GRK
Sensory Computation
in Neural Systems

bccn

# Bayesian inference of inhomogeneous point process models

## Methodological advances and modelling of neuronal spiking data


vorgelegt von

Master of Science

Christian Donner

ORCID: 0000-0002-4499-2895


von der Fakultät IV – Elektrotechnik und Informatik
der Technischen Universität Berlin


zur Erlangung des akademischen Grades
Doktor der Naturwissenschaften
- Dr. rer. nat. -
genehmigte Dissertation

Promotionsausschuss:

Vorsitzender: Prof. Dr. Georgios Smaragdakis
Gutachter: Prof. Dr. Manfred Opper
Gutachter: Prof. Dr. Guido Sanguinetti
Gutachter: Prof. Dr. Jakob Macke

Tag der wissenschaftlichen Aussprache: 21. Februar 2019

Berlin 2019

# Acknowledgements

I dedicate the first lines of my PhD thesis to explain why this is the only section I write in the first person singular.

I want to express my sincere gratitude to Prof. Manfred Opper, who has not only been my PhD supervisor, but became a mentor to me in the past three years. Without the many hours of inspirational conversations, the many advices and ideas this thesis in its present form would not have been possible.

My special thanks go to Josef, and Hideaki. The vivid discussions and collaboration with each of them always ended up in interesting projects, because the common scientific interest quickly turned into friendship. I also thank Hideaki for giving me the great opportunity to come to his lab in Kyoto for one month.

I am grateful to the Group of Artificial Intelligence, namely Andreas, Burak, Cordula, Dimitra, Florian, Ludovica, and Theo for many interesting discussions, Christmas parties and the flowers at every birthday. In particular, I thank Noa for bringing the Pólya–Gamma to the group and proofreading this thesis in the final stages.

I am obliged to the Bernstein Center for Computational Neuroscience Berlin and the Graduiertenlolleg GRK1589/2 "Sensory Computation and Neural Systems" for their financial and formative support, which gave me all the guidance and freedom a PhD student can wish for. I also thank them for the travel support to go to conferences, summer schools, and retreats. Furthermore, I was kindly supported financially by the Deutsche Forschungsgemeinschaft through the grant CRC 1294 for two months.

For giving me the opportunity to present my work regularly, and the honest, constructive feedback I am grateful to Prof. Klaus Obermayer and the Neural Information Processing Group. I thank Franzi for proofreading the thesis and for the many coffee breaks in the afternoon, that helped to mobilise the concentration for the final hours of the day.

Many thanks to Lara, Greg, and Erik – the students I had the pleasure to co-supervised. During their months in our group I surely learnt as much from them as they did from me.

Finally, I thank my family for their strong support despite my troubles to explain what I was doing.

The last lines I want to devote to Roberta: You have all my gratitude for your strong encouragement, patience, and love! The past years would have been difficult without you.

C.D.

*Berlin, 18$^{th}$ of December 2018*

# Abstract

Arrival times of airplanes, positions of car accidents or astronomical objects in space, locations of ecological crisis, spike times of neurons, etc. are all data that surround us and can be viewed as realisations of point processes. Nowadays, the modelling of these data becomes increasingly more important, when we attempt to draw meaningful conclusions from this ever expanding amount of data. Models describing the statistics of point process data have been proposed in the past. However, to extract the model parameters given the observation of point process data, is generally challenging. Point process likelihoods of the observed data given the model parameters are difficult to deal with in practice because of their functional form. For Bayesian inference, where we aim at a tractable posterior distribution over the model parameters given the data, the task is even more demanding.

In the first part of this thesis we focus on a specific model class for point process data. The central object of point process models is the non–negative intensity function, which determines the likelihood of registering an event at any given position in the observed space. To enforce non–negativity, point process models have been proposed, where the intensity function depends non–linearly on the model parameters via a scaled sigmoidal link function. By the augmentation of latent variables we show, that the likelihood of this model class can be rendered into a novel favourable form enabling efficient and fast Bayesian inference schemes for a tractable posterior over the model parameters. We utilise this new augmented form of the likelihood to perform inference for a Poisson process model, where the intensity function depends on a Gaussian process. The resulting algorithms are one order of magnitude faster than state-of-the-art methods solving the same problem. Furthermore, we show that the same algorithms can be utilised for Bayesian density estimation, i.e. inferring a posterior over densities for an observed set of points. Concluding the first part, the inference problem for a Markov jump process model, namely the kinetic Ising model from statistical physics, is addressed using the new favourable representation of point process likelihoods.

The second part of the thesis is devoted to the statistical description of a specific instance of point process data – the cell-resolved spiking activity of neurons. These data are believed to reflect the information processing in the brain, and are highly non–stationary. We address the problem of statistical modelling such non–stationary spiking data. First, we propose a continuous time model accounting for effective couplings and temporal changes of the neuronal dynamics. Deriving an efficient inference algorithm, we demonstrate that the model can capture activity structures of in–vivo recorded data, that are not related to any controlled variables of the experiment. Finally, we propose a model which attempts to minimise the gap to the underlying system, based on the assumption of observing a population of integrate–and–fire neurons receiving common non–stationary input. We demonstrate how to efficiently evaluate the model likelihood, such that subsequent inference can be performed given spiking data recorded from a neuronal population.

The novel scalable inference algorithms for point process data, and the new description of non–stationary spiking data presented in this thesis expand our ability to investigate large and complex point process datasets and draw meaningful conclusions from these data.

# Zusammenfassung

Ankunftszeiten von Flugzeugen, Positionen von Autounfällen oder astronomischen Objekten im Weltraum, Orte von ökologischen Katastrophen, Impulse von Neuronen usw. sind allesamt Daten, die uns umgeben und als Realisierung von Punktprozessen betrachtet werden können. Die Modellierung dieser Daten wird heutzutage immer wichtiger, wenn wir versuchen, aus dieser ständig wachsenden Datenmenge aussagekräftige Schlüsse zu ziehen. In der Vergangenheit wurden Modelle zur statistischen Beschreibung von Punktprozessdaten vorgeschlagen.
Die Extraktion der Modellparameter bei der Beobachtung von Punktprozessdaten ist jedoch in der Regel eine Herausforderung. Die Punktprozessdichte der beobachteten Daten gegeben der Modellparameter, auch Likelihood-Funktion genannt, ist in der Praxis schwer zu handhaben aufgrund ihrer funktionalen Form. Für eine Bayes'sche Inferenz, bei der wir eine posteriore Verteilung über gewonnenen Modellparameter gegeben der Daten anstreben, ist die Aufgabe noch anspruchsvoller. Im ersten Teil dieser Arbeit konzentrieren wir uns auf eine bestimmte Modellklasse für Punktprozessdaten. Das zentrale Objekt von Punktprozessmodellen ist die nichtnegative Intensitätsfunktion, die die Wahrscheinlichkeit bestimmt, ein Ereignis an einer beliebigen Stelle im Raum zu beobachten. Um die Nicht-Negativität zu gewährleisten, wurden Punktprozessmodelle vorgeschlagen, bei denen die Intensitätsfunktion nicht-linear von den Modellparametern über eine skalierte sigmoide Funktion abhängt. Mit einer Modellaugmentation durch latente Variablen zeigen wir, dass die Punktprozessdichte dieser Modellklasse in eine neuartige, vorteilhafte Form gebracht werden kann, die effiziente und schnelle Bayes'sche Inferenzalgorithmen ermöglicht für eine praktisch nutzbare posteriore Verteilung über die Modellparameter. Wir nutzen diese neue augmentierte Darstellung der Likelihood-Funktion, um Inferenz für ein Poisson-Prozessmodell durchzuführen, bei dem die Intensitätsfunktion von einem Gaußprozess abhängt. Die resultierenden Algorithmen sind um eine Größenordnung schneller als moderne Methoden, die das gleiche Problem lösen. Anschließend zeigen wir, dass die gleichen Algorithmen verwendet werden können für eine Bayes'sche Dichteschätzung, d.h. die Inferenz einer posterioren Verteilung über die Dichte gegeben eine beobachtete Menge von Punkten. Abschließend wird das Inferenzproblem für ein Markov-Sprung-Prozessmodell, nämlich das kinetische Ising-Modell aus der statistischen Physik, mit der neuen augmentierten Darstellung der Likelihood-Funktion angegangen.
Der zweite Teil der Arbeit widmet sich der statistischen Beschreibung einer bestimmten Instanz von Punktprozessdaten - der zellaufgelösten Spiking-Aktivität von Neuronen, die im Allgemeinen nicht stationär ist. Wir befassen uns mit dem Problem der statistischen Modellierung solcher nicht stationärer Spiking-Daten. Zunächst schlagen wir ein kontinuierliches Zeitmodell vor, das effektive neuronale Kopplungen und zeitliche Veränderung der Daten berücksichtigt. Mit Hilfe der Herleitung eines effizienten Inferenzalgorithmus zeigen wir, dass die inferierten Modellparameter Aktivitätsstrukturen von *in–vivo* aufgezeichneten Daten aufzeigen können, die nicht mit den kontrollierten Variablen des Experiments verbunden sind. Schließlich schlagen wir ein Modell vor, das versucht, den Abstand zum zugrunde liegenden System zu minimieren, basierend auf der Annahme, dass die Population von *Integrate-and-Fire* Neuronen beobachtet wird, die einen gemeinsamen nichtstationären Input erfährt. Wir zeigen, wie man die Modell-Likelihood-Funktion effizient evaluiert, so dass mit Hilfe von Spikingdaten, die von einer neuronalen Population aufgenommen wurden, eine nachfolgende Inferenz durchgeführt werden kann.
Die neuartigen skalierbaren Inferenzalgorithmen für Punktprozessdaten und nicht stationäre Spiking-Daten erweitern unsere Möglichkeiten, große und komplexe Punktprozessdatensätze zu untersuchen und aus diesen Daten neue und aussagekräftige Schlussfolgerungen zu ziehen.

# Contents

# Chapter 1

# Introduction

In present days we are surrounded by an immeasurable amount of data. As we try to make sense of the constant stream of data, one of the major challenges we are facing is finding meaningful patterns and draw conclusions from them. In the machine learning community drawing conclusions from data is often thought of as obtaining a model that explains statistical properties of the data well. These models should be carefully designed, so that structures of interest are revealed.

In the Bayesian community 'obtaining a model' is defined as finding the posterior density over the model parameters $Z$ given the observed data $\mathcal{D}$. The posterior is obtained by *Bayes' rule* (Stuart and Ord 2010)

$$p(Z|\mathcal{D}) = \frac{L(\mathcal{D}|Z)p(Z)}{p(\mathcal{D})}, \tag{1}$$

where one assumes a likelihood of data given a set of parameters $L(\mathcal{D}|Z)$ and also some prior beliefs for those parameters $p(Z)$. The denominator $p(\mathcal{D})$ is the normalisation also known as evidence, which requires marginalising the nominator with respect to model parameters $Z$. Practical computation of expectations with respect to the posterior in Eq (1) is in general infeasible. *Bayesian inference* is concerned with finding a tractable posterior form. A simple example of Bayesian inference is regression, where one assumes that data $\mathcal{D}$ are noisy measurements of some underlying function, which we want to estimate. The type of assumed noise dictates the likelihood $L(\mathcal{D}|Z)$. Furthermore, one needs to assume which class of functions the target function belongs to, by defining the parameters $Z$. Prior beliefs (e.g. the function is continuous or differentiable) determine the prior $p(Z)$.

A specific problem, where Eq (1) actually results in a tractable posterior is Gaussian regression (Bishop 2006). In this case the measurement noise is normally distributed, the underlying function depends linearly on the parameters $Z$, and the prior $p(Z)$ is a Gaussian density. Under these assumptions the model is 'conjugate', i.e. the posterior has the same form as the prior distribution. Hence, Eq (1) yields a posterior, for which we are able to practically compute normalisation, expectation, etc.

In general this is not the case, as other (non–Gaussian) noise models and different types of problems (e.g. classification) impose likelihoods, for which posteriors are intractable. For these cases, maximising the (logarithm of the) nominator with respect to the parameters $Z$ is already difficult and one is often required to utilise numerical optimisation procedures. Those are often slow and contain themselves parameters, that need to be tuned to efficiently arrive at an optimal solution. To solve the Bayesian problem one needs to obtain not only a point estimate, but an optimal posterior distribution over the model parameters $Z$. This is usually intractable, because it is practically infeasible to compute the normalisation constant $p(\mathcal{D})$, and expectations with respect

to the posterior in Eq 1. To circumvent this issue many approaches resort to approximations, which in turn can be very slow, such that they do not scale to large datasets $\mathcal{D}$ or require numerical optimisation, which come with the same problems mentioned above. To obtain a tractable posterior many Bayesian inference methods have been proposed such as sampling schemes (Hastings 1970), Laplace approximation (Bishop 2006), expectation propagation (Minka 2001; Opper and Winther 2000), variational methods (Feynman et al. 1964), belief propagation (Yedidia 2013) etc. The optimisation problem becomes even more challenging for inference of stochastic processes with an infinite number of parameters $Z$.

In this thesis we will address Bayesian inference for a specific class of random processes, where the data are discrete events in an observed space. Data of this type appear in a broad range of applications, such as healthcare (Ahmed and Alkhamis 2009), forestry (Penttinen and Stoyan 2000), financial markets (Embrechts et al. 2011), weather forecast (Kilsby et al. 2007), and crisis prediction (Zammit-Mangion et al. 2012). Given this type of point observations the main challenges are (i) to infer the frequency of events at any given point in the observed space, and (ii) the probability of an event being in a certain subspace.

The simplest scenario in which these questions can be addressed assumes that all events are independent. Under this assumption the problem (i) is equivalent to estimating the intensity function of a Poisson process (Kingman 1993). Problem (ii) is called density estimation (Silverman 1986) and is closely related to problem (i) as we will see in this thesis. Those fundamental problems have been addressed in the Bayesian community several times (Adams et al. 2009; Lloyd et al. 2014; Murray et al. 2009; Riihimäki and Vehtari 2014), but model inference suffers from the obstacles stated above, i.e. they do not scale to problems with large datasets or require numerical optimisation schemes.

If the observed events are interdependent, the models mentioned above are not applicable any more. Hence another challenge, in addition to the aforementioned ones, is to create (iii) models that account for dependencies between data points. A particular case of such interdependent data can be encountered in neuroscience, where neuronal activity from many neurons is recorded in parallel. These data contain time points and neuron identifiers of observed neuronal events, namely action potentials also known as spikes. This kind of data, also known as spike train data, will be of major interest in the latter part of this thesis. The sequence of spikes generated by each neuron is believed to contain the main information which is transmitted to other neurons (Rieke et al. 1997) and allows neuronal ensembles to perform the computations required for perception, behaviour etc. In general, past events are important for an accurate description of temporal data and spiking data are no exception to that (Dayan and Abbott 2001). In addition to temporal dependence, these data are characterised by spatial dependencies, i.e. the intermingled activity of cells the observed neurons are synaptically connected to. Thus, for spiking data the challenge of quantifying the dependence of events translates to the question of how to infer the effective network structure. Several models have been suggested to tackle this issue (Chornoboy et al. 1988; Pillow et al. 2008; Zeng et al. 2013), and as before, inference often requires numerical optimisation.

Solving the inference problem becomes even more complex, when the data are non–stationary, i.e. the data statistics change over time, which is usually the case for spiking data. This poses the problem of (iv) how to infer a time–varying model structure. Several approaches in neuroscience addressed this issue with varying set of assumptions (Cunningham and Yu 2014; Kim and Shinomoto 2012; Pandarinath et al. 2018).

The models commonly chosen for statistical descriptions of spiking data are purely phenomenological, i.e. they provide a flexible statistical description without being physiologically constrained. There is little work that (v) tries to use plausible mechanistic models, that are constrained by incorporating a credible spiking mechanism. Some work (Ladenbauer et al. 2018; Mullowney and Iyengar 2008) showed that likelihoods can be derived and evaluated efficiently for a simple neuron model class. These models are broadly used to simulate plausible spiking data and are popular because they

allow analytical investigation of neuronal networks, while preserving a minimal set of the neuron's biophysical properties (Gerstner and Kistler 2002). However, the attempts to perform inference with these models given recorded spiking data are scarce, because the evaluation of likelihoods is thought to be difficult.

## Thesis outline

This thesis addresses specific instances of problems (i)–(v), mainly in the context of Bayesian inference, and consists of two parts.

The thesis' first part addresses the inference of models that have been already proposed in the past (Adams et al. 2009; Glauber 1963; Murray et al. 2009). We derive novel Bayesian inference schemes that are substantially more efficient than those proposed before. Our approach relies on iterative updates, that are analytically tractable and do not require numerical optimisation. Specifically, we focus on the inference of 3 models, that have similar likelihoods $L(\mathcal{D}|Z)$ and consider the problems (i)–(iii), respectively.

In chapter 2, we discuss a particular class of likelihoods, that is shared by the models mentioned previously, and briefly sketch how an alternative favourable representation can be obtained. With specific (Gaussian) priors the model becomes conditionally conjugate (Bishop and Tipping 2000; Blei et al. 2017), which allows for efficient optimisation procedures where the updates have an analytical form. Chapter 3 makes use of this representation for a doubly stochastic Poisson process (Adams et al. 2009), where the intensity to be inferred depends non–linearly on a Gaussian process (GP). We show in chapter 4, that a GP density model (Murray et al. 2009) can be transformed to the previous Poisson process model, which allows us to use the same techniques for inference as in chapter 3. In chapter 5, which constitutes the end of the first part, we show that the described augmentation is also applicable to a Markov jump process accounting for effective couplings among parallel binary processes (Glauber 1963; Zeng et al. 2013).

The second part addresses questions (iv)–(v) in the context of statistical modelling of non–stationary spiking data. Chapter 7 briefly introduces common problems in the statistical modelling of these data. In chapter 8, we propose a non–stationary extension of the model discussed in chapter 5, and derive an efficient Bayesian inference algorithm. After validating the accuracy of the inferred results for simulated data, we demonstrate practicality on experimentally recorded spike train data. Finally, in chapter 9 we tackle the problem of inference for physiologically constraint models for spiking data. We propose a neuronal model with a minimally plausible spiking mechanism, namely the leaky integrate–and–fire (LIF) neuron (Gerstner and Kistler 2002). Unlike the phenomenological models considered before, the model parameters have a straightforward biophysical interpretation. Here, we make use of the fact that the model likelihood can be efficiently evaluated (Ladenbauer et al. 2018). We consider a non–stationary scenario, where observed neurons are driven by an unknown doubly stochastic process, which we attempt to infer.

# Part I

# Efficient Bayesian inference for point processes

# Chapter 2

# Gaussian representation of a point process likelihood

In the previous chapter we mentioned, that exact Bayesian inference is practically infeasible, because the posterior in Eq (1) is intractable. However, in particular cases, such as linear regression, discussed in chapter 1, the posterior is analytically tractable due to the model conjugacy. For non–conjugate models *conditional* conjugacy can be achieved in particular cases by rewriting the model likelihood as expectation over a set of new latent augmentation variables (Meng and Van Dyk 1999). Conditional conjugacy means that the model is conjugate for the original model parameters given the augmentation variables and vice versa. Bayesian inference for conditionally conjugate models can not be solved exactly, but in general allows for more efficient inference algorithms than non–conjugate models (Bishop and Tipping 2000; Blei et al. 2017; Meng and Van Dyk 1999). In the following, we derive an augmentation scheme for a particular model class, which allows a new conditionally conjugate representations for those models.

The model class of interest in this part has specific properties. Observed data $\mathcal{D} = \{\boldsymbol{x}_n\}_{n=1}^N$ are discrete events in a continuous or discrete domain $\boldsymbol{x}_n \in \mathcal{X}$, and $\mathcal{X}$ is observed completely. The likelihood for such *point processes* (Daley and Vere-Jones 2008) has the form

$$p(\mathcal{D}|Z) = \prod_{\boldsymbol{x}_n \in \mathcal{D}} \Lambda_Z(\boldsymbol{x}_n) \exp\left(-\int_{\mathcal{X}} \Lambda_Z(\boldsymbol{x}) d\boldsymbol{x}\right), \tag{2}$$

where $\Lambda_Z : \mathcal{X} \to \mathbb{R}^+$ is the 'intensity' or 'rate' function parametrised by the model variables $Z$. The product on the right-hand side in Eq (2) corresponds to the events $\mathcal{D}$, while the integral accounts for the locations where no events are observed. Eq (2) is also called *complete data likelihood* (Wilkinson 2006). While not addressed here, Eq (2) can be easily extended to situations where different types of events exist. In the following we focus on models that assume Gaussian priors $p(Z)$. Since a Gaussian density is defined on the whole real space, but the intensity function is restricted to positive real numbers, $\Lambda_Z(\cdot)$ has to depend non–linearly on $Z$. While many choices of such *link functions* are possible (exponential, square, cumulative Gaussian or any mixture of non–negative functions), in this work we focus on models with a scaled sigmoidal link function

$$\Lambda_Z(\boldsymbol{x}) = c\ \sigma(h_Z(\boldsymbol{x})) = \frac{c}{1 + \exp(-h_Z(\boldsymbol{x}))}, \tag{3}$$

where $c \in \mathbb{R}^+$ and $h_Z : \mathcal{X} \to \mathbb{R}$ is a linear function of model parameters $Z$. Models of this form are non–conjugate to a Gaussian density and hence inference with such priors is challenging.

Another complication arises, when the model parameters depend on the space $\mathcal{X}$. If e.g. $\mathcal{X} \subseteq \mathbb{R}^d$, then Eq (2) depends on a continuum of variables $Z$, due to the integral in the exponent. The

inference problem in Eq (1) yields a posterior over an infinite dimensional object, which is practically infeasible. To obtain a tractable posterior over $Z$, we have to resort to approximations (Matthews et al. 2016), as we will see in the subsequent chapters.

## Approximate inference for complete data likelihood

As we have seen, the posterior in Eq (1) is intractable for models previously discussed. However, a zoo of approximate inference methods for such problems exists. The major body of work relies either on Markov Chain Monte Carlo (MCMC) (Murray et al. 2012) or variational inference methods (Matthews et al. 2016) and this thesis is no exception to that. However, the algorithms we propose diverge from previously proposed approaches.

MCMC Sampling algorithms for models with complete data likelihood are usually of the Metropolis Hastings type, based on rejections (Adams et al. 2009; Murray et al. 2012, 2009). While long Markov chains converge theoretically to the correct posterior, in practice, convergence is slow and the algorithms do not scale well with the amount of data.

On the other hand, variational algorithms assume that the approximate posterior belongs to a certain family of densities, for which the variational problem is tractable. The goal is to minimise the Kullback–Leibler divergence between the variational and the true posterior. This is equivalent to maximising a lower bound of the evidence $p(\mathcal{D})$ (Bishop 2006; Blei et al. 2017). Often one assumes a variational posterior density of a specific functional form, e.g. a Gaussian density. The variational lower bound is then maximised with respect to the posterior parameters via numerical optimisation, e.g. gradient ascent algorithms (Hensman et al. 2015b; Lloyd et al. 2014). These methods have been proven faster than sampling schemes and yield fairly good approximate posteriors. The drawback of these approaches is that the required gradients can be solved analytically only for specific functions $\Lambda_Z(\boldsymbol{x})$, subject to Gaussian priors with specific kernels and particular domains $\mathcal{X}$, even under the Gaussian posterior assumption (Lloyd et al. 2014). In other cases discretisation of the domain $\mathcal{X}$, sampling or numerical approximations of the involved integrals are required (Flaxman et al. 2017; Hensman et al. 2015b).

In the following sections we show that for the model class with complete data likelihood in Eq (2), sigmoid link function (3), and a Gaussian prior, we can derive an augmented representation of Eq (2) such that the model becomes conditionally conjugate. Conditioned on the new latent variables the posterior density over the model parameters $Z$ is a Gaussian density, for which we can analytically derive the mean and covariance matrix. The analytical calculation of the Gaussian posterior is the major difference to the previously discussed approaches, which require numerical optimisation procedures.

