescence data into Ca^{2+} concentration,” *Cold Spring Harbor Protocols*, vol. 2013, p. pdb.prot072827, feb 2013.
- [86] F. L. Graham, W. C. Russell, J. Smiley, and R. Nairn, “Characteristics of a human cell line transformed by DNA from human adenovirus type 5,” *Journal of General Virology*, vol. 36, pp. 59–72, jul 1977.
- [87] Y. C. Lin, M. Boone, L. Meuris, I. Lemmens, N. V. Roy, A. Soete, J. Reumers, M. Moisse, S. Plaisance, R. Drmanac, J. Chen, F. Speleman, D. Lambrechts, Y. V. de Peer, J. Tavernier, and N. Callewaert, “Genome dynamics of the human embryonic kidney 293 lineage in response to cell biology manipulations,” *Nature Communications*, vol. 5, sep 2014.

- [88] P. Thomas and T. G. Smart, “HEK293 cell line: A vehicle for the expression of recombinant proteins,” *Journal of Pharmacological and Toxicological Methods*, vol. 51, pp. 187–200, may 2005.
- [89] J. A. Hubbard and D. K. Binder, “Chapter 1 - history of astrocytes,” in *Astrocytes and Epilepsy* (J. A. Hubbard and D. K. Binder, eds.), pp. 1–38, San Diego: Academic Press, 2016.
- [90] M. V. Sofroniew and H. V. Vinters, “Astrocytes: biology and pathology,” *Acta Neuropathologica*, vol. 119, pp. 7–35, dec 2009.
- [91] J. G. Emsley and J. D. Macklis, “Astroglial heterogeneity closely reflects the neuronal-defined anatomy of the adult murine CNS,” *Neuron Glia Biology*, vol. 2, pp. 175–186, jun 2006.
- [92] N. J. Abbott, L. Rönnbäck, and E. Hansson, “Astrocyte-endothelial interactions at the blood-brain barrier,” *Nature Reviews Neuroscience*, vol. 7, pp. 41–53, jan 2006.
- [93] D. L. Montgomery, “Astrocytes: Form, functions, and roles in disease,” *Veterinary Pathology*, vol. 31, pp. 145–167, mar 1994.
- [94] H. Phatnani and T. Maniatis, “Astrocytes in neurodegenerative disease,” *Cold Spring Harbor Perspectives in Biology*, vol. 7, apr 2015.
- [95] N. A. Russell, A. D. Johnstone, and M. Alexander, “Long-term imaging of fibroblasts in a microfluidic device,” in *2011 Functional Optical Imaging*, pp. 1–2, 2011.
- [96] D. J. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes. Vol. I. Probability and its Applications* (New York), New York: Springer-Verlag, second ed., 2003. Elementary theory and methods.
- [97] D. J. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes. Volume II: General Theory and Structure*. New York, NY: Springer New York, 2008.

- [98] W. Gerstner, W. M. Kistler, R. Naud, and L. Paninski, *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press, 2014.
- [99] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue, and E. N. Brown, “A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects,” *Journal of Neurophysiology*, vol. 93, pp. 1074–1089, feb 2005.
- [100] L. Paninski, J. Pillow, and J. Lewi, “Statistical models for neural encoding, decoding, and optimal stimulus design,” in *Progress in Brain Research*, pp. 493–507, Elsevier, 2007.
- [101] W. Gerstner and W. M. Kistler, *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge university press, 2002.
- [102] R. Jolivet, T. J. Lewis, and W. Gerstner, “Generalized integrate-and-fire models of neuronal activity approximate spike trains of a detailed model to a high degree of accuracy,” *Journal of Neurophysiology*, vol. 92, pp. 959–976, aug 2004.
- [103] K. Thurley, I. F. Smith, S. C. Tovey, C. W. Taylor, I. Parker, and M. Falcke, “Timescales of IP₃-evoked Ca²⁺ spikes emerge from Ca²⁺ puffs only at the cellular level,” *Biophysical Journal*, vol. 101, pp. 2638–2644, dec 2011.
- [104] D. R. Cox, *Renewal theory*. Methuen, 1962.
- [105] R. Barbieri, M. C. Quirk, L. M. Frank, M. A. Wilson, and E. N. Brown, “Construction and analysis of non-poisson stimulus-response models of neural spiking activity,” *Journal of Neuroscience Methods*, vol. 105, pp. 25–37, jan 2001.
- [106] J. Pillow, “Time-rescaling methods for the estimation and assessment of non-poisson neural encoding models,” in *Advances in Neural Information Processing Systems* (Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, eds.), vol. 22, Curran Associates, Inc., 2009.

- [107] E. N. Brown, R. Barbieri, V. Ventura, R. E. Kass, and L. M. Frank, “The time-rescaling theorem and its application to neural spike train data analysis,” *Neural Computation*, vol. 14, pp. 325–346, feb 2002.
- [108] L. Devroye, *Non-Uniform Random Variate Generation*. Springer-Verlag New York, 2013.
- [109] J. P. Cunningham, K. V. Shenoy, and M. Sahani, “Fast gaussian process methods for point process intensity estimation,” in *Proceedings of the 25th International Conference on Machine Learning*, pp. 192–199, Stanford University, 2008.
- [110] G. Casella and E. I. George, “Explaining the Gibbs Sampler,” *The American Statistician*, vol. 46, no. 3, pp. 167–174, 1992.
- [111] S. Chib and E. Greenberg, “Understanding the Metropolis-Hastings algorithm,” *The American Statistician*, vol. 49, no. 4, pp. 327–335, 1995.
- [112] P. J. Green, “Reversible jump Markov chain Monte Carlo computation and Bayesian model determination,” *Biometrika*, vol. 82, no. 4, pp. 711–732, 1995.
- [113] E. Arjas and D. Gasbarra, “Nonparametric Bayesian Inference from Right Censored Data, using the Gibbs Sampler,” *Statistica Sinica*, vol. 4, no. 2, pp. 505–524, 1994.
- [114] C. E. Rasmussen and C. Williams, *Gaussian Processes in Machine Learning*. MIT Press, 2006.
- [115] A. Pepelyshev, “The role of the nugget term in the gaussian process method,” in *mODa 9 – Advances in Model-Oriented Design and Analysis* (A. Giovagnoli, A. C. Atkinson, B. Torsney, and C. May, eds.), (Heidelberg), pp. 149–156, Physica-Verlag HD, 2010.
- [116] J. Bernardo, J. Berger, A. Dawid, A. Smith, *et al.*, “Regression and classification using gaussian process priors,” *Bayesian statistics*, vol. 6, p. 475, 1998.