## Achieving conjugacy via variable augmentation

As we know from the case of Gaussian regression, the Gaussian likelihood is conjugate to a Gaussian prior. Hence, we aim at a Gaussian representation of Eq (2) with intensity function of form as in Eq (3). First, we focus on the product of the intensity function at the observed events $\mathcal{D}$, i.e. $\prod_{\boldsymbol{x}_n \in \mathcal{D}} c\sigma(h_Z(\boldsymbol{x}_n))$.

**Pólya–Gamma augmentation** Polson et al. (2013) showed that the sigmoidal function can be rewritten as an infinite scale Gaussian mixture model

$$\sigma(z) = \frac{1}{2} \int_{\mathbb{R}^+} \exp\left(\frac{z}{2} - \frac{z^2}{2}\omega\right) p_{\text{PG}}(\omega|1,0) d\omega, \tag{4}$$

where the precision $\omega$ of the Gaussian density is distributed according to a Pólya–Gamma density $p_{\text{PG}}(\cdot|1,0)$. This general Pólya–Gamma density $p_{\text{PG}}(\cdot|b_1,b_2)$ parametrised by $b_1, b_2$ has several interesting properties. It can be sampled efficiently, it is conjugate to $\exp(-\frac{z}{2}\omega)$, and its moments can be computed analytically (Polson et al. 2013). For the product over observations $\mathcal{D}$ in Eq (2) we obtain a Gaussian representation in terms of latent variables $Z$

$$\prod_{\boldsymbol{x}_n \in \mathcal{D}} c\, \sigma(h_Z(\boldsymbol{x}_n)) \propto c^N \prod_{\boldsymbol{x}_n \in \mathcal{D}} \int_{\mathbb{R}^+} \exp\left(\frac{h_Z(\boldsymbol{x}_n)}{2} - \frac{[h_Z(\boldsymbol{x}_n)]^2}{2}\omega_n\right) p_{\text{PG}}(\omega_n|1,0) d\omega_n. \tag{5}$$

This term is fully conditionally conjugate, i.e. Eq (5) is proportional to a Pólya–Gamma density for each $\omega_n$, given the model parameters $Z$, and proportional to a Gaussian over $Z$ given $\{\omega_n\}_{n=1}^N$.

For models with binomial likelihoods the conditional conjugate likelihood is achieved with this augmentation. It has been utilised for Bayesian logistic regression problems (Linderman et al. 2016; Wenzel et al. 2018) and other likelihoods, that can be written as products of sigmoids (Linderman et al. 2015; Scott and Pillow 2012). In the following we aim at finding a similar representation for the exponential term in Eq (2).

**Marked Poisson process augmentation** For the sigmoidal link function the equality $\sigma(z) = 1 - \sigma(-z)$ holds, and the exponent in Eq (2) can be written as

$$\exp\left(-\int_{\mathcal{X}} c\sigma(h_Z(\boldsymbol{x}))d\boldsymbol{x}\right) = \exp\left(\int_{\mathcal{X}} (\sigma(-h_Z(\boldsymbol{x})) - 1)\, c d\boldsymbol{x}\right). \tag{6}$$

This equation has the form of a characteristic function of a Poisson process. Thus, Campbell's theorem (Kingman 1993, chap. 3) allows us to rewrite Eq (6) as

$$\exp\left(\int_{\mathcal{X}} (\sigma(-h_Z(\boldsymbol{x})) - 1)\, \Lambda(\boldsymbol{x}) d\boldsymbol{x}\right) = \mathbb{E}_{P_{\Lambda(\boldsymbol{x})}}\left[\prod_{\boldsymbol{x}\in\Pi_{\mathcal{X}}} \sigma(-h_Z(\boldsymbol{x}))\right], \qquad \text{with } \Lambda(\boldsymbol{x}) = c, \tag{7}$$

where the expectation is taken with respect to the probability measure $P_\Lambda(\boldsymbol{x})$ of a Poisson process with intensity $\Lambda(\boldsymbol{x})$ on domain $\mathcal{X}$. $\Pi_{\mathcal{X}}$ is a random set of points on this domain. In fact this representation was used to derive Metropolis Hastings sampler for a Poisson process model (Adams et al. 2009), being part of the model class discussed here. Directly applying Eq (7) to Eq (6) does not yield the desired conjugate form of the likelihood. However, by first applying the Pólya-Gamma augmentation (4), and then invoking Eq (7), the conjugate form is achieved

$$\begin{aligned}
&\exp\left(\int_{\mathcal{X}} (\sigma(-h_Z(\boldsymbol{x})) - 1)\, c\; d\boldsymbol{x}\right) \\
&= \exp\left(\int_{\mathcal{X}\times\mathbb{R}^+} \left(\frac{1}{2}\exp\left(-\frac{h_Z(\boldsymbol{x})}{2} - \frac{[h_Z(\boldsymbol{x})]^2}{2}\omega\right) - 1\right) p_{\text{PG}}(\omega|1,0) c\; d\omega d\boldsymbol{x}\right) \\
&= \mathbb{E}_{P_{\Lambda(\boldsymbol{x},\omega)}}\left[\prod_{(\boldsymbol{x},\omega)\in\Pi_{\mathcal{X}\times\mathbb{R}^+}} \frac{1}{2}\exp\left(-\frac{h_Z(\boldsymbol{x})}{2} - \frac{[h_Z(\boldsymbol{x})]^2}{2}\omega\right)\right],
\end{aligned} \tag{8}$$

where the intensity $\Lambda(\boldsymbol{x},\omega) = c\; p_{\text{PG}}(\omega|1,0)$. In fact, $P_{\Lambda(\boldsymbol{x},\omega)}$ is a measure over a *marked Poisson process* in the product space $\mathcal{X}\times\mathbb{R}^+$, where the Pólya–Gamma variables are *marks* on the events in the data domain $\mathcal{X}$ (Kingman 1993, chap. 5). Note that a Poisson process measure is conjugate to a Poisson process likelihood (Kingman 1993).

**Full conditional conjugacy** By Eq (5) and Eq (8) we achieve a conditional conjugate representation of likelihood in Eq (2). The resulting joint likelihood of model variables $Z$ and of the latent variables $\{\omega_n\}_{n=1}^N, \Pi_{\mathcal{X}\times\mathbb{R}^+}$ is conditionally conjugate for every set of variables. This fact allows deriving inference algorithms, which are much more efficient compared to others that can be

obtained from working with Eq (2) directly (Meng and Van Dyk 1999).

## Inference algorithms for the augmented model

For conditionally conjugate models, block Gibbs sampling (Geman and Geman 1984) is a MCMC algorithm that is rejection free, and hence expected to converge much faster than other Metropolis–Hastings algorithms discussed earlier (Meng and Van Dyk 1999). For the augmented model derived here, it is in fact possible to sample from each conditional posterior efficiently.

The new conjugate form of the model also allows for a variational mean–field algorithm (Bishop 2006; Blei et al. 2017), where it is assumed that the model parameters $Z$ are independent of the augmentation variables $\{\omega_n\}_{n=1}^N, \Pi_{\mathcal{X}\times\mathbb{R}^+}$. Due to the conjugacy, we can derive the variational mean–field posterior analytically and no gradient optimisation is required, unlike the previously discussed methods (Hensman et al. 2015b; Lloyd et al. 2014).

Furthermore, the augmentation allows to find the maximum likelihood or maximum a posteriori estimate of $Z$ exactly via an efficient expectation–maximisation algorithm (EM) (Dempster et al. 1977; Meng and Van Dyk 1999). To obtain an approximate posterior, one can perform the Laplace approximation (Bishop 2006) by calculating the Hessian of the original likelihood (2) with respect to the model variables $Z$.

## Outline of part I

Having established the common ground of the first part of this thesis, we now present a short overview of the models discussed in chapter 3–5.

In chapter 3 we discuss the sigmoidal Gaussian Cox process model, which was originally introduced by Adams et al. (2009). In this work we derive the augmentation scheme outlined above in a more rigorous way, together with deriving a variational mean field algorithm and an EM algorithm with Laplace approximation. We establish that our algorithms are one order of magnitude faster than state-of-the-art inference methods for the same model, and are compatible with variational inference algorithms for competing models.

Chapter 4 deals with a Gaussian process (GP) density model suggested by Murray et al. (2009). This model is, strictly speaking, not part of the discussed model class, because its likelihood does not have the form of Eq (2). However, we show that with one additional variable augmentation the required functional form of the likelihood is obtained. While in chapter 3 we restrict ourselves to compact domains $\mathcal{X}$, the GP density model allows to consider domains without boundaries, by introducing base measures. We develop the block Gibbs sampler and a variational mean field algorithm for approximate inference.

Finally, in chapter 5 we apply the previously discussed augmentation scheme to a specific Markov jump process model, namely the kinetic Ising model. It was proposed first in statistical physics to describe the dynamics of binary spins, which are coupled to each other (Glauber 1963). In recent years the *inverse* problem (Nguyen et al. 2017), i.e. obtaining the model parameters from observed binary data, received increasing attention due to the use of such models in neuroscience (Dunn et al. 2015; Schneidman et al. 2006). With the derived augmentation scheme, we establish the EM algorithm for obtaining the $L1$–penalised maximum likelihood estimate exactly. We make use of the fact, that the Laplace prior can be also rendered into a Gaussian form, by an additional augmentation (Pontil et al. 2000). We develop a variational mean field algorithm for this model.

Chapter 6 discusses related work, limiting factors and future directions of research.

# Chapter 3

# Journal article: *Efficient Bayesian Inference of Sigmoidal Gaussian Cox Processes*

Published in the journal *Journal of Machine Learning Research* (JMLR, Inc. and Microtome Publishing, United States).

Authors:
Christian Donner[1,2], Manfred Opper[1,2]
[1]Technische Universität Berlin. [2]Bernstein Center for Computational Neuroscience Berlin.

Details:
Submitted: December 2017
Accepted: October 2018
URL: http://jmlr.org/papers/v19/17-759.html


**Chapter 3** This chapter comprises the publication (Donner and Opper 2018b), which is authored by myself (CD), and Prof. Manfred Opper (MO).

Contributions:
CD and MO conceived and designed the work. CD derived the inference algorithms and developed the Python code. CD performed the numerical experiments. CD wrote the manuscript with substantial contribution of MO.

Python code on GitHub: https://github.com/christiando/SGCP_Inference.git

Journal of Machine Learning Research 19 (2018) 1-34 Submitted 12/17; Revised 10/18; Published 11/18

# Efficient Bayesian Inference of Sigmoidal Gaussian Cox Processes

**Christian Donner** CHRISTIAN.DONNER@BCCN-BERLIN.DE
**Manfred Opper** MANFRED.OPPER@TU-BERLIN.DE
*Artificial Intelligence Group*
*Technische Universität Berlin*
*Berlin, Germany*

**Editor:** Ryan Adams

## Abstract

We present an approximate Bayesian inference approach for estimating the intensity of an inhomogeneous Poisson process, where the intensity function is modelled using a Gaussian process (GP) prior via a sigmoid link function. Augmenting the model using a latent marked Poisson process and Pólya–Gamma random variables we obtain a representation of the likelihood which is conjugate to the GP prior. We estimate the posterior using a variational free–form mean field optimisation together with the framework of sparse GPs. Furthermore, as alternative approximation we suggest a sparse Laplace's method for the posterior, for which an efficient expectation–maximisation algorithm is derived to find the posterior's mode. Both algorithms compare well against exact inference obtained by a Markov Chain Monte Carlo sampler and standard variational Gauss approach solving the same model, while being one order of magnitude faster. Furthermore, the performance and speed of our method is competitive with that of another recently proposed Poisson process model based on a quadratic link function, while not being limited to GPs with squared exponential kernels and rectangular domains.

**Keywords:** Poisson process; Cox process; Gaussian process; data augmentation; variational inference

## 1. Introduction

Estimating the intensity rate of discrete events over a continuous space is a common problem for real world applications such as modeling seismic activity (Ogata, 1998), neural data (Brillinger, 1988), forestry (Stoyan and Penttinen, 2000) and so forth. A particularly common approach is a Bayesian model based on a so–called Cox process (Cox, 1955). The observed events are assumed to be generated from a Poisson process, whose intensity function is modeled as another random process with a given prior probability measure. The problem of inference for such type of models has also attracted interest in the Bayesian machine learning community in recent years. Møller et al. (1998); Brix and Diggle (2001); Cunningham et al. (2008) assumed that the intensity function is sampled from a Gaussian Process (GP) prior (Rasmussen and Williams, 2006). However, to restrict the intensity function of the Poisson process to nonnegative values, a common strategy is to choose a nonlinear link function which takes the GP as its argument and returns a valid intensity. Based on the success of variational approximations to deal with complex Gaussian process

models, the inference problem for such Poisson models has attracted considerable interest in the machine learning community.

While powerful black–box variational Gaussian inference algorithms are available which can be applied to arbitrary link–functions, the choice of link–functions is not only crucial for defining the prior over intensities but can also be important for the efficiency of variational inference. The 'standard' choice of Cox processes with an exponential link function was treated in (Hensman et al., 2015). However, variational Gaussian inference for this link function has the disadvantage that the posterior variance becomes decoupled from the observations (Lloyd et al., 2015).[1] An interesting choice is the quadratic link function of (Lloyd et al., 2015) for which integrations over the data domain, which are necessary for sparse GP inference, can be (for specific kernel) computed analytically.[2] For both models, the minimisation of the variational free energies is performed by gradient descent techniques.

In this paper we will deal with approximate inference for a model with a sigmoid link–function. This model was introduced by (Adams et al., 2009) together with a MCMC sampling algorithm which was further improved by (Gunter et al., 2014) and (Teh and Rao, 2011). Kirichenko and van Zanten (2015) have shown that the model has favourable (frequentist) theoretical properties provided priors and hyperparameters are chosen appropriately. In contrast to a direct variational Gaussian approximation for the posterior distribution of the latent function, we will introduce an alternative type of variational approximation which is specially designed for the *sigmoidal Gaussian Cox process*. We build on recent work on Bayesian logistic regression by data augmentation with Pólya–Gamma random variables (Polson et al., 2013). This approach was already used in combination with GPs (Linderman et al., 2015; Wenzel et al., 2017), for stochastic processes in discrete time (Linderman et al., 2017), and for jump processes (Donner and Opper, 2017). We extend this method to an augmentation by a latent, marked Poisson process, where the marks are distributed according to a Pólya–Gamma distribution.[3] In this way, the augmented likelihood becomes conjugate to a GP distribution. Using a combination of a mean–field variational approximation together with sparse GP approximations (Csató and Opper, 2002; Csató, 2002; Titsias, 2009) we obtain explicit analytical variational updates leading to fast inference. In addition, we show that the same augmentation can be used for the computation of the maximum a posteriori (MAP) estimate by an expectation–maximisation (EM) algorithm. With this we obtain a Laplace approximation to the non–augmented posterior.

The paper is organised as follows: In section 2, we introduce the sigmoidal Gaussian Cox process model and its transformation by the variable augmentation. In section 3, we derive a variational mean field method and an EM–algorithm to obtain the MAP estimate, followed by the Laplace approximation of the posterior. Both methods are based on a sparse GP approximation to make the infinite dimensional problem tractable. In section 4, we demonstrate the performance of our method on synthetic datasets and compare with the results of a Monte Carlo sampling method for the model and the variational approximation of Hensman et al. (2015), which we modify to solve the Cox–process model with the scaled sigmoid link function. Then we compare our method to the state-of-the-art inference

1. Samo and Roberts (2015) propose an efficient approximate sampling scheme.
2. For a frequentist nonparametric approach to this model, see (Flaxman et al., 2017). For a Bayesian extension see (Walder and Bishop, 2017).
3. For a different application of marked Poisson processes, see (Lloyd et al., 2016).

algorithm (Lloyd et al., 2015) on artificial and real datasets with up to $10^4$ observations. Section 5 presents a discussion and an outlook.

## 2. The Inference problem

We assume that $N$ events $\mathcal{D} = \{\boldsymbol{x}_n\}_{n=1}^N$ are generated by a Poisson process. Each point $\boldsymbol{x}_n$ is a $d$–dimensional vector in the compact domain $\mathcal{X} \subset \mathbb{R}^d$. The goal is to infer the varying *intensity function* $\Lambda(\boldsymbol{x})$ (the mean measure of the process) for all $\boldsymbol{x} \in \mathcal{X}$ based on the likelihood

$$L(\mathcal{D}|\Lambda) = \exp\left(-\int_{\mathcal{X}} \Lambda(\boldsymbol{x}) d\boldsymbol{x}\right) \prod_{n=1}^{N} \Lambda(\boldsymbol{x}_n),$$

which is equal (up to a constant) to the density of a Poisson process having intensity $\Lambda$ (see Appendix C and (Konstantopoulos et al., 2011)) with respect to a Poisson process with unit intensity. In a Bayesian framework, a prior over the intensity makes $\Lambda$ a random process. Such a doubly stochastic point process is called *Cox process* (Cox, 1955). Since one needs $\Lambda(\boldsymbol{x}) \geq 0$, Adams et al. (2009) suggested a reparametrization of the intensity function by $\Lambda(\boldsymbol{x}) = \lambda\sigma(g(\boldsymbol{x}))$, where $\sigma(x) = (1 + e^{-x})^{-1}$ is the sigmoid function and $\lambda$ is the maximum intensity rate. Hence, the intensity $\Lambda(\boldsymbol{x})$ is positive everywhere, for any arbitrary function $g(\boldsymbol{x}) : \mathcal{X} \to \mathbb{R}$ and the inference problem is to determine this function. Throughout this work we assume that $g(\cdot)$ will be modelled as a GP (Rasmussen and Williams, 2006) and the resulting process is called *sigmoidal Gaussian Cox process*. The likelihood for $g$ becomes

$$L(\mathcal{D}|g, \lambda) = \exp\left(-\int_{\mathcal{X}} \lambda\sigma(g(\boldsymbol{x})) d\boldsymbol{x}\right) \prod_{n=1}^{N} \lambda\sigma(g_n), \tag{1}$$

where $g_n \doteq g(\boldsymbol{x}_n)$. For Bayesian inference we define a GP prior measure $P_{\mathrm{GP}}$ with zero mean and covariance kernel $k(\boldsymbol{x}, \boldsymbol{x}') : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$. $\lambda$ has as prior density (with respect to the ordinary Lebesgue measure) $p(\lambda)$ which we take to be a Gamma density with shape-, and rate parameter $\alpha_0$ and $\beta_0$, respectively. Hence, for the prior we get the product measure $dP_{\mathrm{prior}} = dP_{\mathrm{GP}} \times p(\lambda)d\lambda$. The posterior density $\boldsymbol{p}$ (with respect to the prior measure) is given by

$$\boldsymbol{p}(g, \lambda|\mathcal{D}) \doteq \frac{dP_{\mathrm{posterior}}}{dP_{\mathrm{prior}}}(g, \lambda|\mathcal{D}) = \frac{L(\mathcal{D}|g, \lambda)}{\mathbb{E}_{P_{\mathrm{prior}}}\left[L(\mathcal{D}|g, \lambda)\right]}. \tag{2}$$

The normalising expectation in the denominator on the right hand side is with respect to the probability measure $P_{\mathrm{prior}}$. To deal with the infinite dimensionality of GPs and Poisson processes we require a minimum of extra notation. We introduce densities or *Radon–Nikodým derivatives* such as defined in Equation (2) (see Appendix C or de G. Matthews et al. (2016)) with respect to infinite dimensional measures by boldface symbols $\boldsymbol{p}(\boldsymbol{z})$. On the other hand, non–bold densities $p(\boldsymbol{z})$ denote densities in the 'classical' sense, which means they are with respect to Lebesgue measure $d\boldsymbol{z}$.

Bayesian inference for this model is known to be doubly intractable (Murray et al., 2006). The likelihood in Equation (1) contains the integral of $g$ over the space $\mathcal{X}$ in the exponent and the normalisation of the posterior in Equation (2) requires calculating expectation of Equation (1). In addition inference is hampered by the fact, that likelihood (1) depends

non–linearly on $g$ (through sigmoid and exponent of sigmoid). In the following we tackle this by an augmentation scheme for the likelihood, such that it becomes conjugate to a GP prior and we subsequently can derive an analytic form of a variational posterior given one simple mean field assumption (Section 3).

### 2.1 Data augmentation I: Latent Poisson process

We will briefly introduce a data augmentation scheme by a latent Poisson process which forms the basis of the sampling algorithm of Adams et al. (2009). We will then extend this method further to an augmentation by a *marked* Poisson process. We focus on the exponential term in Equation (1). Utilizing the well known property of the sigmoid that $\sigma(x) = 1 - \sigma(-x)$ we can write

$$\exp\left(-\int_{\mathcal{X}} \lambda\sigma(g(\boldsymbol{x}))d\boldsymbol{x}\right) = \exp\left(-\int_{\mathcal{X}} \left(1-\sigma(-g(\boldsymbol{x}))\right)\lambda d\boldsymbol{x}\right). \tag{3}$$

The left hand side has the form of a characteristic functional of a Poisson process. Generally, for a random set of points $\Pi_{\mathcal{Z}} = \{\boldsymbol{z}_m; \boldsymbol{z}_m \in \mathcal{Z}\}$ on a space $\mathcal{Z}$ and with a function $h(\boldsymbol{z})$, this is defined as

$$\mathbb{E}_{P_\Lambda}\left[\prod_{\boldsymbol{z}_m \in \Pi_{\mathcal{Z}}} e^{h(\boldsymbol{z}_m)}\right] = \exp\left(-\int_{\mathcal{Z}} \left(1-e^{h(\boldsymbol{z})}\right)\Lambda(\boldsymbol{z})d\boldsymbol{z}\right), \tag{4}$$

where $P_\Lambda$ is the probability measure of a Poisson process with intensity $\Lambda(\boldsymbol{z})$. Equation (4) can be derived by Campbell's theorem (see Appendix A and (Kingman, 1993, chap. 3)) and identifies a Poisson process uniquely.

Setting $h(\boldsymbol{z}) = \ln\sigma(-g(\boldsymbol{z}))$, and $\mathcal{Z} = \mathcal{X}$, and combining Equation (3) and (4) we obtain the likelihood used by Adams et al. (2009, Eq. 4). However, in this work we make use of another augmentation, before invoking Campbell's theorem. This will result in a likelihood which is conjugate to the model priors and further simplifies inference.

### 2.2 Data augmentation II: Pólya–Gamma variables and marked Poisson process

Following Polson et al. (2013) we represent the inverse of the hyperbolic cosine as a scaled Gaussian mixture model

$$\cosh^{-b}(z/2) = \int_0^\infty e^{-\frac{z^2}{2}\omega} p_{\mathrm{PG}}(\omega|b,0)d\omega, \tag{5}$$

where $p_{\mathrm{PG}}$ is a *Pólya–Gamma* density (Appendix B). We further define the *tilted* Pólya–Gamma density by

$$p_{\mathrm{PG}}(\omega|b,c) \propto e^{-\frac{c^2}{2}\omega} p_{\mathrm{PG}}(\omega|b,0), \tag{6}$$

where $b > 0$ and $c$ are parameters. We will not need an explicit form of this density, since the subsequently derived inference algorithms will only require the first moments. Those can be obtained directly from the moment generating function, which can be calculated straightforwardly from Equation (5) and (6) (see Appendix B). Equation (5) allows us to

rewrite the sigmoid function as

$$\sigma(z) = \frac{e^{\frac{z}{2}}}{2\cosh(\frac{z}{2})} = \int_0^\infty e^{f(\omega,z)} p_{\mathrm{PG}}(\omega|1,0) d\omega, \tag{7}$$

where we define

$$f(\omega, z) \doteq \frac{z}{2} - \frac{z^2}{2}\omega - \ln 2.$$

Setting $z = -g(\boldsymbol{x})$ in Equation (3) and substituting Equation (7) we get

$$\exp\left(-\int_{\mathcal{X}} \lambda\left(1 - \sigma(-g(\boldsymbol{x}))\right) d\boldsymbol{x}\right) = \exp\left(-\int_{\mathcal{X}\times\mathbb{R}^+} \left(1 - e^{f(\omega,-g(\boldsymbol{x}))}\right)\, p_{\mathrm{PG}}(\omega|1,0)\; \lambda d\omega d\boldsymbol{x}\right). \tag{8}$$

Finally, we apply Campbell's theorem (Equation (4)) to Equation (8). The space is a product space $\mathcal{Z} = \hat{\mathcal{X}} \doteq \mathcal{X} \times \mathbb{R}^+$ and the intensity $\Lambda(\boldsymbol{x}, \omega) = \lambda p_{\mathrm{PG}}(\omega|1,0)$. This results in the final representation of the exponential in Equation (8)

$$\exp\left(-\int_{\hat{\mathcal{X}}} \left(1 - e^{f(\omega,-g(\boldsymbol{x}))}\right) \Lambda(\boldsymbol{x},\omega)\; d\omega d\boldsymbol{x}\right) = \mathbb{E}_{P_\Lambda}\left[\prod_{(\boldsymbol{x},\omega)_m \in \Pi_{\hat{\mathcal{X}}}} e^{f(\omega_m,-g_m)}\right].$$

Interestingly, the new Poisson process $\Pi_{\hat{\mathcal{X}}}$ with measure $P_\Lambda$ has the form of a *marked* Poisson process (Kingman, 1993, chap. 5), where the latent Pólya-Gamma variables $\omega_m$ denote the 'marks' being independent random variables at each location $\boldsymbol{x}_m$. It is straightforward to sample such processes by first sampling the inhomogeneous Poisson process on domain $\mathcal{X}$ (for example by 'thinning' a process with constant rate (Lewis and Shedler, 1979; Adams et al., 2009)) and then drawing a mark $\omega$ on each event independently from the density $p_{\mathrm{PG}}(\omega|1,0)$.

Finally, using the Pólya–Gamma augmentation also for the discrete likelihood factors corresponding to the observed events in Equation (1) we obtain the following joint likelihood of the model

$$\begin{aligned} L(\mathcal{D}, \boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}|g,\lambda) &\doteq \frac{dP_{\mathrm{joint}}}{dP_{\mathrm{aug}}}(\mathcal{D}, \boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}|g,\lambda) \\ &= \prod_{(\boldsymbol{x},\omega)_m \in \Pi_{\hat{\mathcal{X}}}} e^{f(\omega_m,-g_m)} \prod_{n=1}^N \lambda e^{f(\omega_n,g_n)}, \end{aligned} \tag{9}$$

where we define the prior measure of augmented variables as $P_{\mathrm{aug}} = P_\Lambda \times P_{\boldsymbol{\omega}_N}$ and where $\boldsymbol{\omega}_N = \{\omega_n\}_{n=1}^N$ are the Pólya–Gamma variables for the observations $\mathcal{D}$ with the prior measure $dP_{\boldsymbol{\omega}_N} = \prod_{n=1}^N p(\omega_n|1,0)d\omega_n$. This augmented representation of the likelihood contains the function $g(\cdot)$ only linearly and quadratically in the exponents and is thus conjugate to the GP prior of $g(\cdot)$. Note that the original likelihood in Equation (1) can be recovered by $\mathbb{E}_{P_{\mathrm{aug}}}\left[L(\mathcal{D}, \boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}|g,\lambda)\right] = L(\mathcal{D}|g,\lambda)$.

## 3. Inference in the augmented space

Based on the augmentation we define a posterior density for the joint model with respect to the product measure $P_{\text{prior}} \times P_{\text{aug}}$

$$\begin{aligned} \boldsymbol{p}(\boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}, g, \lambda | \mathcal{D}) &\doteq \frac{dP_{\text{posterior}}}{d(P_{\text{prior}} \times P_{\text{aug}})}(\boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}, g, \lambda | \mathcal{D}) \\ &= \frac{L(\mathcal{D}, \boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}} | g, \lambda)}{L(\mathcal{D})}, \end{aligned} \tag{10}$$

where the denominator is the marginal likelihood $L(\mathcal{D}) = \mathbb{E}_{P_{\text{prior}} \times P_{\text{aug}}} \left[ L(\mathcal{D}, \boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}} | g, \lambda) \right]$. The posterior density of Equation (10) could be sampled using Gibbs sampling with explicit, tractable conditional densities. Similar to the variational approximation in the next section, one can show that the conditional measure of the point sets $\Pi_{\hat{\mathcal{X}}}$ and the variables $\boldsymbol{\omega}_N$, given the function $g(\cdot)$ and maximal intensity $\lambda$ is a product of a specific marked Poisson process and independent (tilted) Pólya–Gamma densities. On the other hand, the distribution over function $g(\cdot)$ conditioned on $\Pi_{\hat{\mathcal{X}}}$ and $\boldsymbol{\omega}_N$ is a Gaussian process. Note, however, one needs to sample this GP only at the finite points $\boldsymbol{x}_m$ in the random set $\Pi_{\hat{\mathcal{X}}}$ and the fixed set $\mathcal{D}$.

### 3.1 Variational mean–field approximation

For variational inference one assumes that the desired posterior probability measure belongs to a family of measures for which the inference problem is tractable. Here we make a simple structured mean field assumption in order to fully utilise its conjugate structure: We approximate the posterior measure by

$$P_{\text{posterior}}(\boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}, g, \lambda | \mathcal{D}) \approx Q_1(\boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}) \times Q_2(g, \lambda), \tag{11}$$

meaning that the dependencies between the Pólya–Gamma variables $\boldsymbol{\omega}_N$ and the marked Poisson process $\Pi_{\hat{\mathcal{X}}}$ on the one hand, and the function $g$ and the maximal intensity $\lambda$ on the other hand, are neglected. As we will see in the following, this simple mean–field assumption allows us to derive the posterior approximation analytically.

The variational approximation is optimised by minimising the Kullback–Leibler divergence between exact and approximated posteriors. This is equivalent to maximising the lower bound on the marginal likelihood of the observations

$$\mathcal{L}(\boldsymbol{q}) = \mathbb{E}_Q \left[ \log \left\{ \frac{L(\mathcal{D}, \boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}} | g, \lambda)}{\boldsymbol{q}_1(\boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}) \boldsymbol{q}_2(g, \lambda)} \right\} \right] \leq \log L(\mathcal{D}), \tag{12}$$

where $Q$ is the probability measure of the variational posterior in Equation (11) and we introduced approximate likelihoods

$$\boldsymbol{q}_1(\boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}) \doteq \frac{dQ_1}{dP_{\text{aug}}}(\boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}), \qquad \boldsymbol{q}_2(g, \lambda) \doteq \frac{dQ_2}{dP_{\text{prior}}}(g, \lambda).$$

Using standard arguments for mean field variational inference (Bishop, 2006, chap. 10) and Equation (11), one can then show that the optimal factors satisfy

$$\ln \boldsymbol{q}_1 \left( \boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}} \right) = \mathbb{E}_{Q_2} \left[ \log L(\mathcal{D}, \boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}} | g, \lambda) \right] + \text{const.} \tag{13}$$

and

$$\ln \boldsymbol{q}_2(g, \lambda) = \mathbb{E}_{Q_1}\left[\log L(\mathcal{D}, \boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}} | g, \lambda)\right] + \text{const.}\,, \tag{14}$$

respectively. These results lead to an iterative scheme for optimising $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ in order to increase the lower bound in Equation (12) in every step. From the structure of the likelihood one derives two further factorisations:

$$\boldsymbol{q}_1(\boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}) = \boldsymbol{q}_1(\boldsymbol{\omega}_N)\boldsymbol{q}_1(\Pi_{\hat{\mathcal{X}}}), \tag{15}$$

$$\boldsymbol{q}_2(g, \lambda) = \boldsymbol{q}_2(g)\boldsymbol{q}_2(\lambda), \tag{16}$$

where the densities are defined with respect to the measures $dP(\boldsymbol{\omega}_N)$, $dP_\Lambda$, $dP_{\text{GP}}$, and $p(\lambda)d\lambda$, respectively. The subsequent section describes these updates explicitly.

**Optimal Pólya–Gamma density** Following Equation (13) and (15) we obtain

$$\boldsymbol{q}_1(\boldsymbol{\omega}_N) = \prod_{n=1}^{N} \frac{\exp\left(-\frac{c_1^{(n)}}{2}\omega_n\right)}{\cosh^{-1}\left(c_1^{(n)}/2\right)} = \prod_{n=1}^{N} \frac{p_{\text{PG}}\left(\omega_n | 1, c_1^{(n)}\right)}{p_{\text{PG}}\left(\omega_n | 1, 0\right)},$$

where the factors are *tilts* of the prior Pólya-Gamma densities (see Equation (6) and Appendix B) with $c_1^{(n)} = \sqrt{\mathbb{E}_{Q_2}\left[g_n^2\right]}$. By simple density transformation we obtain the density with respect to the Lebesgue measure as

$$q_1(\boldsymbol{\omega}_N) = \boldsymbol{q}_1(\boldsymbol{\omega}_N)\left|\frac{dP_{\boldsymbol{\omega}_N}}{d\boldsymbol{\omega}_N}\right| = \prod_{n=1}^{N} p_{\text{PG}}\left(\omega_n | 1, c_1^{(n)}\right), \tag{17}$$

being a product of *tilted* Pólya–Gamma densities.

**Optimal Poisson process** Using Equation (13) and (15) we obtain

$$\boldsymbol{q}_1(\Pi_{\hat{\mathcal{X}}}) = \frac{\prod_{(\boldsymbol{x},\omega)_m \in \Pi_{\hat{\mathcal{X}}}} e^{\mathbb{E}_{Q_2}[f(\omega_m, -g_m)]}\lambda_1}{\exp\left(\int_{\hat{\mathcal{X}}}\left(e^{\mathbb{E}_{Q_2}[f(\omega, -g(\boldsymbol{x}))]} - 1\right)\lambda_1 p_{\text{PG}}(\omega|1,0)d\boldsymbol{x}d\omega\right)}, \tag{18}$$

with $\lambda_1 \doteq e^{\mathbb{E}_{Q_2}[\log \lambda^*]}$. Note, that $\mathbb{E}_{Q_2}\left[f(\omega_m, -g_m)\right]$ involves the expectations $\mathbb{E}_{Q_2}\left[g_m\right]$ and $\mathbb{E}_{Q_2}\left[(g_m)^2\right]$. One can show, that Equation (18) is again a marked Poisson process with intensity

$$\begin{aligned}\Lambda_1(\boldsymbol{x}, \omega) =& \lambda_1 \frac{\exp\left(-\frac{\mathbb{E}_{Q_2}[g(\boldsymbol{x})]}{2}\right)}{2\cosh\left(\frac{c_1(\boldsymbol{x})}{2}\right)} p_{\text{PG}}\left(\omega | 1, c_1(\boldsymbol{x})\right) \\ =& \lambda_1 \sigma(-c_1(\boldsymbol{x})) \exp\left(\frac{c_1(\boldsymbol{x}) - \mathbb{E}_{Q_2}\left[g(\boldsymbol{x})\right]}{2}\right) p_{\text{PG}}\left(\omega | 1, c_1(\boldsymbol{x})\right)\end{aligned} \tag{19}$$

where $c_1(\boldsymbol{x}) = \sqrt{\mathbb{E}_{Q_2}\left[g(\boldsymbol{x})^2\right]}$ (for a proof see Appendix D).

**Optimal Gaussian process** From Equation (14) and (16) we obtain the optimal approximation of the posterior likelihood (note that this is defined relative to GP prior)

$$\boldsymbol{q}_2(g) \propto e^{U(g)},$$

where the effective log–likelihood is given by

$$U(g) = \mathbb{E}_{Q_1}\left[\sum_{(\boldsymbol{x},\omega)_m \in \Pi_{\hat{\mathcal{X}}}} f(\omega_m, -g_m)\right] + \sum_{n=1}^{N} \mathbb{E}_{Q_1}\left[f(\omega_n, g(\boldsymbol{x}_n))\right].$$

The first expectation is over the variational Poisson process $\Pi_{\hat{\mathcal{X}}}$ and the second one over the Pólya–Gamma variables $\boldsymbol{\omega}_N$. These can be easily evaluated (see Appendix A) and one finds

$$U(g) = -\frac{1}{2}\int_{\mathcal{X}} A(\boldsymbol{x}) g(\boldsymbol{x})^2 d\boldsymbol{x} + \int_{\mathcal{X}} B(\boldsymbol{x}) g(\boldsymbol{x}) d\boldsymbol{x}, \tag{20}$$

with

$$\begin{aligned} A(\boldsymbol{x}) =& \sum_{n=1}^{N} \mathbb{E}_{Q_1}\left[\omega_n\right] \delta(\boldsymbol{x} - \boldsymbol{x}_n) + \int_0^\infty \omega \Lambda_1(\boldsymbol{x}, \omega) d\omega, \\ B(\boldsymbol{x}) =& \frac{1}{2}\sum_{n=1}^{N} \delta(\boldsymbol{x} - \boldsymbol{x}_n) - \frac{1}{2}\int_0^\infty \Lambda_1(\boldsymbol{x}, \omega) d\omega, \end{aligned}$$

where $\delta(\cdot)$ is the Dirac delta function. The expectations and integrals over $\omega$ are

$$\begin{aligned} &\mathbb{E}_{Q_1}\left[\omega_n\right] = \frac{1}{2c_1^{(n)}} \tanh\left(\frac{c_1^{(n)}}{2}\right), \\ &\int_0^\infty \Lambda_1(\boldsymbol{x}, \omega) d\omega = \lambda_1 \sigma(-c_1(\boldsymbol{x})) \exp\left(\frac{c_1(\boldsymbol{x}) - \mathbb{E}_{Q_2}\left[g(\boldsymbol{x})\right]}{2}\right) \doteq \Lambda_1(\boldsymbol{x}), \\ &\int_0^\infty \omega \Lambda_1(\boldsymbol{x}, \omega) d\omega = \frac{1}{2c_1(\boldsymbol{x})} \tanh\left(\frac{c_1(\boldsymbol{x})}{2}\right) \Lambda_1(\boldsymbol{x}). \end{aligned}$$

The resulting variational distribution defines a Gaussian process. Because of the mean–field assumption the integrals in Equation (20) do not require integration over random variables, but only solving two deterministic integrals over space $\mathcal{X}$. However, those integrals depend on function $g$ over the entire space and it is not possible for a general kernel to compute the marginal posterior density at an input $\boldsymbol{x}$ in closed form. For specific GP kernel operators, which are the inverses of differential operators, a solution in terms of linear partial differential equations would be possible. This could be of special interest for one–dimensional problems where Matern kernels with integer parameters (Rasmussen and Williams, 2006) fulfill this condition. Here, the problem becomes equivalent to inference for a (continuous time) Gaussian hidden Markov model and could be solved by performing a forward–backward algorithm (Solin, 2016). This would reduce the computations to the solution of ordinary differential equations. We will discuss details of such an approach elsewhere. To deal with general kernels we will resort instead to a the well known variational sparse GP approximation with inducing points.

**Optimal sparse Gaussian process** The sparse variational Gaussian approximation follows the standard approach (Csató and Opper, 2002; Csató, 2002; Titsias, 2009) and its generalisation to a continuum likelihood (Batz et al., 2018; de G. Matthews et al., 2016). For completeness, we repeat the derivation here and more detailed in Appendix E. We approximate $\boldsymbol{q}_2(g)$ by a sparse likelihood GP $\boldsymbol{q}_2^s(g)$ with respect to the GP prior

$$\frac{dQ_2^s}{dP}(g) = \boldsymbol{q}_2^s(\boldsymbol{g}_s), \tag{21}$$

which depends only on a finite dimensional vector of function values $\boldsymbol{g}_s = (g(\boldsymbol{x}_1), \dots, g(\boldsymbol{x}_L))^\top$ at a set of *inducing points* $\{\boldsymbol{x}_l\}_{l=1}^L$. With this approach it is again possible to marginalise out exactly all the infinitely many function values outside of the set of inducing points. The sparse likelihood $\boldsymbol{q}_2^s$ is optimised by minimising the Kullback–Leibler divergence

$$\mathrm{D}_{\mathrm{KL}}(Q_2^s \| Q_2) = \mathbb{E}_{Q_2^s}\left[\log \frac{\boldsymbol{q}_2^s(g)}{\boldsymbol{q}_2(g)}\right].$$

A short computation (Appendix E) shows that

$$q_2^s(\boldsymbol{g}_s) \propto e^{U^s(\boldsymbol{g}_s)} \qquad \text{with } U^s(\boldsymbol{g}_s) = \mathbb{E}_{P(g|\boldsymbol{g}_s)}\left[U(g)\right],$$

where the conditional expectation is with respect to the GP prior measure given the function $\boldsymbol{g}_s$ at the inducing points. The explicit calculation requires the conditional expectations of $g(\boldsymbol{x})$ and of $(g(\boldsymbol{x}))^2$. We get

$$\mathbb{E}_{P(g|\boldsymbol{g}_s)}\left[g(\boldsymbol{x})\right] = \boldsymbol{k}_s(\boldsymbol{x})^\top K_s^{-1} \boldsymbol{g}_s, \tag{22}$$

where $\boldsymbol{k}_s(\boldsymbol{x}) = (k(\boldsymbol{x}, \boldsymbol{x}_1), \dots, k(\boldsymbol{x}, \boldsymbol{x}_L))^\top$ and $K_s$ is the kernel matrix between inducing points. For the second expectation, we get

$$\mathbb{E}_{P(g|\boldsymbol{g}_s)}\left[g^2(\boldsymbol{x})\right] = \left(\mathbb{E}_{P(g|\boldsymbol{g}_s)}\left[g(\boldsymbol{x})\right]\right)^2 + \text{const.} \tag{23}$$

The constant equals the conditional variance of $g(\boldsymbol{x})$ which does not depend on the sparse set $\boldsymbol{g}_s$, but only on the locations of the sparse points. Because we are dealing now with a finite problem we can define the 'ordinary' posterior density of the GP at the inducing points with respect to the Lebesgue measure $d\boldsymbol{g}_s$. From Equation (20), (22), and (23), we conclude that the sparse posterior at the inducing variables is a multivariate Gaussian density

$$q_2^s(\boldsymbol{g}_s) = \mathcal{N}(\boldsymbol{\mu}_2^s, \Sigma_2^s), \tag{24}$$

with the covariance matrix given by

$$\Sigma_2^s = \left[K_s^{-1} \int_{\mathcal{X}} A(\boldsymbol{x}) \boldsymbol{k}_s(\boldsymbol{x}) \boldsymbol{k}_s(\boldsymbol{x})^\top d\boldsymbol{x}\; K_s^{-1} + K_s^{-1}\right]^{-1}, \tag{25}$$

and the mean

$$\boldsymbol{\mu}_2^s = \Sigma_2^s \left(K_s^{-1} \int_{\mathcal{X}} B(\boldsymbol{x}) \boldsymbol{k}_s(\boldsymbol{x}) d\boldsymbol{x}\right). \tag{26}$$

In contrast to other variational approximations (see for example (Lloyd et al., 2015; Hensman et al., 2015)) we obtain a closed analytic form of the variational posterior mean and

covariance which holds for arbitrary GP kernels. However, these results depend on finite dimensional integrals over the space $\mathcal{X}$ which cannot be computed analytically. This is different to the sparse approximation for the Poisson model with square link function (Lloyd et al., 2015), where similar integrals in the case of the squared exponential kernel can be obtained analytically. Hence, we resort to a simple Monte–Carlo integration, where *integration points* are sampled uniformly on $\mathcal{X}$ as

$$I_F = \int_{\mathcal{X}} F(\boldsymbol{x}) d\boldsymbol{x} \approx \frac{|\mathcal{X}|}{R} \sum_{r=1}^{R} F(\boldsymbol{x}_r).$$

The set of integration points $\{\boldsymbol{x}_r\}_{r=1}^R$ is drawn uniformly from the space $\mathcal{X}$.

Finally, from Equation (21) and (24) we obtain the mean function and the variance of the sparse approximation for every point $\boldsymbol{x} \in \mathcal{X}$, which is

$$\mu_2(\boldsymbol{x}) = \mathbb{E}_{Q_2}\left[g(\boldsymbol{x})\right] = \boldsymbol{k}_s(\boldsymbol{x})^\top K_s^{-1} \boldsymbol{\mu}_2^s, \tag{27}$$

and variance

$$\left(s_2(\boldsymbol{x})\right)^2 = k(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}_s(\boldsymbol{x})^\top K_s^{-1} \left(\mathbf{I} - \Sigma_2^s K_s^{-1}\right) \boldsymbol{k}_s(\boldsymbol{x}), \tag{28}$$

where $\mathbf{I}$ is the identity matrix.

**Optimal density for maximal intensity $\lambda$** From Equation (14) we identify the optimal density as a Gamma density

$$q_2(\lambda) = \mathrm{Gamma}(\lambda|\alpha_2, \beta_2) = \frac{\beta_2^{\alpha_2} (\lambda)^{\alpha_2 - 1} \mathrm{e}^{-\beta_2 \lambda}}{\Gamma(\alpha_2)}, \tag{29}$$

where $\alpha_2 = N + \mathbb{E}_{Q_1}\left[\mathbf{1}_\Pi(\boldsymbol{x})\right] + \alpha_0$, $\beta_2 = \beta_0 + \int_{\mathcal{X}} d\boldsymbol{x}$ and $\Gamma(\cdot)$ is the gamma function. $\mathbf{1}_\Pi(\boldsymbol{x})$ denotes the indicator function being 1 if $\boldsymbol{x} \in \Pi$ and 0 otherwise and the integral is again solved by Monte Carlo integration. This defines the required expectations for updating $q_1$ by $\mathbb{E}_{Q_2}\left[\lambda\right] = \frac{\alpha_2}{\beta_2}$ and $\mathbb{E}_{Q_2}\left[\log \lambda\right] = \psi(\alpha_2) - \log \beta_2$, where $\psi(\cdot)$ is the digamma function.

**Hyperparameters** Hyperparameters of the model are (i) the covariance parameters $\boldsymbol{\theta}$ of the GP, (ii) the locations of the inducing points $\{\boldsymbol{x}_l\}_{l=1}^L$, and (iii) the prior parameters $\alpha_0, \beta_0$ for the maximal intensity $\lambda$. The covariance parameters (i) $\boldsymbol{\theta}$ are optimised by gradient ascent following the gradient of the lower bound in Equation (12) with respect to $\boldsymbol{\theta}$ (Appendix F). As gradient ascent algorithm we employ the ADAM algorithm (Kingma and Ba, 2014). We perform always one step after the variational posterior $q$ is updated as described before. (ii) The locations of the sparse GP $\{\boldsymbol{x}_l\}_{l=1}^L$ could in principle be optimised as well, but we keep them fixed and position them on a regular grid over the space $\mathcal{X}$. From this choice it follows that $K_s$ is a Toeplitz matrix, when the kernel is translationally invariant. This could be inverted in $\mathcal{O}(L(\log L)^2)$ instead of $\mathcal{O}(L^3)$ operations (Press et al., 2007) but we do not employ this fact. Finally, (iii) the value for prior parameters $\alpha_0$ and $\beta_0$ are chosen such that $p(\lambda)$ has a mean twice and standard deviation once the intensity one would expect for a homogeneous Poisson Process observing $\mathcal{D}$. The complete variational procedure is outlined in Algorithm 1.