-
- [117] N. L. Johnson, S. Kotz, and N. Balakrishnan, *Continuous Univariate Distributions*, vol. 1. Wiley Series in Probability and Mathematical Statistics, 2nd ed., 1994.
- [118] R. M. Neal, “Suppressing random walks in markov chain monte carlo using ordered overrelaxation,” tech. rep., Department of Statistics, University of Toronto, 1995.
- [119] I. Murray, D. MacKay, and R. P. Adams, “The gaussian process density sampler,” in *Advances in Neural Information Processing Systems* (D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, eds.), vol. 21, Curran Associates, Inc., 2008.
- [120] J. S. Simonoff, *Smoothing Methods in Statistics*. Springer New York, 1996.
- [121] L. Csató and M. Opper, “Sparse on-line gaussian processes,” *Neural Computation*, vol. 14, pp. 641–668, mar 2002.
- [122] J. Quiñonero Candela and C. E. Rasmussen, “A unifying view of sparse approximate gaussian process regression,” *J. Mach. Learn. Res.*, vol. 6, p. 1939?1959, Dec. 2005.
- [123] J. Hensman, N. Fusi, and N. D. Lawrence, “Gaussian processes for big data,” *CoRR*, vol. abs/1309.6835, 2013.
- [124] J. Gentle, *Computational Statistics*. Springer, 2009.
- [125] P. Bientinesi, B. Gunter, and R. A. van de Geijn, “Families of algorithms related to the inversion of a symmetric positive definite matrix,” *ACM Transactions on Mathematical Software*, vol. 35, pp. 1–22, jul 2008.
- [126] R. M. Neal, “Monte carlo implementation of gaussian process models for bayesian regression and classification,” *arXiv preprint physics/9701026*, 1997.
- [127] M. Shinozuka and G. Deodatis, “Simulation of stochastic processes by spectral representation,” *Applied Mechanics Reviews*, vol. 44, pp. 191–204, apr 1991.

- [128] J. W. Cooley and J. W. Tukey, “An algorithm for the machine calculation of complex fourier series,” *Mathematics of Computation*, vol. 19, no. 90, pp. 297–301, 1965.
- [129] M. Frigo and S. Johnson, “The design and implementation of FFTW3,” *Proceedings of the IEEE*, vol. 93, pp. 216–231, feb 2005.
- [130] T. A. Rooney, E. J. Sass, and A. P. Thomas, “Agonist-induced cytosolic calcium oscillations originate from a specific locus in single hepatocytes,” *Journal of Biological Chemistry*, vol. 265, no. 18, pp. 10792–10796, 1990.
- [131] F. Papangelou, “Integrability of expected increments of point processes and a related random change of scale,” *Transactions of the American Mathematical Society*, vol. 165, pp. 483–506, 1972.
- [132] M. B. Wilk, R. Gnanadesikan, and M. J. Huyett, “Probability plots for the gamma distribution,” *Technometrics*, vol. 4, no. 1, pp. 1–20, 1962.
- [133] M. B. Wilk and R. Gnanadesikan, “Probability plotting methods for the analysis of data,” *Biometrika*, vol. 55, no. 1, pp. 1–17, 1968.
- [134] Y. Ogata, “Statistical models for earthquake occurrences and residual analysis for point processes,” *Journal of the American Statistical Association*, vol. 83, no. 401, pp. 9–27, 1988.
- [135] M. Falcke, M. Moein, A. Tilūnaitė, R. Thul, and A. Skupin, “On the phase space structure of IP₃ induced Ca²⁺ signalling and concepts for predictive modeling,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 28, p. 045115, apr 2018.
- [136] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer New York, 2009.
- [137] V. Estivill-Castro, “Why so many clustering algorithms,” *ACM SIGKDD Explorations Newsletter*, vol. 4, pp. 65–75, jun 2002.

- [138] D. Arthur and S. Vassilvitskii, “K-means++: the advantages of careful seeding,” in *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1027–1035, 2007.
- [139] D. Comaniciu and P. Meer, “Mean shift: a robust approach toward feature space analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.
- [140] M. Ester, H. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pp. 226–231, AAAI Press, 1996.
- [141] A. Ng, M. Jordan, and Y. Weiss, “On spectral clustering: Analysis and an algorithm,” in *Advances in Neural Information Processing Systems* (T. Dietterich, S. Becker, and Z. Ghahramani, eds.), vol. 14, pp. 849–856, MIT Press, 2001.
- [142] A. Delaigle, P. Hall, and T. Pham, “Clustering functional data into groups by using projections,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 81, pp. 271–304, Feb 2019.
- [143] A. Haar, “Zur theorie der orthogonalen funktionensysteme,” *Mathematische Annalen*, vol. 69, pp. 331–371, Sep 1910.
- [144] L. Debnath and F. A. Shah, *Wavelet Transforms and Their Applications*. Birkhäuser Boston, 2015.