**Algorithm 1:** Variational Bayes algorithm for sigmoidal Gaussian Cox process.

**Init:** $\mathbb{E}_Q\left[g(\boldsymbol{x})\right], \mathbb{E}_Q\left[(g(\boldsymbol{x}))^2\right]$ at $\mathcal{D}$ and integration points, and $\mathbb{E}_Q\left[\lambda\right], \mathbb{E}_Q\left[\log\lambda\right]$
**while** $\mathcal{L}$ *not converged* **do**
  **Update** $q_1$
    **PG distributions at observations**: $q_1(\boldsymbol{\omega}_N)$ with Eq. (17)
    **Rate of latent process**: $\Lambda_1(\boldsymbol{x}, \omega)$ at integration points with Eq. (19)
  **Update** $q_2$
    **Sparse GP distribution**: $\Sigma_2^s, \boldsymbol{\mu}_2^s$ with Eq. (25), (26)
    **GP at $\mathcal{D}$ and integration points**: $\mathbb{E}_{Q_2}\left[g(\boldsymbol{x})\right], \mathbb{E}_{Q_2}\left[(g(\boldsymbol{x}))^2\right]$ with Eq. (27), (28)
    **Gamma-distribution of $\lambda$**: $\alpha_2, \beta_2$ with Eq. (29)
  **Update kernel parameters with gradient ascent**
**end**

### 3.2 Laplace approximation

In this section we will show that our variable augmentation method is well suited for computing a Laplace approximation (Bishop, 2006, chap. 4) to the joint posterior of the GP function $g(\cdot)$ and the maximal intensity $\lambda$ as an alternative to the previous variational scheme. To do so we need the maximum a posteriori (MAP) estimate (equal to the mode of the posterior distribution) and a second order Taylor expansion around this mode. The augmentation method will be used to compute the MAP estimator iteratively using an EM algorithm.

**Obtaining the MAP estimate** In general, a proper definition of the posterior mode would be necessary, because the GP posterior is over a space of functions, which is an infinite dimensional object and does not have a density with respect to Lebesgue measure. A possibility to avoid this problem would be to discretise the spatial integral in the likelihood and to approximate the posterior by a multivariate Gaussian density for which the mode can then be computed by setting the gradient equal to zero. In this paper, we will use a different approach which defines the mode directly in function space and allows us to utilise the sparse GP approximation developed previously for the computations. A mathematically proper way would be to derive the MAP estimator by maximising a properly penalised log–likelihood. As discussed e.g. in Rasmussen and Williams (2006, chap. 6) for GP models with likelihoods which depend on finitely many inputs only, this penalty is given by the squared reproducing kernel Hilbert space (RKHS) norm that corresponds to the GP kernel. Hence, we would have

$$(g^*, \lambda^*) = \operatorname{argmin}_{g\in\mathcal{H}_k,\lambda}\left\{-\ln L(\mathcal{D}|g,\lambda) - \ln p(\lambda) + \frac{1}{2}\|g\|^2_{\mathcal{H}_k}\right\},$$

where $\|g\|^2_{\mathcal{H}_k}$ is the RKHS norm for the kernel $k$. This penalty term can be understood as a proper generalisation of a Gaussian log–prior density to function space. We will not give a formal definition here but work on a more heuristic level in the following. Rather than attempting a direct optimisation, we will use an EM algorithm instead, applying the

variable augmentation with the Poisson process and Pólya–Gamma variables introduced in the previous sections. In this case, the likelihood part of the resulting '$\mathcal{Q}$–function'

$$\mathcal{Q}((g,\lambda)|(g,\lambda)^{\text{old}}) \doteq \mathbb{E}_{P(\boldsymbol{\omega}_N,\Pi_{\hat{\mathcal{X}}}|(g,\lambda)^{\text{old}})}\left[\ln L(\mathcal{D},\boldsymbol{\omega}_N,\Pi_{\hat{\mathcal{X}}}|g,\lambda)\right] + \ln p(\lambda) - \frac{1}{2}\|g\|^2_{\mathcal{H}_k}, \quad (30)$$

that needs to be maximised in the M–step becomes (as in the variational approach before) the likelihood of *a Gaussian model* in the GP function $g$. Hence, we can argue that the function $g$ which maximises $\mathcal{Q}$ is equal to the *posterior mean* of the resulting Gaussian model and can be computed without discussing the explicit form of the RKHS norm.

The conditional probability measure $P(\boldsymbol{\omega}_N,\Pi_{\hat{\mathcal{X}}}|(g,\lambda)^{\text{old}})$ is easily obtained similar to the optimal measure $Q_1$ by not averaging over $g$ and $\lambda$. This gives us straightforwardly the density

$$\boldsymbol{p}(\boldsymbol{\omega}_N,\Pi_{\hat{\mathcal{X}}}|(g,\lambda)^{\text{old}}) = \boldsymbol{p}(\boldsymbol{\omega}_N|(g,\lambda)^{\text{old}})\boldsymbol{p}(\Pi_{\hat{\mathcal{X}}}|(g,\lambda)^{\text{old}}).$$

The first factor is

$$p(\boldsymbol{\omega}_N|(g,\lambda)^{\text{old}}) = \boldsymbol{p}(\boldsymbol{\omega}_N|(g,\lambda)^{\text{old}})\left|\frac{dP_{\boldsymbol{\omega}_N}}{d\boldsymbol{\omega}_N}\right| = \prod_{n=1}^{N} p_{\text{PG}}\left(\omega_n|1,\tilde{c}_n\right),$$

with $\tilde{c}_n = |g_n^{\text{old}}|$. The latent point process $\Pi_{\hat{\mathcal{X}}}$ is again a Poisson process density

$$\boldsymbol{p}(\Pi_{\hat{\mathcal{X}}}|(g,\lambda)^{\text{old}}) = \frac{dP_{\tilde{\Lambda}}}{dP_{\Lambda}}(\Pi_{\hat{\mathcal{X}}}|(g,\lambda)^{\text{old}}),$$

where the intensity is

$$\tilde{\Lambda}(\boldsymbol{x},\omega) = \lambda^{\text{old}}\sigma(-g^{\text{old}}(\boldsymbol{x}))p_{\text{PG}}\left(\omega|1,\tilde{c}(\boldsymbol{x})\right),$$

with $\tilde{c}(\boldsymbol{x}) = |g^{\text{old}}(\boldsymbol{x})|$. The first term in the $\mathcal{Q}$–function is

$$\begin{aligned} U(g,\lambda) &\doteq \mathbb{E}_{P(\boldsymbol{\omega}_N,\Pi_{\hat{\mathcal{X}}}|(g,\lambda)^{\text{old}})}\left[\ln L(\mathcal{D},\boldsymbol{\omega}_N,\Pi_{\hat{\mathcal{X}}}|g,\lambda)\right] \\ &= -\frac{1}{2}\int_{\mathcal{X}} \tilde{A}(\boldsymbol{x})g(\boldsymbol{x})^2 d\boldsymbol{x} + \int_{\mathcal{X}} \tilde{B}(\boldsymbol{x})g(\boldsymbol{x})d\boldsymbol{x}, \end{aligned}$$

with

$$\begin{aligned} \tilde{A}(\boldsymbol{x}) &= \sum_{n=1}^{N} \mathbb{E}_{P(\omega_n|(g,\lambda)^{\text{old}})}\left[\omega_n\right]\delta(\boldsymbol{x}-\boldsymbol{x}_n) + \int_0^{\infty} \mathbb{E}_{P(\omega|(g,\lambda)^{\text{old}})}\left[\omega\right]\tilde{\Lambda}(\boldsymbol{x},\omega)d\omega, \\ \tilde{B}(\boldsymbol{x}) &= \frac{1}{2}\sum_{n=1}^{N}\delta(\boldsymbol{x}-\boldsymbol{x}_n) - \frac{1}{2}\int_0^{\infty}\tilde{\Lambda}(\boldsymbol{x},\omega)d\omega. \end{aligned}$$

We have already tackled almost identical log–likelihood expressions in Section 3.1 (see Equation (20)). While for specific priors (with precision kernels given by differential operators) an exact treatment in terms of solutions of ODEs or PDEs is possible, we will again resort to the sparse GP approximation instead. The sparse version $U^s(\boldsymbol{g}_s,\lambda)$ is obtained by replacing $g(\boldsymbol{x}) \to \mathbb{E}_{P(g|\boldsymbol{g}_s)}\left[g(\boldsymbol{x})\right]$ in $U(g,\lambda)$. From this we obtain the sparse $\mathcal{Q}$–function as

$$\mathcal{Q}^s((\boldsymbol{g}_s,\lambda)|(\boldsymbol{g}_s,\lambda)^{\text{old}}) \doteq U^s(\boldsymbol{g}_s,\lambda) + \ln p(\lambda) - \frac{1}{2}\boldsymbol{g}_s^{\top}K_s^{-1}\boldsymbol{g}_s. \quad (31)$$

The function values $\boldsymbol{g}_s$ and the maximal intensity $\lambda$ that maximise Equation (31) can be found analytically by solving

$$\frac{\partial \mathcal{Q}^s}{\partial \boldsymbol{g}_s} = \mathbf{0} \text{ and } \frac{\partial \mathcal{Q}^s}{\partial \lambda} = 0.$$

The final MAP estimate is obtained after convergence of the EM algorithm and the desired sparse MAP solution for $g(x)$ is given by (see Equation (27))

$$g_{MAP}(\boldsymbol{x}) = \boldsymbol{k}_s(\boldsymbol{x})^\top K_s^{-1} \boldsymbol{g}^s$$

As for the variational scheme, integrals over the space $\mathcal{X}$ are approximated by Monte–Carlo integration. An alternative derivation of the sparse MAP solution can be based on restricting the minimisation of (30) to functions which are linear combinations of kernels centred at the inducing points and using the definition of the RKHS norm (see (Rasmussen and Williams, 2006, chap. 6)).

**Sparse Laplace posterior** To complete the computation of the Laplace approximation, we need to evaluate the quadratic fluctuations around the MAP solution. We will also do this with the previously obtained sparse approximation. The idea is that from the converged MAP solution, we define a sparse likelihood of the Poisson model via the replacement

$$L^s(\boldsymbol{g}_s, \lambda) \doteq L(\mathcal{D} | \mathbb{E}_{P(g|\boldsymbol{g}_s)}[g], \lambda)$$

For this sparse likelihood it is easy to compute the Laplace posterior using second derivatives. Here, the change of variables $\rho = \ln \lambda$ will be made to ensure that $\lambda > 0$. This results in an effective log–normal density over the maximal intensity rate $\lambda$. While we do not address hyperparameter selection for the Laplace posterior in this work, a straightforward approach, as suggested by Flaxman et al. (2017), could be to use cross validation to optimise the kernel parameters while finding the MAP estimate or to use the Laplace approximation to approximate the evidence. As in the variational case the inducing point locations $\{\boldsymbol{x}_l\}_{l=1}^L$ will be on a regular grid over space $\mathcal{X}$.

Note that for the Laplace approximation, the augmentation scheme is only used to compute the MAP estimate in an efficient way. There are no further mean–field approximations involved. This also implies, that dependencies between $\boldsymbol{g}_s$ and $\lambda$ are retained.

### 3.3 Predictive density

Both variational and Laplace approximation yield a posterior distribution $q$ over $\boldsymbol{g}_s$ and $\lambda$. The GP approximation at any given points in $\mathcal{X}$ is given by

$$q(g(\boldsymbol{x})) = \int \int p(g(\boldsymbol{x}) | \boldsymbol{g}_s) q(\boldsymbol{g}_s, \lambda) \, d\boldsymbol{g}_s d\lambda,$$

which for both methods results in a normal density. To find the posterior mean of the intensity function at a point $\boldsymbol{x} \in \mathcal{X}$ one needs to compute

$$\mathbb{E}_Q[\Lambda(\boldsymbol{x})] = \mathbb{E}_Q\left[\lambda \int_{-\infty}^{\infty} \sigma(g(\boldsymbol{x}))\right].$$

For variational and Laplace posterior the expectation over $\lambda$ can be computed analytically, leaving the expectation over $g(\boldsymbol{x})$, which is computed numerically via quadrature methods. To evaluate the performance of inference results we are interested in computing the likelihood on test data $\mathcal{D}_{\text{test}}$, generated from the ground truth. We will consider two methods:

Sampling GPs $g$ from the posterior we calculate the (log) mean of the test likelihood

$$\begin{aligned}\ell(\mathcal{D}_{\text{test}}) &= \ln \mathbb{E}_P \left[L(\mathcal{D}_{\text{test}}|\Lambda)|\mathcal{D}\right] \approx \ln \mathbb{E}_Q \left[L(\mathcal{D}_{\text{test}}|\Lambda)\right] \\ &= \ln \mathbb{E}_Q \left[\exp\left(-\int_{\mathcal{X}} \lambda\sigma(g(\boldsymbol{x}))d\boldsymbol{x}\right) \prod_{\boldsymbol{x}_n \in \mathcal{D}_{\text{test}}} \lambda\sigma(g(\boldsymbol{x}_n))\right]\end{aligned} \tag{32}$$

where the integral in the exponent is approximated by Monte–Carlo integration. The expectation is approximated by averaging over $2 \times 10^3$ samples from the inferred posterior $Q$ of $\lambda$ and $g$ at the observations of $\mathcal{D}_{\text{test}}$ and the integration points.
Instead of sampling one can also obtain an analytic approximation for the log test likelihood in Equation (32) by a second order Taylor expansion around the mean of the obtained posterior. Applying this idea to the variational mean field posterior we get

$$\begin{aligned}\ell(\mathcal{D}_{\text{test}}) \approx \ln L(\mathcal{D}_{\text{test}}|\Lambda_Q) &+ \frac{1}{2}\mathbb{E}_Q\left[(\boldsymbol{g}_s - \boldsymbol{\mu}_2^s)^\top \mathbf{H}_{\boldsymbol{g}_s}\big|_{\Lambda_Q} (\boldsymbol{g}_s - \boldsymbol{\mu}_2^s)\right] \\ &+ \frac{1}{2} H_\lambda|_{\Lambda_Q} \operatorname{Var}_Q(\lambda),\end{aligned} \tag{33}$$

where $\Lambda_Q(\boldsymbol{x}) = \mathbb{E}_Q[\lambda]\,\sigma(\mathbb{E}_Q[g(\boldsymbol{x})])$ and $\mathbf{H}_{\boldsymbol{g}_s}\big|_{\Lambda_Q}$, $H_\lambda|_{\Lambda_Q}$ are the second order derivative of the likelihood in Equation (1) with respect to $\boldsymbol{g}_s$ and $\lambda$ at $\Lambda_Q$. While an approximation only involving the first term would neglect the uncertainties in the posterior (as done by John and Hensman (2018)), the second and third term take these into account.

## 4. Results

**Generating data from the model** To evaluate the two newly developed algorithms we generate data according to the sigmoidal Gaussian Cox process model

$$\begin{aligned} g &\sim \quad \boldsymbol{p}_{\text{GP}}(\cdot|0,k), \\ \mathcal{D} &\sim \quad \boldsymbol{p}_\Lambda(\cdot), \end{aligned}$$

where $\boldsymbol{p}_\Lambda(\cdot)$ is the Poisson process density over sets of point with $\Lambda(\boldsymbol{x}) = \lambda\sigma(g(\boldsymbol{x}))$ and $\boldsymbol{p}_{\text{GP}}(\cdot|0,k)$ is a GP density with mean 0 and covariance function $k$. As kernel we choose a squared exponential function

$$k(\boldsymbol{x},\boldsymbol{x}') = \theta \prod_{i=1}^{d} \exp\left(-\frac{(x_i - x_i')^2}{2\nu_i^2}\right),$$

where the hyperparameters are scalar $\theta$ and length scales $\boldsymbol{\nu} = (\nu_1,\ldots,\nu_d)^\top$. Sampling of the inhomogeneous Poisson process is done via *thinning* (Lewis and Shedler, 1979; Adams et al., 2009). We assume that hyperparameters are known for subsequent experiments with data sampled from the generative model.

**Benchmarks for sigmoidal Gaussian Cox process inference** We compare the proposed algorithms to two alternative inference methods for the sigmoidal Gaussian Cox process model. As an exact inference method we use the sampling approach of Adams et al. (2009)[4]. In terms of speed, a competitor is a different variational approach given by Hensman et al. (2015) who proposed to discretise space $\mathcal{X}$ in several regular bins with size $\Delta$. Then the likelihood in Equation (1) is approximated by

$$L(\mathcal{D}|\lambda\sigma(g(\boldsymbol{x}))) \approx \prod_i p_{\text{po}}(n_i|\lambda\sigma(g(\boldsymbol{x}_i))\Delta),$$

where $p_{\text{po}}$ is the Poisson distribution conditioned on the mean parameter, $\boldsymbol{x}_i$ is the centre of bin $i$, and $n_i$ the number of observations within this bin. Using a (sparse) Gaussian variational approximation the corresponding Kullback–Leibler divergence is minimised by gradient ascent to find the optimal posterior over the GP $g$ and a point estimate for $\lambda$. This method was originally proposed for the log Cox-process ($\Lambda(\boldsymbol{x}) = e^{g(\boldsymbol{x})}$), but with the elegant GPflow package (Matthews et al., 2017) implementation of the scaled sigmoid link function is straightforward. It should be noted, that this method requires numerical integration over the sigmoid link function to evaluate the variational lower bound at every spatial bin and every gradient step, since it does not make use of our augmentation scheme (see Section 5 for discussion, how the proposed augmentation can be used for this model). We refer to this inference algorithm as 'variational Gauss'. To have fair comparison between the different methods, the inducing points for all algorithms (except for the sampler) are equal and the number of bins used to discretise the domain $\mathcal{X}$ for the variational Gauss algorithm is set equal to the number of integration points used for the MC integration in the variational mean field and the Laplace method.

**Experiments on data from generative model** As an illustrative example we sample a one dimensional Poisson process with the generative model and perform inference with the sampler ($2 \times 10^3$ samples after $10^3$ burn-in iterations), the mean field algorithm, the Laplace approximation and the variational Gauss. In Figure 1 **(a)**–**(d)** the different posterior mean intensity functions with their standard deviations are shown. For **(b)**–**(d)** 50 regularly spaced inducing points are used. For **(b)**–**(c)** $2 \times 10^3$ random integration points are drawn uniformly over the space $\mathcal{X}$, while for **(d)** $\mathcal{X}$ is discretised into the same number of bins. All algorithms recover the true intensity well. The mean field and the Laplace algorithm show smaller posterior variance compared to the sampler. The fastest inference result is obtained by the Laplace algorithm in 0.02 s, followed by the mean field (0.09), variational Gauss (80) and the sampler ($1.8 \times 10^3$). The fast convergence of the Laplace and the variational mean field algorithm is illustrated in Figure 1 **(e)**, where objective functions of our two algorithms (minus the maximum they converged to) is shown as a function of run time. Both algorithms reach a plateau in only a few ($\sim 6$) iterations. To compare performance in terms of log expected test likelihood $\ell_{\text{test}}$ (test sets $\mathcal{D}_{\text{test}}$ sampled from the ground truth), we averaged results over ten independent datasets. The posterior of the sampler yields the highest value with 875.5, while variational ($\ell_{\text{test}} = 686.2$, approximation by Equation (33) yields 686.5), variational Gauss (686.7) and Laplace (686.1) yield all similar results (see also Figure 4 **(a)**). The posterior density of the maximal intensity $\lambda$ is shown in Figure 1 **(f)**.

4. To increase efficiency, the GP values $g$ are sampled by elliptical slice sampling (Murray et al., 2010).

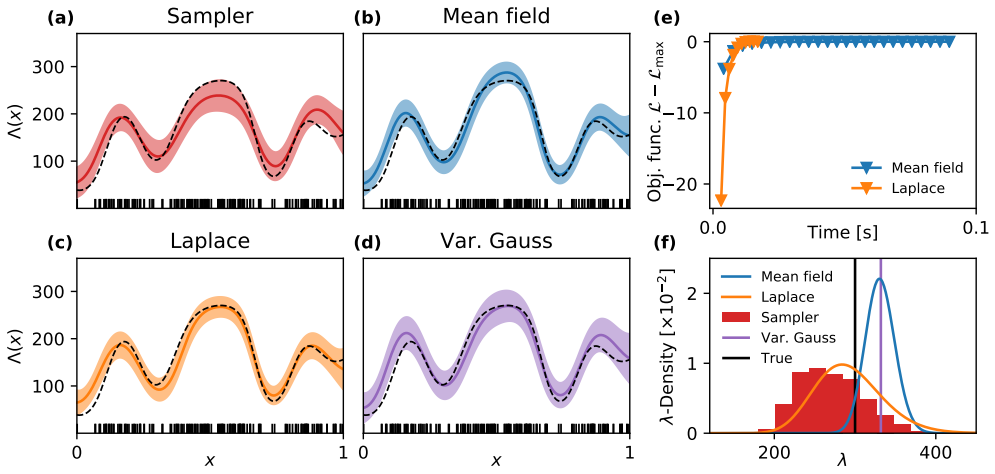



Figure 1: **Inference on 1D dataset.** **(a)**–**(d)** Inference result for sampler, mean field algorithm, Laplace approximation, and variational Gauss. Solid coloured lines denote the mean intensity function, shaded areas mean ± standard deviation, and dashed black lines the true rate functions. Vertical bars are observations $\mathcal{D}$. **(e)** Convergence of mean field and EM algorithm. Objective functions (Lower bound for mean–field and log likelihood for EM algorithm, shifted such that convergence is at 0) as function of run time (triangle marks one finished iteration of the respective algorithm). **(f)** Inferred posterior densities over the maximal intensity $\lambda$. Variational Gauss provides only a point estimate. Black vertical bar denotes the true $\lambda$.

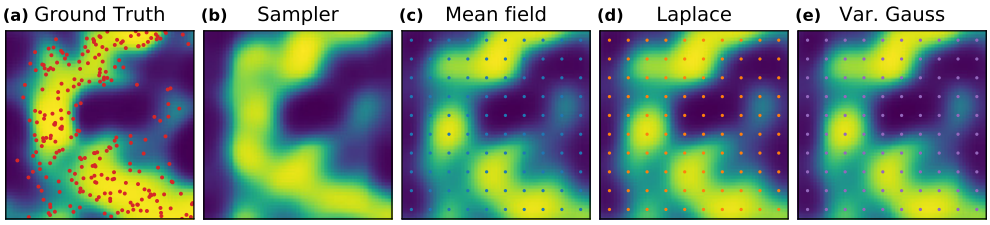



Figure 2: **Inference on 2D dataset. (a)** Ground truth intensity function $\Lambda(\boldsymbol{x})$ with observed dataset $\mathcal{D}$ (red dots).**(b)**–**(e)** Mean posterior intensity of the sampler, mean field algorithm, Laplace, and variational Gauss are shown. 100 inducing points on a regular grid (shown as coloured points) and 2500 integration points/bins are used.

In Figure 2 we show inference results for a two dimensional Cox process example. $10 \times 10$ inducing points and 2500 integration points/bins are used for mean field, Laplace and variational Gauss algorithm. The posterior mean of sampler **(b)**, of the mean field **(c)**, of the Laplace **(d)** and of the variational Gauss algorithm **(e)** recover the true intensity rate $\Lambda(\boldsymbol{x})$ **(a)** well.

To evaluate the role of the number of inducing points and number of integration points we generate 10 test sets $\mathcal{D}_{\text{test}}$ from a process with the same intensity as in Figure 2**(a)**. We evaluate the log expected likelihood (Equation (32)) on these test sets and compute the average. The result is shown for different numbers of inducing points (Figure 3**(a)** with 2500 integration points) and different numbers of integration points (Figure 3**(b)** with $10 \times 10$ inducing points). To account for randomness of integration points the fitting is repeated five times and the shaded area is between the minimum and maximum obtained by these fits. For all approximate algorithms the log predictive test likelihood saturates already for few inducing points ($\approx 49$ $(7 \times 7)$) of the sparse GP. However, as expected, the inference approximations are slightly inferior to the sampler. The log expected test likelihood is hardly affected by the number of integration points as shown in Figure 3 **(b)**. Also the approximated test likelihood for the mean field algorithm in Equation (33) yields good estimates of the sampled value (dashed line in **(a)** and **(b)**). In terms of runtime (Figure 4 **(c)**–**(d)**) the mean field algorithm and the Laplace approximation are superior by more than one order of magnitude to the variational Gauss algorithm for this particular example. Difference increases with increasing number of inducing points.

In Figure 4 the four algorithms are compared on five different datasets sampled from the generative model. As we observed for the previous examples the three different approximating algorithms yield qualitatively similar performance in terms of log test likelihood $\ell_{\text{test}}$, but the sampler is superior. Again the approximated test likelihood in Equation (33) (blue star) provides good estimate of the sampled value. In addition we provide the approximated root mean squared error (RMSE, evaluated on a fine grid and normalised by maximal intensity $\lambda$) between inferred mean and ground truth. In terms of run time the mean field and Laplace algorithm are by at least on order of magnitude faster than the variational

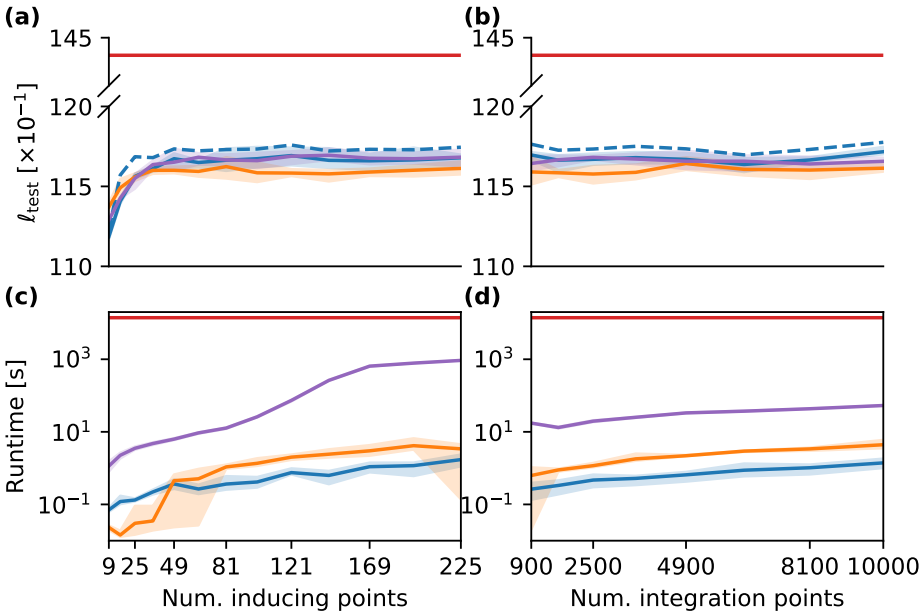



Figure 3: **Evaluation of inference. (a)** The log expected predictive likelihood averaged over ten test sets as a function of the number of inducing points. Number of integration points/bins is fixed to 2500. Results for sampler in (red), mean field (blue), Laplace (orange), and variational Gauss (purple) algorithm. Solid line denotes mean over five fits (same data), and shaded area denotes min. and max. result. Dashed blue line shows the approximated log expected predictive likelihood for the mean field algorithm. **(b)** Same as (a), but as function of number of integration points. Number of inducing points is fixed to $10 \times 10$. Below: Run time of the different algorithms as function of number of inducing points **(c)** and number of integration points **(d)**. Data are the same as in Figure 2.

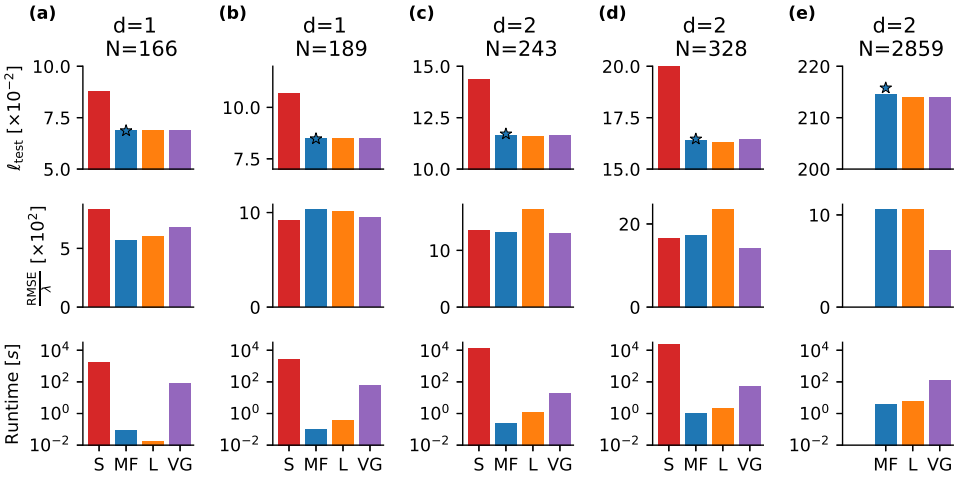



Figure 4: **Performance on different artificial datasets.** The sampler (S), the mean field algorithm (MF), the Laplace (L), and variational Gauss (VG) are compared on five different datasets with $d$–dimensions and $N$ observations (one column corresponds to one dataset). Top row: Log expected test likelihood of the different inference results. The star denotes the approximated test likelihood of the variational algorithm. Center row: The approximated root mean squared error (normalised by true maximal intensity rate $\lambda$). Bottom row: Run time in seconds. The dataset **(e)** is intractable for the sampler due to the many observations. Data in Figure 1 and 2 correspond to **(a)** and **(c)**.

Gauss algorithm. In general, the mean–field algorithm seems to be slightly faster than the Laplace.

**General datasets and comparison to the approach of Lloyd et al.** Next, we test our variational mean field algorithm on datasets not coming from the generative model. On such datasets we do not know, whether our model provides a good prior. As discussed previously an alternative model was proposed by Lloyd et al. (2015) making use of the link function $\Lambda(\boldsymbol{x}) = g^2(\boldsymbol{x})$. While the sigmoidal Gaussian Cox process with the proposed augmentation scheme has analytic updates for the variational posterior, in case of the squared Gaussian Cox process the likelihood integral can be solved analytically and does not need to be sampled (if the kernel is a squared exponential and the domain is rectangular). Both algorithms rely on the sparse GP approximation. To compare the two methods empirically first we consider one dimensional data generated using a known intensity function. We choose $\Lambda(x) = 2\exp(-x/15) + \exp(-(x-25)^2/100)$ on an interval $[0, 50]$ already proposed by Adams et al. (2009). We generate three training and test sets, where we scale this rate function by factors of 1, 10, and 100 and fit the sigmoidal and squared Gaussian Cox

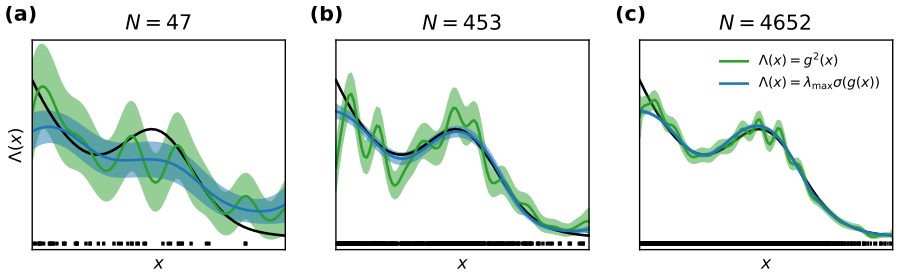



Figure 5: **1D example.** Observations (black bars) are sampled from the same function (black line) scaled by **(a)** 1, **(b)** 10, and **(c)** 100. Blue and green line show the mean posterior of the sigmoidal and squared Gaussian Cox process, respectively. Shaded area denotes mean ± standard deviation.

| | $\Lambda(x) = \lambda_{\max}\sigma(g(x))$ | | | $\Lambda(x) = g^2(x)$ | | |
|---|---|---|---|---|---|---|
| $N$ | Runtime [s] | RMSE | $\ell_{\text{test}}$ | Runtime [s] | RMSE | $\ell_{\text{test}}$ |
| 47 | $0.27 \pm 0.30$ | $0.24 \pm 0.02$ | $-43.43 \pm 0.42$ | $0.41 \pm 0.05$ | 0.24 | $-44.26 \pm 0.09$ |
| 453 | $0.50 \pm 0.04$ | $0.97 \pm 0.13$ | $720.81 \pm 0.28$ | $0.23 \pm 0.05$ | 2.11 | $710.43 \pm 1.38$ |
| 4652 | $0.41 \pm 0.01$ | $7.68 \pm 0.75$ | $17497.31 \pm 2.13$ | $0.79 \pm 0.09$ | 8.16 | $17496.75 \pm 1.65$ |

Table 1: **Benchmarks for Figure 5** The mean and standard deviation of runtime, RMSE, and log expected test likelihood for Figure 5**(a)**–**(c)** obtained from 5 fits. Note that the RMSE for $\Lambda(\boldsymbol{x}) = g^2(\boldsymbol{x})$ has no standard deviation, because the inference algorithm is deterministic.

process with their corresponding variational algorithm to each training set[5]. The number of inducing points is 40 in this example. For our variational mean field algorithm we used 5000 integration points. The posterior intensity $\Lambda(\boldsymbol{x})$ for the three datasets can be seen in Figure 5. The model with the sigmoidal link function infers smoother posterior functions with smaller variance compared to the posterior with the squared link function. For datasets shown in Figure 5 we run the fits five times and report mean and standard deviation of runtime, RMSE and log expected test likelihood $\ell_{\text{test}}$ in Table 1. Run times of the two algorithms are comparable, where for the intermediate dataset the algorithm with the squared link function is faster while for the largest data set the one with the sigmoidal link function converges first. RMSE and $\ell_{\text{test}}$ are also comparable except for the intermediate dataset, where the sigmoidal model is the superior one.

Next we deal with two real world two dimensional datasets for comparison. The first one is neuronal data, where spiking activity was recorded from a mouse, that was freely moving in an arena (For The Biology Of Memory and Sargolini, 2014; Sargolini et al., 2006). Here we consider as data $\mathcal{D}$ the position of the mouse when the recorded cell fired and the observations are randomly assigned to either training or test set. In Figure 6 **(a)**

5. We thank Chris Lloyd and Tom Gunter for providing the code for inferring the variational posterior of the squared Gaussian Cox process.

the observations in the training set ($N = 583$) are shown. In Figure 6 **(b)** and **(c)** the variational posterior's mean intensity $\Lambda(\boldsymbol{x})$ is shown obtained for the sigmoidal and the squared link function, respectively, inferred with a regular grid of $20 \times 20$ inducing points. As in Figure 5 we see that the sigmoidal posterior is the smoother one. The major difference between the two algorithms (apart from the link function) is the fact that for the sigmoidal model we are required to sample an interval over the space. We investigate the effect of the number of integration points in terms of runtime[6] and log expected test likelihood in Figure 6 **(d)**. First, we observe regardless of the number of integration points that the variational posterior of the squared link function yields the superior expected test likelihood. For the sigmoidal model the test likelihood does not improve significantly with more integration points. Runtimes of both algorithms are comparable, when 5000 integration points are chosen. A speed up for our mean field algorithm is achieved by first fitting the model with 1000 integration points and once converged, redrawing the desired number of integration points and rerun the algorithm (dotted line in Figure 6**(d)**). This method allows for a significant speed up without loss in terms of test likelihood $\ell_{\text{test}}$. The variational mean-field algorithm with the sigmoid link function is faster with up to 5000 integration points and equally fast with 10000 integration points.

As second dataset we consider the Porto taxi dataset (Moreira-Matias et al., 2013). These data contain trajectories of taxi travels from the years 2013/14 in the city of Porto. As John and Hensman (2018) we consider the pick-ups as observations of a Poisson process[7]. We consider 20000 taxi rides randomly split into training and test set ($N = 10017$ and $N = 9983$, respectively). The training set is shown in Figure 6**(e)**. Inducing points are positioned on a regular grid of $20 \times 20$. The variational posterior mean of the respective intensity is shown in Figure 6 **(f)** and **(g)**. With as many data points as in these data the differences between the two models are more subtle as compared to **(b)** and **(c)**. In terms of test likelihood $\ell_{\text{test}}$ the variational posterior of the sigmoidal model (with $\geq 2000$ integration points) outperforms the model with squared link function (Figure 6 **(h)**). For similar test likelihoods $\ell_{\text{test}}$ our variational algorithm is $\sim 2\times$ faster than the variational posterior with squared link function. The results show that the choice of number of integration points reduces to the question of speed vs accuracy trade–off. As for the previous dataset, the strategy of first fitting the posterior with 1000 integration points and then with the desired number of integration points (dotted line) proves that we can get a significant speed up without loosing predictive power.

## 5. Discussion and Outlook

Using a combination of two known variable augmentation methods, we derive a conjugate representation for the posterior measure of a sigmoidal Gaussian Cox process. The approximation of the augmented posterior by a simple mean field factorisation yields an efficient variational algorithm. The rationale behind this method is that the variational updates in the conjugate model are explicit and analytical and do not require (black–box) gradient

6. Note, that - in contrast to Figures 3 and 4 - the runtime is displayed on linear scale, meaning both algorithms are of same order of magnitude.
7. As John and Hensman (2018) report some regions to be highly peaked we consider only pickups happening within the coordinates $(41.147, -8.58)$ and $(41.18, -8.65)$ in order to exclude those regions.

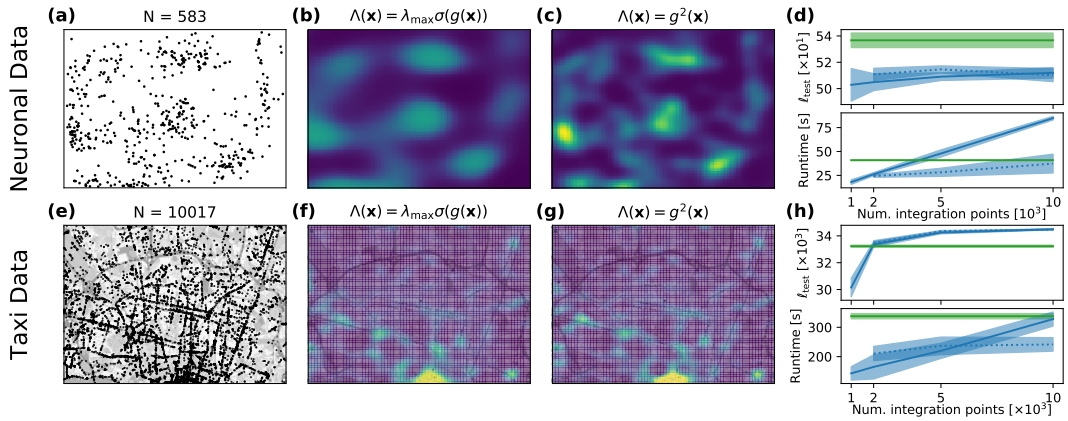



Figure 6: **Fits to real world datasets. (a)** Position of the mouse while the recorded neuron spiked. **(b)** Posterior mean obtained by the variational mean–field algorithm for the sigmoidal Gaussian Cox process. **(c)** Same as in (b) for the variational approximation of the squared Gaussian Cox process. **(d)** Log expected test–likelihood $\ell_{\text{test}}$ and runtime as function of number of integration points for both algorithms. The dotted line is obtained by first fitting the sigmoidal model with 1000 integration points and then with the number that is indicated on the x-axis. Shaded area is mean $\pm$ standard deviation obtained in 5 repeated fits. **(e)–(h)** Same as (a)–(d), but for a dataset, where the observations are positions of taxi pick–ups in the city of Porto.

descent methods. In fact, a comparison with a different variational algorithm for the same model - not based on augmentation, but on direct approximation of the posterior with a Gaussian - shows that the qualities of inference for both approaches are similar, while the mean field algorithm is at least one order of magnitude faster. We use the same variable augmentation method for computation of the MAP estimate for the (unaugmented) posterior by a fast EM algorithm. This is finally applied to the calculation of Laplace's approximation. Both methods yield an explicit result for the approximate GP posterior. Since the corresponding effective likelihood contains a continuum of the GP latent variables, the exact computations of means and marginal variances would require the inversion of a linear operator instead of a simpler matrix inverse. While for specific priors, this problem could be solved by PDE or ODE methods, we resort to a well known sparse GP approach with inducing points in this paper. We can apply this to arbitrary kernels but need to solve spatial integrals over the domain. These can be (at least for moderate dimensionality) well approximated by simple Monte Carlo integration. Advantage of this approach is, that one is not limited to rectangular domains. The only requirement is that the volume $|\mathcal{X}|$ is known. An alternative Poisson model for which similar spatial integrals can be performed analytically (Lloyd et al., 2015) within the sparse GP approximation (limited to squared exponential kernels and rectangular domains) is based on a quadratic link function (Lloyd et al., 2015; Flaxman et al., 2017; John and Hensman, 2018). We compare our variational algorithm with the variational algorithm of Lloyd et al. (2015) on different datasets and observe that both algorithms act on the same order of magnitude in terms of runtime (with slight advantages for our variational mean field algorithm). As expected, we show that whether one or the other model is better in predictive power is highly data dependent.

As an alternative to the Monte Carlo integration in our approach we could avoid the infinite dimensionality of the latent GP from the beginning by working with a binning scheme for the Poisson observations as in Hensman et al. (2015). It would be straightforward to adopt our augmentation method to this case. The resulting Poisson likelihoods would then be augmented by pairs of Poisson and Pólya–Gamma variables (see Donner and Opper (2017)) for each bin. This approach could be favourable when the number of observed data points becomes very large, because the discretisation method does not scale with the number data points but with the resolution of discretisation. However, we do expect, that any approach based on either spatial discretisation or on the sparse, inducing point method would become problematic for large or high dimensional domains $\mathcal{X}$. Alternative methods based on spectral representations of kernels (Knollmüller et al., 2017; John and Hensman, 2018) are promising for tackling those problems.

It will be interesting to apply the variable augmentation method to other Bayesian models with the sigmoid link function. For example, the inherent boundedness of the resulting intensity can be crucial for point processes such as the nonlinear *Hawkes process* (Hawkes, 1971) which is widely used for modelling stock market data (Embrechts et al., 2011) or seismic activity (Ogata, 1998). For other point process models the sigmoid function appears naturally. We mention the kinetic Ising model, a Markov jump process (Donner and Opper, 2017) which was originally introduced to model the dynamics of classical spin systems in physics. More recently it was used to model the joint activity of neurons (Dunn et al., 2015). Finally, a Gaussian process density model introduced by (Murray et al., 2009) can be treated by the augmentations developed in this work (Donner and Opper, 2018).

## Acknowledgments

CD was supported by the Deutsche Forschungsgemeinschaft (GRK1589/2) and partially funded by Deutsche Forschungsgemeinschaft (DFG) through grant CRC 1294 "Data Assimilation", Project (A06) "Approximative Bayesian inference and model selection for stochastic differential equations (SDEs)".

## Appendix A. Poisson processes

In this paragraph we briefly summarise those properties of a Poisson process, which are relevant for this work. For a thorough and more complete description we recommend the concise book by Kingman (1993), particularly chapter 3 and 5.

We consider a general space $\mathcal{Z}$ and a countable subset $\Pi_{\mathcal{Z}} = \{\boldsymbol{z}; \boldsymbol{z} \in \mathcal{Z}\}$.

**Definition of a Poisson process** A random countable subset $\Pi_{\mathcal{Z}} \subset \mathcal{Z}$ is a Poisson process on $\mathcal{Z}$, if

i) for any sequence of disjoint subsets $\{\mathcal{Z}_k \subset \mathcal{Z}\}_{k=1}^{K}$ the cardinality of the union $N(\mathcal{Z}_k) \doteq |\{\Pi_{\mathcal{Z}} \cap \mathcal{Z}_k\}|$ is independent of $N(\mathcal{Z}_l)$ for all $l \neq k$.

ii) $N(\mathcal{Z}_k)$ is Poisson distributed with mean $\int_{\mathcal{Z}_k} \Lambda(\boldsymbol{z})d\boldsymbol{z}$, and mean measure $\Lambda(\boldsymbol{z}) : \mathcal{X} \to \mathbb{R}^+$.

If the mean measure is constant ($\Lambda(\boldsymbol{z}) = \Lambda$) the Poisson process is *homogeneous*, and *inhomogeneous* otherwise.

**Campbell's Theorem** Let $\Pi_{\mathcal{Z}}$ be a Poisson process on $\mathcal{Z}$ with mean measure $\Lambda(\boldsymbol{z})$. Furthermore, we define a function $h(\boldsymbol{z}) : \mathcal{Z} \to \mathbb{R}$ and the sum

$$H(\Pi_{\mathcal{Z}}) = \sum_{\boldsymbol{z} \in \Pi_{\mathcal{Z}}} h(\boldsymbol{z}).$$

If $\Lambda(\boldsymbol{z}) < \infty$ for $\boldsymbol{z} \in \mathcal{Z}$, then

$$\mathbb{E}_{P_\Lambda}\left[e^{\xi H(\Pi_{\mathcal{Z}})}\right] = \exp\left\{\int_{\mathcal{Z}} \left(e^{\xi h(\boldsymbol{z})} - 1\right) \Lambda(\boldsymbol{z})d\boldsymbol{z}\right\}, \tag{36}$$

for any $\xi \in \mathbb{C}$, such that the integral converges. $P_\Lambda$ is the probability measure of a Poisson process with intensity $\Lambda(\boldsymbol{z})$. Mean and variance are obtained as

$$\mathbb{E}_{P_\Lambda}\left[H(\Pi_{\mathcal{Z}})\right] = \int_{\mathcal{Z}} h(\boldsymbol{z})\Lambda(\boldsymbol{z})d\boldsymbol{z},$$
$$\mathrm{Var}_{P_\Lambda}\left[H(\Pi_{\mathcal{Z}})\right] = \int_{\mathcal{Z}} [h(\boldsymbol{z})]^2\Lambda(\boldsymbol{z})d\boldsymbol{z}.$$

Note, that Equation (36) defines the *characteristic functional* of a Poisson process.

**Marked Poisson process** Let $\Pi_{\mathcal{Z}} = \{\boldsymbol{z}_n\}_{n=1}^{N}$ a Poisson process on $\mathcal{Z}$ with intensity $\Lambda(\boldsymbol{z})$. Then $\Pi_{\hat{\mathcal{Z}}} = \{(\boldsymbol{z}_n, \boldsymbol{m}_n)\}_{n=1}^{N}$ is again a Poisson process on the product space $\hat{\mathcal{Z}} = \mathcal{Z} \times \mathcal{M}$, if $\boldsymbol{m}_n \sim p(\boldsymbol{m}_n|\boldsymbol{z}_n)$ is drawn independently at each $\boldsymbol{z}_n$. The $\boldsymbol{m}_n \in \mathcal{M}$ are the so–called 'marks', and the resulting Process is a *marked Poisson process* with intensity

$$\Lambda(\boldsymbol{z}, \boldsymbol{m}) = \Lambda(\boldsymbol{z})p(\boldsymbol{m}|\boldsymbol{z}).$$

It is straightforward to extend Campbell's theorem and to show that the characteristic functional of such a process is

$$\mathbb{E}_{P_\Lambda}\left[e^{\xi H(\Pi_{\hat{\mathcal{Z}}})}\right] = \exp\left\{\int_{\hat{\mathcal{Z}}} \left(e^{\xi h(\boldsymbol{z},\boldsymbol{m})} - 1\right) \Lambda(\boldsymbol{z}, \boldsymbol{m})\, d\boldsymbol{m}d\boldsymbol{z}\right\}, \tag{37}$$

with $h(\boldsymbol{z}, \boldsymbol{m}) : \hat{\mathcal{Z}} \to \mathbb{R}$ and $H(\Pi_{\hat{\mathcal{Z}}}) = \sum_{(\boldsymbol{z},\boldsymbol{m}) \in \Pi_{\hat{\mathcal{Z}}}} h(\boldsymbol{z}, \boldsymbol{m})$.

## Appendix B. The Pólya-Gamma density

The Pólya-Gamma density (Polson et al., 2013) has the useful property, that it allows to represent the inverse hyperbolic cosine by an infinite Gaussian mixture as

$$\cosh^{-b}(c/2) = \int_0^\infty \exp\left(-\frac{c^2}{2}\omega\right) p_{\mathrm{PG}}(\omega|b,0)d\omega,$$

with parameter $b > 0$. Furthermore, one can define a *tilted Pólya-Gamma density* as

$$p_{\mathrm{PG}}(\omega|b,c) = \frac{\exp\left(-\frac{c^2}{2}\omega\right)}{\cosh^{-b}(c/2)} p_{\mathrm{PG}}(\omega|b,0).$$

From those two equations the moment generating function can be obtained from the basic definition, being

$$\int_0^\infty e^{\xi\omega} p_{\mathrm{PG}}(\omega|b,c)d\omega = \frac{\cosh^b(c/2)}{\cosh^b\left(\sqrt{\frac{c^2/2-\xi}{2}}\right)},$$

and differentiating with respect to $\xi$ at $\xi = 0$ yields the first moment

$$\mathbb{E}_{p_{\mathrm{PG}}}[\omega] = \frac{b}{2c}\tanh(c/2).$$

## Appendix C. Variational inference for stochastic processes

**Densities for random processes** A stochastic process $X$ with probability measure $P(X)$ often has no density with respect to Lebesgue measure, since $X$ can be an infinite dimensional object such as a function for the case of a Gaussian process. However, one can define densities with respect to another (reference) measure $R(X)$ written as

$$\boldsymbol{p}(X) = \frac{dP}{dR}(X), \tag{38}$$

if $R(X)$ is absolutely continuous with respect to $P(X)$ (if $R(X) = 0$ then $P(X) = 0$). Using such a density, expectations are

$$\mathbb{E}_P[f(X)] = \int f(X)dP(X) = \int f(x)\boldsymbol{p}(x)dR(X) = \mathbb{E}_R[f(x)\boldsymbol{p}(x)].$$

The density in Equation (38) is known as the *Radon–Nikodým derivative* of $R$ with respect to $P$ (Konstantopoulos et al., 2011).

**Poisson process density** As specific example consider the prior density of the Poisson process in Equation (9), which is defined with respect to a reference measure

$$\boldsymbol{p}_\Lambda(\Pi_\mathcal{Z}) = \frac{dP_\Lambda}{dP_{\Lambda_0}}(\Pi_\mathcal{Z}) = \exp\left(-\int_\mathcal{Z}(\Lambda(\boldsymbol{z}) - \Lambda_0(\boldsymbol{z}))d\boldsymbol{z}\right) \prod_{\boldsymbol{z}_n \in \Pi_\mathcal{Z}} \frac{\Lambda(\boldsymbol{z}_n)}{\Lambda_0(\boldsymbol{z}_n)},$$

where $P_{\Lambda_0}$ is the probability measure with intensity $\Lambda_0$ and the expectation is defined as

$$\mathbb{E}_{P_\Lambda}\left[\sum_{\boldsymbol{z}_n \in \Pi_\mathcal{Z}} u(\boldsymbol{z}_n)\right] = \mathbb{E}_{P_{\Lambda_0}}\left[\boldsymbol{p}_\Lambda(\Pi_\mathcal{Z}) \sum_{\boldsymbol{z}_n \in \Pi_\mathcal{Z}} u(\boldsymbol{z}_n)\right]. \tag{39}$$

Calculating the expectation of $e^{\xi H(\Pi_\mathcal{Z})}$ with Equation (39) we identify the characteristic function of a Poisson process (see Equation (37)) with intensity $\Lambda(\boldsymbol{z})$.

**Kullback-Leibler divergence** Using these densities we can express the Kullback-Leibler divergence between two probability measures.

The KL–divergence between $\boldsymbol{q}(X)$ and $\boldsymbol{p}(X)$ is defined as

$$\mathrm{D}_{\mathrm{KL}}(Q\|P) = \mathbb{E}_Q\left[\log \frac{dQ}{dP}(X)\right] = \int \log \frac{\boldsymbol{q}(X)}{\boldsymbol{p}(X)} dQ(X),$$

where

$$\boldsymbol{q}(X) = \frac{dQ}{dR}(X),$$

and where $R(X)$ also is absolutely continuous to $Q(X)$. The KL–divergence does not depend on the reference measure $R(X)$.

## Appendix D. The posterior point process is a marked Poisson process

Here we prove that the optimal variational posterior point process in Equation (18) again is a Poisson process using Campbell's theorem. As posterior process in Equation (18) one gets

$$\boldsymbol{q}(\Pi_\mathcal{Z}) = \frac{dQ}{dP_\lambda}(\Pi_\mathcal{Z}) = \frac{\prod_{\boldsymbol{z}_m \in \Pi_\mathcal{Z}} e^{f(\boldsymbol{z}_m)}}{\mathbb{E}_{P_\lambda}\left[\prod_{\boldsymbol{z}_m \in \Pi_\mathcal{Z}} e^{f(\boldsymbol{z}_m)}\right]} = \frac{\prod_{\boldsymbol{z}_m \in \Pi_\mathcal{Z}} e^{f(\boldsymbol{z}_m)}}{\exp\left(\int_\mathcal{Z} (e^{f(\boldsymbol{z})} - 1)\lambda(\boldsymbol{z}) d\boldsymbol{z}\right)},$$

where $\Pi_\mathcal{Z}$ is some random set of points on space $\mathcal{Z}$ and $P_\lambda$ is a random Poisson measure with intensity $\lambda(\boldsymbol{z})$. To proof, that the resulting point process density $\boldsymbol{q}(\Pi_\mathcal{Z})$ is again a Poisson process we calculate the characteristic functional for some arbitrary function $h : \mathcal{Z} \to \mathbb{R}$

$$\begin{aligned}
\mathbb{E}_Q\left[\prod_{\boldsymbol{z}_m \in \Pi_\mathcal{Z}} e^{h(\boldsymbol{z}_m)}\right] &= \frac{\mathbb{E}_{P_\lambda}\left[\prod_{\boldsymbol{z}_m \in \Pi_\mathcal{Z}} e^{h(\boldsymbol{z}_m) + f(\boldsymbol{z}_m)}\right]}{\exp\left(\int_\mathcal{Z} (e^{f(\boldsymbol{z})} - 1)\lambda(\boldsymbol{z}) d\boldsymbol{z}\right)} \\
&= \frac{\exp\left(\int_\mathcal{Z} (e^{h(\boldsymbol{z}) + f(\boldsymbol{z})} - 1)\lambda(\boldsymbol{z}) d\boldsymbol{z}\right)}{\exp\left(\int_\mathcal{Z} (e^{f(\boldsymbol{z})} - 1)\lambda(\boldsymbol{z}) d\boldsymbol{z}\right)} \\
&= \exp\left(\int_\mathcal{Z} (e^{h(\boldsymbol{z})} - 1) e^{f(\boldsymbol{z})} \lambda(\boldsymbol{z}) d\boldsymbol{z}\right) \\
&= \exp\left(\int_\mathcal{Z} (e^{h(\boldsymbol{z})} - 1) \Lambda_Q(\boldsymbol{z}) d\boldsymbol{z}\right).
\end{aligned}$$

We identify the last row as the generating functional of a Poisson process (37) with $\xi = 1$. The intensity of the process is $\Lambda_Q(\boldsymbol{z}) = e^{f(\boldsymbol{z})}\lambda(\boldsymbol{z})$. With the fact that a Poisson process is uniquely characterised by its generating function (Kingman, 1993, chap. 3), the proof is complete.

## Appendix E. Sparse Gaussian process approximation

To solve the inference problem for the function $g$, we define a sparse GP, using the same prior $P$, but by an effective likelihood which depends on a finite set of function values $\boldsymbol{g}_s = (g_1, \ldots, g_L)^\top$ only. Hence, we get

$$\frac{dQ_2^s}{dP}(g) = \boldsymbol{q}_2^s(\boldsymbol{g}_s) \tag{40}$$

and the sparse posterior measure is

$$dQ_2^s(g) = \boldsymbol{q}_2^s(\boldsymbol{g}_s)dP(g) = dP(g|\boldsymbol{g}_s) \times \boldsymbol{q}_2^s(\boldsymbol{g}_s)dP(\boldsymbol{g}_s),$$

where the last equality holds true, since Equation (40) only depends on $\boldsymbol{g}_s$. The KL–divergence between the full posterior density

$$\boldsymbol{q}_2(g) = \frac{dQ_2}{dP}(g) = \frac{e^{U(g)}}{\mathbb{E}_P\left[e^{U(g)}\right]}$$

and the sparse one $\boldsymbol{q}_2^s(\boldsymbol{g}_s)$ is given by

$$\begin{aligned}\mathrm{D}_{\mathrm{KL}}(Q_2^s \| Q_2) &= \mathbb{E}_{Q_2^s}\left[\log \frac{\boldsymbol{q}_2^s(\boldsymbol{g}_s)}{\boldsymbol{q}_2(g)}\right] = \mathbb{E}_{P(\boldsymbol{g}_s)}\left[\boldsymbol{q}_2^s(\boldsymbol{g}_s)\mathbb{E}_{P(g|\boldsymbol{g}_s)}\left[\log \frac{\boldsymbol{q}_2^s(\boldsymbol{g}_s)}{e^{U(g)}}\right]\right] + \text{const.} \\ &= \mathbb{E}_{P(\boldsymbol{g}_s)}\left[\boldsymbol{q}_2^s(\boldsymbol{g}_s)\log \frac{\boldsymbol{q}_2^s(\boldsymbol{g}_s)}{e^{\mathbb{E}_{P(g|\boldsymbol{g}_s)}[U(g)]}}\right] + \text{const.}\end{aligned}$$

From this we derive directly the posterior density for the sparse GP

$$\boldsymbol{q}_2^s(g) \propto e^{U^s(\boldsymbol{g}_s)},$$

with the sparse log–likelihood

$$U^s(\boldsymbol{g}_s) = \mathbb{E}_{P(g|\boldsymbol{g}_s)}\left[U(g)\right] = \int U(g)dP(g|\boldsymbol{g}_s).$$

## Appendix F. Lower bound & hyperparameter optimization

The lower bound in Equation (12) is given by

$$\begin{aligned}
\mathcal{L}(\boldsymbol{q}) =& \mathbb{E}_Q\left[\log \frac{L(\mathcal{D}, \boldsymbol{\omega}_N, \Pi_{\hat{\mathcal{X}}}|g,\lambda)}{\boldsymbol{q}_1(\boldsymbol{\omega}_N)\boldsymbol{q}_1(\Pi_{\hat{\mathcal{X}}})\boldsymbol{q}_2^s(g)\boldsymbol{q}_2(\lambda)}\right] \\
=& \int_{\hat{\mathcal{X}}} \left(\mathbb{E}_Q\left[f(\omega, -g(\boldsymbol{x}))\right] - \mathbb{E}_Q\left[\log \Lambda_1\right] + \mathbb{E}_Q\left[\log \lambda\right] + 1\right) \Lambda_1(\boldsymbol{x},\omega) d\boldsymbol{x} d\omega \\
& - \int_{\hat{\mathcal{X}}} \Lambda_1(\boldsymbol{x},\omega) d\boldsymbol{x} d\omega \\
& + \sum_{n=1}^{N} \left( \mathbb{E}_Q\left[f(\omega_n, g_n)\right] + \mathbb{E}_Q\left[\log \lambda\right] - \cosh\left(\frac{c_1^{(n)}}{2}\right) + \frac{\left(c_1^{(n)}\right)^2}{2} \mathbb{E}_Q\left[\omega_n\right] \right) \\
& - \frac{1}{2} trace(K_s^{-1}(\Sigma_2^s + \boldsymbol{\mu}_2^s(\boldsymbol{\mu}_2^s)^\top)) - \frac{1}{2}\log\det(2\pi K_s) + \frac{1}{2}\log\det(2\pi e \Sigma_2^s) \\
& + \alpha_0 \log \beta_0 - \log(\Gamma(\alpha_0)) + (\alpha_0 - 1)\mathbb{E}_Q\left[\log\lambda\right] - \beta_0 \mathbb{E}_Q\left[\lambda\right] \\
& + \alpha_2 - \log\beta_2 + \log\Gamma(\alpha_2) + (1-\alpha_2)\psi(\alpha_2).
\end{aligned}$$

To optimise the covariance kernel parameters $\boldsymbol{\theta}$ we differentiate the lower bound with respect to these parameters and perform then gradient ascent. The gradient for one specific parameter $\theta$ is given by

$$\begin{aligned}
\frac{\partial \mathcal{L}(\boldsymbol{q})}{\partial \theta} =& \int_{\hat{\mathcal{X}}} \frac{\partial \mathbb{E}_Q\left[f(\omega, -g(\boldsymbol{x}))\right]}{\partial \theta} \Lambda_1(\boldsymbol{x},\omega) d\boldsymbol{x} d\omega + \sum_{n=1}^{N} \frac{\partial \mathbb{E}_Q\left[f(\omega_n, g(\boldsymbol{x}_n))\right]}{\partial \theta} \\
& - \frac{1}{2}\frac{trace(K_s^{-1}(\Sigma_2^s + \boldsymbol{\mu}_2^s(\boldsymbol{\mu}_2^s)^\top))}{\partial\theta} - \frac{1}{2}\frac{\partial \log\det(2\pi K_s)}{\partial\theta} \\
=& \int_{\hat{\mathcal{X}}} \frac{\partial \mathbb{E}_Q\left[f(\omega, -g(\boldsymbol{x}))\right]}{\partial \theta} \Lambda_1(\boldsymbol{x},\omega) d\boldsymbol{x} d\omega + \sum_{n=1}^{N} \frac{\partial \mathbb{E}_Q\left[f(\omega_n, g(\boldsymbol{x}_n))\right]}{\partial \theta} \\
& + \frac{1}{2} trace\left(K_s^{-1}\frac{\partial K_s}{\partial\theta} K_s^{-1}(\Sigma_2^s + \boldsymbol{\mu}_2^s(\boldsymbol{\mu}_2^s)^\top)\right) \\
& - \frac{1}{2} trace\left(K_s^{-1}\frac{\partial K_s}{\partial\theta}\right).
\end{aligned}$$

The derivatives of function $\mathbb{E}_Q\left[f(\omega, g(\boldsymbol{x}))\right]$ are

$$\frac{\partial \mathbb{E}_Q\left[f(\omega, g(\boldsymbol{x}))\right]}{\partial\theta} = \frac{1}{2}\left(\frac{\partial \mathbb{E}_Q\left[g(\boldsymbol{x})\right]}{\partial\theta} - \frac{\partial \mathbb{E}_Q\left[g(\boldsymbol{x})^2\right]}{\partial\theta}\mathbb{E}_Q\left[\omega\right]\right),$$

with

$$\begin{aligned}
\frac{\partial \mathbb{E}_Q\left[g(\boldsymbol{x})\right]}{\partial\theta} =& \frac{\partial \boldsymbol{\kappa}(\boldsymbol{x})}{\partial\theta}\boldsymbol{\mu}_2^s, \\
\frac{\partial \mathbb{E}_Q\left[g(\boldsymbol{x})^2\right]}{\partial\theta} =& \frac{\partial \tilde{k}(\boldsymbol{x},\boldsymbol{x})}{\partial\theta} + \frac{\partial \boldsymbol{\kappa}(\boldsymbol{x})}{\partial\theta}^\top \left(\Sigma_2^s + \boldsymbol{\mu}_2^s(\boldsymbol{\mu}_2^s)^\top\right)\boldsymbol{\kappa}(\boldsymbol{x}) + \boldsymbol{\kappa}(\boldsymbol{x})^\top\left(\Sigma_2^s + \boldsymbol{\mu}_2^s(\boldsymbol{\mu}_2^s)^\top\right)\frac{\partial \boldsymbol{\kappa}(\boldsymbol{x})}{\partial\theta},
\end{aligned}$$

where $\boldsymbol{\kappa}(\boldsymbol{x}) = \boldsymbol{k}_s(\boldsymbol{x})^\top K_s^{-1}$ and $\tilde{k}(\boldsymbol{x}, \boldsymbol{x}) = k(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}_s(\boldsymbol{x}) K_s^{-1} \boldsymbol{k}_s(\boldsymbol{x})^\top$. The remaining two terms are:

$$\begin{aligned}\frac{\partial \tilde{k}(\boldsymbol{x}, \boldsymbol{x})}{\partial \theta} =& \frac{\partial k(\boldsymbol{x}, \boldsymbol{x})}{\partial \theta} - \frac{\partial \boldsymbol{\kappa}(\boldsymbol{x})}{\partial \theta} \boldsymbol{k}_s(\boldsymbol{x}) - \boldsymbol{\kappa}(\boldsymbol{x}) \frac{\partial \boldsymbol{k}_s(\boldsymbol{x})}{\partial \theta}, \\ \frac{\partial \boldsymbol{\kappa}(\boldsymbol{x})}{\partial \theta} =& \frac{\partial \boldsymbol{k}_s(\boldsymbol{x})^\top}{\partial \theta} K_s^{-1} - \boldsymbol{k}_s(\boldsymbol{x}) K_s^{-1} \frac{\partial K_s}{\partial \theta} K_s^{-1}.\end{aligned}$$

After each variational step the hyperparameters are updated by

$$\boldsymbol{\theta}_{\text{new}} = \boldsymbol{\theta}_{\text{old}} + \varepsilon \frac{\partial \mathcal{L}(q)}{\partial \boldsymbol{\theta}},$$

where $\varepsilon$ is the step size.

## References

Ryan P. Adams, Iain Murray, and David J. C. MacKay. Tractable nonparametric bayesian inference in poisson processes with gaussian process intensities. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 9–16, 2009. doi: 10.1145/1553374.1553376.

Philipp Batz, Andreas Ruttor, and Manfred Opper. Approximate Bayes learning of stochastic differential equations. *Phys. Rev.*, E98(2):022109, 2018. doi: 10.1103/PhysRevE.98.022109.

Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.

David R. Brillinger. Maximum likelihood analysis of spike trains of interacting nerve cells. *Biological Cybernetics*, 59(3):189–200, 1988. doi: 10.1007/BF00318010.

Anders Brix and Peter J. Diggle. Spatiotemporal prediction for log-gaussian cox processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(4):823–841, 2001. doi: 10.1111/1467-9868.00315.

D. R. Cox. Some statistical methods connected with series of events. *Journal of the Royal Statistical Society. Series B (Methodological)*, 17(2):129–164, 1955. ISSN 00359246.

Lehel Csató. Gaussian processes-iterative sparse approximations. 2002. URL `http://publications.aston.ac.uk/1327/`.

Lehel Csató and Manfred Opper. Sparse on-line gaussian processes. *Neural Computation*, 14(3):641–668, 2002. doi: 10.1162/089976602317250933.

John P Cunningham, Byron M Yu, Krishna V Shenoy, and Maneesh Sahani. Inferring neural firing rates from spike trains using gaussian processes. In *Advances in Neural Information Processing Systems 20*, pages 329–336. 2008. URL `http://papers.nips.cc/paper/3229-inferring-neural-firing-rates-from-spike-trains-using-gaussian-processes.pdf`.

Alexander G. de G. Matthews, James Hensman, Richard Turner, and Zoubin Ghahramani. On sparse variational methods and the kullback-leibler divergence between stochastic processes. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51, pages 231–239, 2016. URL http://proceedings.mlr.press/v51/matthews16.html.

Christian Donner and Manfred Opper. Inverse ising problem in continuous time: A latent variable approach. *Phys. Rev. E*, 96:062104, 2017. doi: 10.1103/PhysRevE.96.062104.

Christian Donner and Manfred Opper. Efficient bayesian inference for a gaussian process density model. In *Proceedings of the 34th Conference on Uncertainty in Artificial Intelligence*, 2018. URL http://auai.org/uai2018/proceedings/papers/34.pdf.

Benjamin Dunn, Maria Mrreaunet, and Yasser Roudi. Correlations and functional connections in a population of grid cells. *PLOS Computational Biology*, 11(2):1–21, 2015. doi: 10.1371/journal.pcbi.1004052.

Paul Embrechts, Thomas Liniger, and Lu Lin. Multivariate hawkes processes: an application to financial data. *Journal of Applied Probability*, 48(A):367378, 2011. doi: 10.1239/jap/1318940477.

Seth Flaxman, Yee Whye Teh, and Dino Sejdinovic. Poisson intensity estimation with reproducing kernels. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54, pages 270–279. PMLR, 2017. URL http://proceedings.mlr.press/v54/flaxman17a.html.

Centre For The Biology Of Memory and Fransesca Sargolini. Grid cell data of sargolini et al 2006. 2014. doi: 10.11582/2014.00003.

Tom Gunter, Chris Lloyd, Michael A. Osborne, and Stephen J. Roberts. Efficient bayesian nonparametric modelling of structured point processes. In *Uncertainty in Artificial Intelligence (UAI)*, 2014. URL https://arxiv.org/abs/1407.6949.

Alan G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90, 1971. ISSN 00063444.

James Hensman, Alexander G Matthews, Maurizio Filippone, and Zoubin Ghahramani. Mcmc for variationally sparse gaussian processes. In *Advances in Neural Information Processing Systems 28*, pages 1648–1656. 2015. URL http://papers.nips.cc/paper/5875-mcmc-for-variationally-sparse-gaussian-processes.pdf.

ST John and James Hensman. Large-scale Cox process inference using variational Fourier features. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 2362–2370, 2018. URL http://proceedings.mlr.press/v80/john18a.html.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *preprint arXiv*, abs/1412.6980, 2014. URL http://arxiv.org/abs/1412.6980.

John Frank Charles Kingman. *Poisson processes*. Oxford University Press, 1993. ISBN 9780198536932.

Alisa Kirichenko and Harry van Zanten. Optimality of poisson processes intensity learning with gaussian processes. *Journal of Machine Learning Research*, 16:2909–2919, 2015. URL http://jmlr.org/papers/v16/kirichenko15a.html.

J. Knollmüller, T. Steininger, and T. A. Enßlin. Inference of signals with unknown correlation structure from nonlinear measurements. *ArXiv e-prints*, 2017. URL https://arxiv.org/abs/1711.02955.

Takis Konstantopoulos, Zurab Zerakidze, and Grigol Sokhadze. *Radon–Nikodým Theorem*, pages 1161–1164. 2011. ISBN 978-3-642-04898-2.

P. A. W Lewis and G. S. Shedler. Simulation of nonhomogeneous poisson processes by thinning. *Naval Research Logistics Quarterly*, 26(3):403–413, 1979. doi: 10.1002/nav.3800260304.

Scott Linderman, Matthew Johnson, and Ryan P Adams. Dependent multinomial models made easy: Stick-breaking with the polya-gamma augmentation. In *Advances in Neural Information Processing Systems 28*, pages 3456–3464. 2015. URL http://papers.nips.cc/paper/5660-dependent-multinomial-models-made-easy-stick-breaking-with-the-polya-gamma-augmentation.

Scott Linderman, Matthew Johnson, Andrew Miller, Ryan Adams, David Blei, and Liam Paninski. Bayesian Learning and Inference in Recurrent Switching Linear Dynamical Systems. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54, pages 914–922, 2017. URL http://proceedings.mlr.press/v54/linderman17a.html.

Chris Lloyd, Tom Gunter, Michael Osborne, and Stephen Roberts. Variational inference for gaussian process modulated poisson processes. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pages 1814–1822, 2015. URL http://proceedings.mlr.press/v37/lloyd15.html.

Chris Lloyd, Tom Gunter, Michael Osborne, Stephen Roberts, and Tom Nickson. Latent point process allocation. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51, pages 389–397, 2016. URL http://proceedings.mlr.press/v51/lloyd16.html.

Alexander G. de G. Matthews, Mark van der Wilk, Tom Nickson, Keisuke. Fujii, Alexis Boukouvalas, Pablo Leon-Villagra, Zoubin Ghahramani, and James Hensman. Gpflow: A gaussian process library using tensorflow. *Journal of Machine Learning Research*, 18 (40):1–6, 2017. URL http://jmlr.org/papers/v18/16-537.html.

Jesper Møller, Anne Randi Syversveen, and Rasmus Plenge Waagepetersen. Log gaussian cox processes. *Scandinavian Journal of Statistics*, 25(3):451–482, 1998. doi: 10.1111/1467-9469.00115.

Luis Moreira-Matias, Joao Gama, Michel Ferreira, Joao Mendes-Moreira, and Luis Damas. Predicting taxi–passenger demand using streaming data. *IEEE Transactions on Intelligent Transportation Systems*, 14(3):1393–1402, 2013.

Iain Murray, Zoubin Ghahramani, and David J. C. MacKay. Mcmc for doubly-intractable distributions. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, pages 359–366, 2006. ISBN 0-9749039-2-2.

Iain Murray, David MacKay, and Ryan P Adams. The gaussian process density sampler. In *Advances in Neural Information Processing Systems 21*, pages 9–16. 2009. URL `http://papers.nips.cc/paper/3410-the-gaussian-process-density-sampler.pdf`.

Iain Murray, Ryan Adams, and David MacKay. Elliptical slice sampling. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9, pages 541–548, 2010. URL `http://proceedings.mlr.press/v9/murray10a.html`.

Yosihiko Ogata. Space-time point-process models for earthquake occurrences. *Annals of the Institute of Statistical Mathematics*, 50(2):379–402, 1998. doi: 10.1023/A:1003403601725.

Nicholas G. Polson, James G. Scott, and Jesse Windle. Bayesian inference for logistic models using plyagamma latent variables. *Journal of the American Statistical Association*, 108 (504):1339–1349, 2013. doi: 10.1080/01621459.2013.829001.

William H Press, Brian P Flannery, Saul A Teukolsky, William T Vetterling, et al. *Numerical recipes*, volume 3. Cambridge University Press, 2007. ISBN 978-0-521-88068-8.

Carl E. Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, USA, 2006. ISBN 0-262-18253-X.

Yves-Laurent Kom Samo and Stephen Roberts. Scalable nonparametric bayesian inference on point processes with gaussian processes. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pages 2227–2236, 2015. URL `http://proceedings.mlr.press/v37/samo15.html`.

Francesca Sargolini, Marianne Fyhn, Torkel Hafting, Bruce L. McNaughton, Menno P. Witter, May-Britt Moser, and Edvard I. Moser. Conjunctive representation of position, direction, and velocity in entorhinal cortex. *Science*, 312(5774):758–762, 2006. doi: 10.1126/science.1125572.

Arno Solin. *Stochastic Differential Equation Methods for Spatio-Temporal Gaussian Process Regression*. Aalto University, 2016. ISBN 978-952-60-6711-7.

Dietrich Stoyan and Antti Penttinen. Recent applications of point process methods in forestry statistics. *Statistical Science*, 15(1):61–78, 2000. ISSN 08834237.

Yee W. Teh and Vinayak Rao. Gaussian process modulated renewal processes. In *Advances in Neural Information Processing Systems 24*, pages 2474–2482, 2011. URL `http://papers.nips.cc/paper/4358-gaussian-process-modulated-renewal-processes.pdf`.

Michalis Titsias. Variational learning of inducing variables in sparse gaussian processes. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5, pages 567–574, 2009. URL http://proceedings.mlr.press/v5/titsias09a.html.

Christian J. Walder and Adrian N. Bishop. Fast Bayesian intensity estimation for the permanental process. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3579–3588, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL http://proceedings.mlr.press/v70/walder17a.html.

Florian Wenzel, Théo Galy-Fajou, Christian Donner, Marius Kloft, and Manfred Opper. Scalable logit gaussian process classification. In *Advances in Approximate Bayesian Inference, NIPS Workshop*, 2017. URL http://approximateinference.org/2017/accepted/WenzelEtAl2017.pdf.

# Chapter 4

# Conference article: *Efficient Bayesian Inference for a Gaussian Process Density Model*

Accepted for publication in the conference proceedings of 34$^{\text{th}}$ *Conference on Uncertainty in Artificial Intelligence (UAI)* (Monterey, United States).

Authors:
Christian Donner[1,2], Manfred Opper[1,2]
[1]Technische Universität Berlin. [2]Bernstein Center for Computational Neuroscience Berlin.

Details:
Submitted: March 2018
Accepted: May 2018
URL: http://auai.org/uai2018/proceedings/papers/34.pdf
arXiv URL: https://arxiv.org/abs/1805.11494


**Chapter 4** This chapter comprises the publication (Donner and Opper 2018a), which is authored by myself (CD), and Prof. Manfred Opper (MO).

Contributions:
CD and MO conceived and designed the work. CD derived the inference algorithms and developed the Python code. CD performed the numerical experiments. CD wrote the manuscript with substantial contribution of MO.

| Python code on GitHub: https://github.com/christiando/SGPD_Inference.git |
|---|

# Efficient Bayesian Inference for a Gaussian Process Density Model

**Christian Donner***
Artificial Intelligence Group
Technische Universität Berlin
christian.donner@bccn-berlin.de

**Manfred Opper**
Artificial Intelligence Group
Technische Universität Berlin
manfred.opper@tu-berlin.de

## Abstract

We reconsider a nonparametric density model based on Gaussian processes. By augmenting the model with latent Pólya–Gamma random variables and a latent marked Poisson process we obtain a new likelihood which is conjugate to the model's Gaussian process prior. The augmented posterior allows for efficient inference by Gibbs sampling and an approximate variational mean field approach. For the latter we utilise sparse GP approximations to tackle the infinite dimensionality of the problem. The performance of both algorithms and comparisons with other density estimators are demonstrated on artificial and real datasets with up to several thousand data points.

## 1 INTRODUCTION

Gaussian processes (GP) provide highly flexible nonparametric prior distributions over functions [1]. They have been successfully applied to various statistical problems such as e.g. regression [2], classification [3], point processes [4] or the modelling of dynamical systems [5, 6]. Hence, it would seem natural to apply Gaussian processes also to density estimation which is one of the most basic statistical problems. GP density estimation, however, is a nontrivial task: Typical realisations of a GP do not respect non–negativity and normalisation of a probability density. Hence, functions drawn from a GP prior have to be passed through a nonlinear squashing function and the results have to be normalised subsequently to model a density. These operations make the corresponding posterior distributions non–Gaussian. Moreover, likelihoods depend on all the infinitely many GP function values in the domain rather than on the finite number of function values at observed data points. Since analytical inference is impossible, [7] introduced an interesting Markov chain Monte–Carlo sampler which allows for (asymptotically) exact inference for a Gaussian process density model, where the GP is passed through a sigmoid link function.[1] The approach is able to deal with the infinite dimensionality of the model, because the sampling of the GP variables is reduced to a finite dimensional problem by a point process representation. However, since the likelihood of the GP variables is not conjugate to the prior, the method has to resort to a time–consuming Metropolis–Hastings approach. In this paper we will use recent results on representing the sigmoidal squashing function as an infinite mixture of Gaussians involving Pólya–Gamma random variables [9] to augment the model in such a way that the model becomes tractable by a simpler Gibbs sampler. The new model structure allows also for a much faster variational Bayesian approximation.

The paper is organised as follows: Sec. 2 introduces the GP density model, followed by an augmentation scheme that makes its likelihood conjugate to the GP prior. With this model representation we derive two efficient Bayesian inference algorithms in Sec. 3, namely an exact Gibbs sampler and an approximate, fast variational Bayes algorithm. The performance of both algorithms is demonstrated in Sec. 4 on artificial and real data. Finally, Sec. 5 discusses potential extensions of the model.

## 2 GAUSSIAN PROCESS DENSITY MODEL

The generative model proposed by [7] constructs densities over some $d$-dimensional data space $\mathcal{X}$ to be of the

*Also affiliated with Bernstein Center for Computational Neuroscience.

[1] See [8] for an alternative model allowing, however, only for approximate inference schemes.

form

$$\rho(\boldsymbol{x}|g) = \frac{\sigma(g(\boldsymbol{x}))\pi(\boldsymbol{x})}{\int_{\mathcal{X}} \sigma(g(\boldsymbol{x}))\pi(\boldsymbol{x})d\boldsymbol{x}}. \quad (1)$$

$\pi(\boldsymbol{x})$ defines a (bounded) base probability measure over $\mathcal{X}$, which is usually taken from a fixed parametric family. The denominator ensures normalisation $\int_{\mathcal{X}} \rho(\boldsymbol{x}|g)d\boldsymbol{x} = 1$. The choice of $\pi(\boldsymbol{x})$ is important as will be discussed Sec. 5. A prior distribution over densities is introduced by assuming a Gaussian process prior [1] over the function $g(\boldsymbol{x}) : \mathcal{X} \to \mathbb{R}$. The GP is defined by a mean function $\mu(\boldsymbol{x})$ (in this paper, we consider only constant mean functions $\mu(\boldsymbol{x}) = \mu_0$) and covariance kernel $k(\boldsymbol{x}, \boldsymbol{x}')$. Finally, $\sigma(z) = \frac{1}{1+e^{-z}}$ is the sigmoid function, which guarantees that the density is non–negative and bounded.

In Bayesian inference, the posterior distribution of $g$ given observed data $\mathcal{D} = \{\boldsymbol{x}_n\}_{n=1}^N$ with $\boldsymbol{x} \in \mathcal{X}$ is computed from the GP prior $p(g)$ and the likelihood as

$$p(g|\mathcal{D}) \propto p(\mathcal{D}|g)p(g).$$

The likelihood is given by

$$p(\mathcal{D}|g) = \frac{\prod_{n=1}^N \sigma(g(\boldsymbol{x}_n))\pi(\boldsymbol{x}_n)}{\left(\int_{\mathcal{X}} \sigma(g(\boldsymbol{x}))\pi(\boldsymbol{x})d\boldsymbol{x}\right)^N}. \quad (2)$$

Practical inference for this problem, however, is non-trivial, because (i) the posterior is non–Gaussian and (ii) the likelihood involves an integral of $g$ over the whole space. Thus, in contrast to simpler problems such as GP regression or classification, it is impossible to reduce inference to finite dimensional integrals. To circumvent the problem that the likelihood is not conjugate to the GP prior, [7] proposed a Metropolis-Hastings MCMC algorithm for this model. We will show in the next sections that one can augment the model with auxiliary latent random variables in such a way that the resulting likelihood is of a conjugate form allowing for a more efficient Gibbs sampler with explicit conditional probabilities.

### 2.1 LIKELIHOOD AUGMENTATION

To obtain a likelihood which is conjugate to the GP $p(g)$ we require that it assumes a Gaussian form in $g$.

**Representing the denominator** As a starting point, we follow [10] and use the representation

$$\frac{1}{z^N} = \frac{\int_0^\infty \lambda^{N-1} e^{-\lambda z} d\lambda}{\Gamma(N)},$$

where $\Gamma(\cdot)$ is the gamma function. Identifying $z = \int_{\mathcal{X}} \sigma(g(\boldsymbol{x}))\pi(\boldsymbol{x})d\boldsymbol{x}$ in Eq. (2) we can rewrite the likelihood as $p(\mathcal{D}|g) = \int_0^\infty p(\mathcal{D}, \lambda|g)d\lambda$ where

$$p(\mathcal{D}, \lambda|g) \propto \exp\left(-\int_{\mathcal{X}} \lambda\sigma(g(\boldsymbol{x}))\pi(\boldsymbol{x})d\boldsymbol{x}\right) \times p(\lambda) \prod_{n=1}^N \lambda\sigma(g(\boldsymbol{x}_n))\pi(\boldsymbol{x}_n), \quad (3)$$

with the improper prior $p(\lambda) = \lambda^{-1}$ over the auxiliary latent variable $\lambda$. To transform the likelihood further into a form which is Gaussian in $g$, we utilise a representation of the sigmoid function as a scale mixture of Gaussians.

**Pólya–Gamma representation of sigmoid function** As discovered by [9], the inverse hyperbolic cosine can be represented as an infinite mixture of scaled Gaussians

$$\cosh^{-b}(z/2) = \int_0^\infty e^{-\frac{z^2}{2}\omega} p_{\mathrm{PG}}(\omega|b, 0)d\omega,$$

where $p_{\mathrm{PG}}(\omega|b, 0)$ is the *Pólya–Gamma density* of random variable $\omega \in \mathbb{R}^+$. Moments of those densities can be easily computed [9]. Later, we will also use the *tilted* Pólya-Gamma densities defined as

$$p_{\mathrm{PG}}(\omega|b, c) \propto \exp\left(-\frac{c^2}{2}\omega\right) p_{\mathrm{PG}}(\omega|b, 0). \quad (4)$$

These definitions allows for a Gaussian representation of the sigmoid function as

$$\sigma(z) = \frac{e^{z/2}}{2\cosh(z/2)} = \int_0^\infty e^{f(\omega,z)} p_{\mathrm{PG}}(\omega|1, 0)d\omega \quad (5)$$

with $f(\omega, z) = \frac{z}{2} - \frac{z^2}{2}\omega - \ln 2$. This result will be used to transform the products over observations $\sigma(g(\boldsymbol{x}_n))$ in the likelihood (3) into a Gaussian form.

We will next deal with the first term in the likelihood (3) which contains the integral over $\boldsymbol{x}$. For this part of the model we will derive a point process representation which can be understood as a generalisation of the approach of [7].

**Marked–Poisson representation** Utilising the sigmoid property $\sigma(z) = 1 - \sigma(-z)$ and the Pólya-Gamma representation (5) the integral in the exponent of Eq. (3) can be written as a double integral

$$-\int_{\mathcal{X}} \lambda\sigma(g(\boldsymbol{x}))\pi(\boldsymbol{x})d\boldsymbol{x} = \int_{\mathcal{X}} (\sigma(-g(\boldsymbol{x})) - 1)\lambda\pi(\boldsymbol{x})d\boldsymbol{x} = \int_{\mathcal{X}} \int_{\mathbb{R}^+} \left(e^{f(\omega,-g(\boldsymbol{x}))} - 1\right) \lambda\pi(\boldsymbol{x}) p_{\mathrm{PG}}(\omega|1, 0)d\omega d\boldsymbol{x}$$

Next we will use a result for the characteristic function of a Poisson process. Following [11, chap. 3] one has

$$\mathbb{E}_\phi\left[\prod_{\boldsymbol{z}\in\Pi} h(\boldsymbol{z})\right] = \exp\left(\int_{\mathcal{Z}} (h(\boldsymbol{z})-1)\phi(\boldsymbol{z})d\boldsymbol{z}\right). \quad (6)$$

$h(\cdot)$ is a function on a space $\mathcal{Z}$ and the expectation is over a Poisson process $\Pi$ with rate function $\phi(\boldsymbol{z})$. $\Pi = \{\boldsymbol{z}_m\}_{m=1}^M$ denotes a random set of points on the space $\mathcal{Z}$. To apply this result to our problem, we identify $\mathcal{Z} = \mathcal{X}\times R^+$, $\boldsymbol{z} = (\boldsymbol{x},\omega)$ and $\phi_\lambda(\boldsymbol{x},\omega) = \lambda\pi(\boldsymbol{x})p_{\mathrm{PG}}(\omega|1,0)$ and finally $h(\boldsymbol{z}) = e^{f(\omega,-g(\boldsymbol{x}))}$ to rewrite the exponential in Eq. (3) as

$$e^{-\int_{\mathcal{X}}\lambda\sigma(g(\boldsymbol{x}))\pi(\boldsymbol{x})d\boldsymbol{x}} = \mathbb{E}_{\phi_\lambda}\left[\prod_{(\omega,\boldsymbol{x})\in\Pi} e^{f(\omega,-g(\boldsymbol{x}))}\right]. \quad (7)$$

By substituting Eq. (5) and (7) into Eq.(3) we obtain the final augmented form of the likelihood of Eq. (2) which is one of the main results of our paper.

$$\begin{aligned} p(\mathcal{D},\lambda,\Pi,\boldsymbol{\omega}_N|g) \propto &\prod_{n=1}^N \phi_\lambda(\boldsymbol{x}_n,\omega_n)e^{f(\omega_n,g(\boldsymbol{x}_n))} \\ &\times p_{\phi_\lambda}(\Pi|\lambda)p(\lambda)\prod_{(\omega,\boldsymbol{x})\in\Pi} e^{f(\omega,-g(\boldsymbol{x}))}, \end{aligned} \quad (8)$$

with $p_\phi(\Pi|\lambda)$ being the density over a Poisson process $\Pi = \{(\boldsymbol{x}_m,\omega_m)\}_{m=1}^M$ in the augmented space $\mathcal{X}\times\mathbb{R}^+$ with intensity $\phi_\lambda(\boldsymbol{x},\omega)$. [2] This new process can be identified as a *marked Poisson process* [11, chap. 5], where the events $\{\boldsymbol{x}_m\}_{m=1}^M$ in the original data space $\mathcal{X}$ follow a Poisson process with rate $\lambda\pi(\boldsymbol{x})$. Then, on each event $\boldsymbol{x}_m$ an independent *mark* $\omega_m \sim p_{\mathrm{PG}}(\omega_m|b,0)$ is drawn at random from the Pólya–Gamma density. Finally, $\boldsymbol{\omega}_N = \{\omega_n\}_{n=1}^N$ is the set of latent Pólya–Gamma variables which result from the sigmoid augmentation at the observations $\boldsymbol{x}_n$.

**Augmented posterior over GP density** With Eq. (8) we obtain the joint posterior over the GP $g$, the rate scaling $\lambda$, the marked Poisson process $\Pi$, and the Pólya–Gamma variables at the observations $\boldsymbol{\omega}_N$ as

$$p(\boldsymbol{\omega}_N,\Pi,\lambda,g|\mathcal{D}) \propto p(\mathcal{D},\boldsymbol{\omega}_N,\Pi,\lambda|g)p(g). \quad (9)$$

In the following, this new representation will be used to derive two inference algorithms.

[2] Densities such as $p_{\phi_\lambda}(\Pi|\lambda)$ could be understood as the Radon–Nykodym derivative [12] of the corresponding probability measure with respect to some fixed dominating measure. However, we will not need an explicit form here.

# 3 INFERENCE

We will first derive an efficient Gibbs sampler which (asymptotically) solves the inference problem exactly, and then a variational mean-field algorithm, which only finds an approximate solution, but in a much faster time.

## 3.1 GIBBS SAMPLER

Gibbs sampling [13] generates samples from the posterior by creating a Markov chain, where at each time, a block of variables is drawn from the conditional posterior given all the other variables. Hence, to perform Gibbs sampling, we have to derive these conditional distributions for each set of variables from Eq. (9). Most of the following results are easily obtained by direct inspection. The only non–trivial case is the conditional distribution over the latent point process $\Pi$.

**Pólya-Gamma variables at observations** The conditional posterior over the set of Pólya–Gamma variables $\boldsymbol{\omega}_N$ depends only on the function $g$ at the observations $\{g(\boldsymbol{x}_n)\}_{n=1}^N$ and turns out to be

$$p(\boldsymbol{\omega}_N|\boldsymbol{g}) = \prod_{n=1}^N p_{\mathrm{PG}}(\omega_n|1,g(\boldsymbol{x}_n)), \quad (10)$$

where we have used the definition of a tilted Pólya-Gamma density in Eq. (4). This density can be efficiently sampled by methods developed by [9][3].

**Rate scaling** The rate scaling $\lambda$ has a conditional Gamma density given by

$$\mathrm{Gamma}(\lambda|\alpha,1) = \frac{(\lambda)^{\alpha-1}\mathrm{e}^{-\lambda}}{\Gamma(\alpha)}. \quad (11)$$

with $\alpha = |\Pi| + N = M + N$. Hence, the posterior is dependent on the number of observations and the number on events of the marked Poisson process $\Pi$.

**Posterior Gaussian process** Due to the form of the augmented likelihood the conditional posterior for the GP $\boldsymbol{g}_{N+M}$ at the observations $\{\boldsymbol{x}_n\}_{n=1}^N$ and the latent events $\{\boldsymbol{x}_m\}_{m=1}^M$ is a multivariate Gaussian density

$$p(\boldsymbol{g}_{N+M}|\Pi,\boldsymbol{\omega}_N) = \mathcal{N}(\boldsymbol{\mu}_{N+M},\Sigma_{N+M}), \quad (12)$$

with covariance matrix $\Sigma_{N+M} = [D + K_{N+M}^{-1}]^{-1}$. The diagonal matrix $D$ has its first $N$ entries given by $\boldsymbol{\omega}_N$ followed by $M$ entries being $\{\omega_m\}_{m=1}^M$. The mean is $\boldsymbol{\mu}_{N+M} = \Sigma_{N+M}\left[\boldsymbol{u} + K_{N+M}^{-1}\boldsymbol{\mu}_0^{(N+M)}\right]$, where the

[3] The sampler implemented by [14] is used for this work.

first $N$ entries of $N+M$ dimensional vector $\boldsymbol{u}$ are $1/2$ and the rest are $-1/2$. $K_{N+M}$ is the prior covariance kernel matrix of the GP evaluated at the observed points $\boldsymbol{x}_n$ and the latent events $\boldsymbol{x}_m$, and $\boldsymbol{\mu}_0^{(N+M)}$ is an $N+M$ dimensional vector with all entries being $\mu_0$.

The predictive conditional posterior for the GP for any set of points in $\mathcal{X}$ is simply given via the conditional prior $p(g|\boldsymbol{g}_{N+M})$, which has a well known form and can be found in [1].

**Sampling the latent marked point process** We easily find that the conditional posterior of the marked point process is given by

$$p(\Pi|g,\lambda)=\frac{\prod_{\omega,\boldsymbol{x}\in\Pi} e^{f(\omega,-g(\boldsymbol{x}))} p_{\phi_\lambda}(\Pi|\lambda)}{\exp\left(\int_{\mathcal{X}\times\mathbb{R}^+}\left(e^{f(\omega,-g(\boldsymbol{x}))}-1\right)\phi_\lambda(\boldsymbol{x},\omega)d\omega d\boldsymbol{x}\right)}, \quad (13)$$

where the form of the normalising denominator is obtained using Eq. (6). By computing the characteristic function of this conditional point process (see App. A) we can show that it is again a marked Poisson process with intensity

$$\Lambda(\boldsymbol{x},\omega) = \lambda\pi(\boldsymbol{x})\sigma(-g(\boldsymbol{x}))p_{\mathrm{PG}}(\omega|1,g(\boldsymbol{x})). \quad (14)$$

To sample from this process we first draw Poisson events $\boldsymbol{x}_m$ in the original data space $\mathcal{X}$ using the rate $\int_{\mathbb{R}^+}\Lambda(\boldsymbol{x},\omega)d\omega = \lambda\pi(\boldsymbol{x})\sigma(-g(\boldsymbol{x}))$ [11, chap. 5]. Subsequently for each event $\boldsymbol{x}_m$ a mark $\omega_m$ is generated from the conditional density $\omega_m \sim p_{\mathrm{PG}}(\omega|1,g(\boldsymbol{x}_m))$.

To sample the events $\{\boldsymbol{x}_m\}_{m=1}^M$, we use the well known approach of *thinning* [4]. We note, that the rate is upper bounded by the base measure $\lambda\pi(\boldsymbol{x})$. Hence, we first generate points $\tilde{\boldsymbol{x}}_m$ from a Poisson process with intensity $\lambda\pi(\boldsymbol{x})$. This is easily achieved by noting that the required number $M_{\max}$ of such events is Poisson distributed with mean parameter $\int_{\mathcal{X}}\lambda\pi(\boldsymbol{x})dx = \lambda$. The position of the events can then be obtained by sampling $\{\tilde{\boldsymbol{x}}_m\}_{m=1}^{M_{\max}}$ independent points from the base density $\tilde{\boldsymbol{x}}_m \sim \pi(\boldsymbol{x})$. These events are *thinned* by keeping each point $\tilde{\boldsymbol{x}}_m$ with probability $\sigma(-g(\tilde{\boldsymbol{x}}_m))$. The kept events constitute the final set $\{\boldsymbol{x}_m\}_{m=1}^M$.

**Sampling hyperparameters** In this work we will consider specific functional forms for the kernel $k(\boldsymbol{x},\boldsymbol{x}')$ and the base measure $\pi(\boldsymbol{x})$ which are parametrised by hyperparameters $\boldsymbol{\theta}_k$ and $\boldsymbol{\theta}_\pi$. These will be sampled by a Metropolis-Hastings method [15]. The GP prior mean $\mu_0$ can be directly sampled from the conditional posterior given $\boldsymbol{g}_{M+N}$. In this work, the hyperparameters are sampled every $v = 10$ step. Different choices of $v$ might yield faster convergence of the Markov Chain. Pseudo code for the Gibbs sampler is provided in Alg. 1.

**Algorithm 1:** Gibbs sampler for GP density model.

**Init:** $\{\boldsymbol{x}_m\}_{m=1}^M$, $\boldsymbol{g}_{N+M}$, $\lambda$, and $\boldsymbol{\theta}_k$, $\boldsymbol{\theta}_\pi$, $\mu_0$
**for** *Length of Markov chain* **do**
  **Sample PG variables at** $\{\boldsymbol{x}_m\}$: $\boldsymbol{\omega}_N \sim$ Eq. (10)
  **Sample latent Poisson process**: $\Pi \sim$ Eq. (13)
  **Sample rate scaling**: $\lambda \sim$ Eq. (11)
  **Sample GP**: $\boldsymbol{g}_{N+M} \sim$ Eq. (12)
  **Sample hyperparameters**: Every $v^{\mathrm{th}}$ sample with Metropolis–Hastings
**end**

### 3.2 VARIATIONAL BAYES

While expected to be more efficient than a Metropolis-Hastings sampler based on the unaugmented likelihood [7], the Gibbs sampler is practically still limited. The main computational bottleneck comes from the sampling of the conditional Gaussian over function values of $g$. The computation of the covariances requires the inversion of matrices of dimensions $N+M$, with a complexity $\mathcal{O}((N+M)^3)$. Hence the algorithm does not only become infeasible, when we have many observations, i.e when $N$ is large, but also if the sampler requires many thinned events, i.e. if $M$ is large. This can happen in particular for bad choices of the base measure $\pi(\boldsymbol{x})$. In the following, we introduce a variational Bayes algorithm [16], which solves the inference problem approximately, but with a complexity which scales linearly in the data size and is independent of structure.

**Structured mean–field approach** The idea of variational inference [16] is to approximate an intractable posterior $p(Z|\mathcal{D})$ by a simpler distribution $q(Z)$ from a tractable family. $q(Z)$ is optimised by minimising the Kullback-Leibler divergence between $q(Z)$ and $p(Z|\mathcal{D})$ which is equivalent to maximising the so called *variational lower bound* (sometimes also called ELBO for evidence lower bound) given by

$$\mathcal{L}(q(Z)) = \mathbb{E}_Q\left[\ln\frac{p(Z,\mathcal{D})}{q(Z)}\right] \leq \ln p(\mathcal{D}), \quad (15)$$

where $Q$ denotes the probability measure with density $q(Z)$. A common approach for variational inference is a structured mean–field method, where dependencies between sets of variables are neglected. For the problem at hand we assume that

$$q(\boldsymbol{\omega}_N,\Pi,g,\lambda) = q_1(\boldsymbol{\omega}_N,\Pi)q_2(g,\lambda). \quad (16)$$

A standard result for the variational mean–field approach shows that the optimal independent factors, which max-

imise the lower bound in Eq. (15) are given by

$$\ln q_1(\boldsymbol{\omega}_N, \Pi) = \mathbb{E}_{Q_2}\left[\ln p(\mathcal{D}, \boldsymbol{\omega}_N, \Pi, \lambda, g)\right] + \text{const.}, \tag{17}$$

$$\ln q_2(g, \lambda) = \mathbb{E}_{Q_1}\left[\ln p(\mathcal{D}, \boldsymbol{\omega}_N, \Pi, \lambda, g)\right] + \text{const.} \tag{18}$$

By inspecting Eq. (9), (17), and (18) it turns out that the densities of all four sets of variables factorise as

$$q_1(\boldsymbol{\omega}_N, \Pi) = q_1(\boldsymbol{\omega}_N) q_1(\Pi),$$
$$q_2(g, \lambda) = q_2(g) q_2(\lambda).$$

We will optimise the factors by a straightforward iterative algorithm, where each factor is updated given expectations over the others based on the previous step. Hence, the lower bound in Eq. (15) is increased in each step. Again we will see that the augmented likelihood in Eq. (8) allows for analytic solutions of all required factors.

**Pólya–Gamma variables at the observations** Similar to the Gibbs sampler, the variational posterior of the Pólya-Gamma variables at the observations is a product of tilted Pólya–Gamma densities given by

$$q_1(\boldsymbol{\omega}_N) = \prod_{n=1}^{N} p_{\mathrm{PG}}(\omega_n | 1, c_n), \tag{19}$$

with $c_n = \sqrt{\mathbb{E}_{Q_2}\left[g(\boldsymbol{x}_n)^2\right]}$. The only difference is, that the second argument of $p_{\mathrm{PG}}$ depends on the expectation of the square of $g(\boldsymbol{x}_n)$.

**Posterior marked Poisson process** Similar to the corresponding result for the Gibbs sampler we can show[4] that the optimal latent point process $\Pi$ is a Poisson process with rate given by

$$\begin{aligned}\Lambda_1(\boldsymbol{x}, \omega) =& \lambda_1 \pi(\boldsymbol{x}) \sigma(-c(\boldsymbol{x})) p_{\mathrm{PG}}(\omega | 1, c(\boldsymbol{x})) \\ &\times e^{(c(x) - g_1(\boldsymbol{x}))/2}\end{aligned} \tag{20}$$

with $\lambda_1 = e^{\mathbb{E}_{Q_2}[\ln \lambda]}$, $c(\boldsymbol{x}) = \sqrt{\mathbb{E}_{Q_2}\left[f(\boldsymbol{x})^2\right]}$, and $g_1(\boldsymbol{x}) = \mathbb{E}_{Q_2}\left[g(x)\right]$. Note also the similarity to the Gibbs sampler in Eq. (14).

**Optimal posterior for rate scaling** The posterior for the rate scaling $\lambda$ is a Gamma distribution given by

$$q_2(\lambda) = \mathrm{Gamma}(\lambda | \alpha_2, 1) = \frac{\lambda^{\alpha_2 - 1} \mathrm{e}^{-\lambda}}{\Gamma(\alpha_2)}, \tag{21}$$

where $\alpha_2 = N + \mathbb{E}_{Q_1}\left[\sum_{\boldsymbol{x}' \in \Pi} \delta(\boldsymbol{x} - \boldsymbol{x}')\right]$, and $\mathbb{E}_{Q_1}\left[\sum_{\boldsymbol{x}' \in \Pi} \delta(\boldsymbol{x} - \boldsymbol{x}')\right] = \int_{\mathcal{X}} \int_{\mathbb{R}^+} \Lambda_1(\boldsymbol{x}, \omega) d\omega d\boldsymbol{x}$, and $\delta(\cdot)$ is the Dirac delta function. The integral is solved by importance sampling as will be explained (see Eq. (25)).

[4] The proof is similar to the one from App. A.

**Approximation of GP via sparse GP** The optimal variational form for the posterior $g$ is a GP given by

$$q_2(g) \propto e^{U(g)} p(g),$$

where $U(g) = \mathbb{E}_{Q_1}\left[\ln p(\mathcal{D}, \boldsymbol{\omega}_N, \Pi, \lambda | g)\right]$ results in the Gaussian log–likelihood

$$U(g) = -\frac{1}{2} \int_{\mathcal{X}} A(\boldsymbol{x}) g(\boldsymbol{x})^2 d\boldsymbol{x} + \int_{\mathcal{X}} B(\boldsymbol{x}) g(\boldsymbol{x}) d\boldsymbol{x} + \text{const.}$$

with

$$A(\boldsymbol{x}) = \sum_{n=1}^{N} \mathbb{E}_{Q_1}\left[\omega_n\right] \delta(\boldsymbol{x} - \boldsymbol{x}_n) + \int_{\mathbb{R}^+} \omega \Lambda_1(\boldsymbol{x}, \omega) d\omega,$$

$$B(\boldsymbol{x}) = \frac{1}{2} \sum_{n=1}^{N} \delta(\boldsymbol{x} - \boldsymbol{x}_n) - \frac{1}{2} \int_{\mathbb{R}^+} \Lambda_1(\boldsymbol{x}, \omega) d\omega.$$

For general GP priors, this free form optimum is intractable by the fact that the likelihood depends on $g$ at infinitely many points. Hence, we resort to an additional approximation which makes the dimensionality of the problem again finite. The well known framework of *sparse* GPs [17, 18, 19] turns out to be useful in this case. This has been introduced for likelihoods with large, but finite dimensional likelihoods [19, 20] and later generalised to infinite dimensional problems [21, 22]. The sparse approximation assumes a variational posterior of the form

$$q_2(g) = p(g | \boldsymbol{g}_s) q_2(\boldsymbol{g}_s),$$

where $\boldsymbol{g}_s$ is the GP evaluated at a finite set of *inducing points* $\{\boldsymbol{x}_l\}_{l=1}^{L}$ and $p(g|\boldsymbol{g}_s)$ is the conditional prior. A variational optimisation yields

$$q_2(\boldsymbol{g}_s) \propto e^{U^s(\boldsymbol{g}_s)} p(\boldsymbol{g}_s), \tag{22}$$

where the first term can be seen as a new 'effective' likelihood only depending on the inducing points. This new (log) likelihood is given by

$$\begin{aligned}&U^s(\boldsymbol{g}_s) = \mathbb{E}_P\left[U(g) | \boldsymbol{g}_s\right] = \\ &-\frac{1}{2} \int_{\mathcal{X}} A(\boldsymbol{x}) \tilde{g}_s(\boldsymbol{x})^2 d\boldsymbol{x} + \int_{\mathcal{X}} B(\boldsymbol{x}) \tilde{g}_s(\boldsymbol{x}) d\boldsymbol{x} + \text{const.},\end{aligned}$$

with $\tilde{g}_s(\boldsymbol{x}) = \mu_0 + \boldsymbol{k}_s(\boldsymbol{x})^\top K_s^{-1}(\boldsymbol{g}_s - \boldsymbol{\mu}_0^{(L)})$, $\boldsymbol{k}_s(\boldsymbol{x})$ being an $L$ dimensional vector, where the $l^{\text{th}}$ entry is $k(\boldsymbol{x}, \boldsymbol{x}_l)$ and $K_s$ being the prior covariance matrix for all inducing points. The expectation is computed with respect to the GP prior conditioned on the sparse GP $\boldsymbol{g}_s$. We identify Eq. (22) being a multivariate normal distribution with covariance matrix

$$\Sigma_2^s = \left[K_s^{-1} \int_{\mathcal{X}} A(\boldsymbol{x}) \boldsymbol{k}_s(\boldsymbol{x})^\top \boldsymbol{k}_s(\boldsymbol{x}) d\boldsymbol{x}\ K_s^{-1} + K_s^{-1}\right]^{-1}, \tag{23}$$

**Algorithm 2:** Variational Bayes algorithm for GP density model

**Init:** Inducing points, $q_2(\boldsymbol{g}_s)$, $q_2(\lambda)$, and $\boldsymbol{\theta}_k, \boldsymbol{\theta}_\pi, \mu_0$
**while** $\mathcal{L}$ *not converged* **do**
  **Update** $q_1$
    **PG distributions at observations**: $q_1^*(\boldsymbol{\omega}_N)$ with Eq. (19)
    **Rate of latent process**: $\Lambda_1(\boldsymbol{x}, \omega)$ with Eq. (20)
  **Update** $q_2$
    **Rate scaling**: $\alpha_2$ with Eq. (21)
    **Sparse GP**: $\Sigma_2^s, \mu_2^s$ with Eq. (23), (24)
  **Update** $\boldsymbol{\theta}_k, \boldsymbol{\theta}_\pi, \mu_0$ **with gradient update**
**end**

and mean

$$\boldsymbol{\mu}_2^s = \Sigma_2^s \left( K_s^{-1} \int_{\mathcal{X}} \boldsymbol{k}_s(\boldsymbol{x}) \tilde{B}(\boldsymbol{x}) d\boldsymbol{x} + K_s^{-1} \boldsymbol{\mu}_0^{(L)} \right), \tag{24}$$

with $\tilde{B}(\boldsymbol{x}) = B(\boldsymbol{x}) - A(x)(\mu_0 - \boldsymbol{k}_s(\boldsymbol{x})^\top K_s^{-1} \boldsymbol{\mu}_0^{(L)})$.

**Integrals over $\boldsymbol{x}$** The sparse GP approximation and the posterior over $\lambda$ in Eq. (21) requires the computation of integrals of the form

$$I \doteq \int_{\mathcal{X}} \int_{\mathbb{R}^+} y(\boldsymbol{x}, \omega) \Lambda_1(\boldsymbol{x}, \omega) d\omega d\boldsymbol{x},$$

with specific functions $y(\boldsymbol{x}, \omega)$. For these functions, the inner integral over $\omega$ can be computed analytically, but the outer one over the space $\mathcal{X}$ has to be treated numerically. We approximate it via importance sampling

$$I \approx \frac{1}{R} \sum_{r=1}^{R} \int_{\mathbb{R}^+} y(\boldsymbol{x}_r, \omega_r) \frac{\Lambda_1(\boldsymbol{x}_r, \omega_r)}{\pi(\boldsymbol{x}_r)} d\omega_r, \tag{25}$$

where every sample point $\boldsymbol{x}_r$ is independently drawn from the base measure $\pi(\boldsymbol{x})$.

**Updating hyperparameters** Having an analytic solution for every factor of the variational posterior in Eq. (16) we further require the optimisation of hyperparameters. $\boldsymbol{\theta}_k$, $\boldsymbol{\theta}_\pi$ and $\mu_0$ are optimised by maximising the lower bound in Eq. (15) (see App. B for explicit form) with a gradient ascent algorithm having an adaptive learning rate (Adam) [23]. Additional hyperparameters are the locations of inducing points $\{\boldsymbol{x}_l\}_{l=1}^L$. Half of them are drawn randomly from the initial base measure, while half of them are positioned on regions with a high density of observations found by a k–means algorithm. Pseudo code for the complete variational algorithm is provided in Alg. 2.

Python code for Alg. 1 and 2 is provided at [24].

# 4 RESULTS

To test our two inference algorithms, the Gibbs sampler and the variational Bayes algorithm (VB), we will first evaluate them on data drawn from the generative model. Then we compare both on an artificial dataset and several real datasets. We will only consider cases with $\mathcal{X} = \mathbb{R}^d$. To evaluate the quality of inference we consider always the logarithm of the expected test likelihood

$$\ell_{\text{test}}(\tilde{\mathcal{D}}) \doteq \ln \left( \mathbb{E} \left[ \prod_{\boldsymbol{x} \in \tilde{\mathcal{D}}} \rho(\boldsymbol{x}) \right] \right),$$

where $\tilde{\mathcal{D}}$ is test data unknown to the inference algorithm and the expectation is over the inferred posterior measure. In practice we sample this expectation from the inferred posterior over $g$. Since this quantity involves an integral, that is again approximated by Eq. (25), we check that the standard deviation $\text{std}(I)$ is less than $1\%$ of the value of the estimated value $I$.

**Data from generative model.** We generate datasets according to Eq. (1), where $g$ is drawn from the GP prior with $\mu_0 = 0$. As covariance kernel we assume a squared exponential throughout this work

$$k(\boldsymbol{x}, \boldsymbol{x}') = \theta_k^{(0)} \prod_{i=1}^{d} \exp\left( -\frac{(x_i - x_i')^2}{2(\theta_k^{(i)})^2} \right).$$

The base measure $\pi(\boldsymbol{x})$ is a standard normal density. We use the algorithm described in [7] to generate exact samples. In this section, the hyperparameters $\boldsymbol{\theta}_k, \boldsymbol{\theta}_\pi$ and $\mu_0$ are fixed to the true values for inference. Unless stated otherwise for the VB the number of inducing points is fixed to 200 and the number of integration points for importance sampling to $5 \times 10^3$. For the Gibbs sampler, we sample a Markov chain of $5 \times 10^3$ samples after a burn–in period of $2 \times 10^3$ samples.

In Fig. 1 we see a 1 dimensional example dataset, where both inference algorithms recover well the structure of the underlying density. The inferred posterior means are barely distinguishable. However, evaluating the inferred densities on an unseen test set, we note that the Gibbs sampler performs slightly better. Of course, this is expected since the sampler provides exact inference for the generative model and should (on average) not be outperformed by the approximate VB as long as the sampled Markov chain is long enough. In Fig. 1 (bottom left) we see that only 13 iterations of the VB are required to meet the convergence criterion. For Markov chain samplers to be efficient, correlations between samples should decay quickly. Fig. 1 (bottom middle) shows the autocorrelation of $\ell_{\text{test}}$, which was evaluated at each sample of the

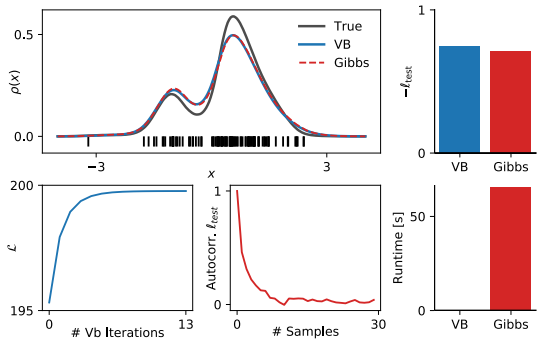



Figure 1: **1D data from the generative model.** Data consist of 100 samples from the underlying density sampled from the GP density model. **Upper left:** True density (black line), data (black vertical bars), mean posterior density inferred by Gibbs sampler (red dashed line) and VB algorithm (blue line). **Upper right:** Negative log expected test likelihood of Gibbs and VB inferred posterior. **Lower left:** Variational lower bound as function of iterations of the VB algorithm. **Lower middle:** Autocorrelation of test likelihood as function of Markov chain samples obtained from Gibbs sampler. **Lower right:** Runtime of the two algorithms (VB took 0.3 s).

| Dim | # points | Gibbs | | VB | |
|---|---|---|---|---|---|
| | | $\ell_{\text{test}}$ | $T$ [s] | $\ell_{\text{test}}$ | $T$ [s] |
| 1 | 50 | -146.9 | 30.1 | -149.2 | 1.13 |
| 2 | 100 | -257.0 | 649.9 | -260.2 | 2.03 |
| 2 | 200 | -285.3 | 546.1 | -289.6 | 1.41 |
| 6 | 400 | -823.9 | 4667 | -822.2 | 0.89 |

Table 1: **Performance of Gibbs sampler and VB** on different datasets sampled from generative model. $\ell_{\text{test}}$ was evaluated on a unknown test set including 50 samples. In addition, runtime $T$ is reported in seconds.

Markov chain. After about 10 samples the correlations reach a plateau close to 0, demonstrating excellent mixing properties of the sampler. Comparing the run time of both algorithms, VB (0.3 s) outperforms the sampler $\sim 1$ min by more than 2 orders of magnitude.

To demonstrate the inference for more complicated problems, 2 dimensional data are generated with 200 samples (Fig. 2). The posterior mean densities inferred by both algorithms capture the structure well. As before, the log expected test likelihood is larger for the Gibbs sampler ($\ell_{\text{test}} = -296.2$) compared to VB ($\ell_{\text{test}} = -306.0$). However, the Gibbs sampler took $> 20$ min while the VB required only 1.8 s to obtain the result.

In Tab. 1 we show results for datasets with different size and different dimensionality. The results confirm that the

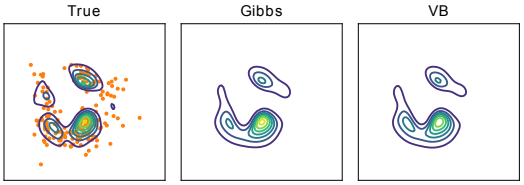



Figure 2: **2D data from generative model. Right:** 200 samples from the underlying two dimensional density. **Middle:** Posterior mean of Gibbs sampler inferred density. **Right:** Posterior mean of VB inferred density.

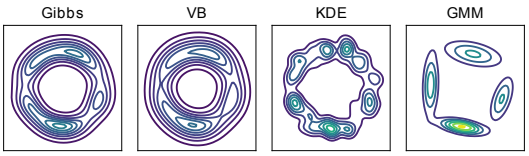



Figure 3: **Comparison to other density estimation methods on artificial 2D data.** Training data consist of 100 data points uniformly distributed on a circle (1.5 radius) and additional Gaussian noise (0.2 std.). **From left to right:** The posterior mean inferred by Gibbs sampler and VB algorithm, followed by density estimation using KDE and GMM.

run time for the Gibbs sampler scales strongly with size and dimensionality of a problem, while the VB algorithm seems relatively unaffected in this regard. However, the VB is in general outperformed by the sampler in terms of expected test likelihood or in the same range. Note, that the runtime of the Gibbs sampler does not solely depend on the number of observed data points $N$ (compare data set 2 and 3 in Tab. 1). As discussed earlier this can happen, when the base measure $\pi(\boldsymbol{x})$ is very different from the target density $\rho(\boldsymbol{x})$ resulting in many latent Poisson events (i.e. $M$ is large).

**Circle data** In the following, we compare the GP density model and its two inference algorithms with two alternative density estimation methods. These are given by a kernel density estimator (KDE) with a Gaussian kernel and a Gaussian mixture model (GMM) [25]. The free parameters of these models (kernel bandwidth for KDE and number of components for GMM) are optimised by 10-fold cross–validation. Furthermore, GMM is initialised 10 times and the best result is reported. For the GP density model a Gaussian density is assumed as base measure $\pi(\boldsymbol{x})$, and hyperparameters $\boldsymbol{\theta}_\pi$, $\boldsymbol{\theta}_k$, and $\mu_0$ are now optimised. Similar to [7] we consider 100 samples uniformly drawn from a circle with additional Gaussian

| | Gibbs | VB | KDE | GMM |
|---|---|---|---|---|
| $\ell_{\text{test}}$ | -220.31 | -230.53 | -228.43 | -237.34 |

Table 2: **Log expected test likelihood for circle data.**

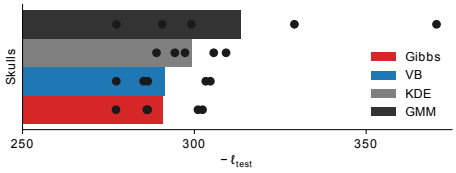



Figure 4: **Performance on 'Egyptian Skulls' dataset [26].** 100 training points and 4 dimensions. Bar height shows average negative log test likelihood obtained by five random permutations of training and test set and points mark single permutation results.

noise. The inferred densities (only the mean of the posterior for Gibbs and VB) are shown in Fig. 3. Both GP density methods recover well the structure of the data, but the VB seems to overestimate the width of the Gaussian noise compared to the Gibbs sampler. While the KDE also recovers relatively well the data structure the GMM fails in this case. This is also reflected on the log expected test likelihoods (Tab. 2).

**Real data sets** The 'Egyptian Skulls' dataset [26] contains 150 data points in 4 dimensions. 100 training points are randomly selected and performance is evaluated on the remaining ones. Before fitting data is whitened. Base measure and fitting procedure for all algorithms are the same as for the circular data. Furthermore, fitting is done for 5 random permutations of training and test set. The results in Fig. 4 show that both algorithms for the GP density model outperform the two other ones on this dataset.

Often practical problems may consist of many more data points and dimensions. As discussed, the Gibbs sampler is not practical for such kind of problems, while the VB could handle larger amounts of data. Unfortunately, the sparsity assumption and the integration via importance sampling is expected to become poorer with increasing number of dimensions. Noting, however, that the 'effective' dimensionality in our model is determined by the base measure $\pi(\boldsymbol{x})$, one can circumvent this problem by an educated choice of $\pi(\boldsymbol{x})$ if data $\mathcal{D}$ lie in a submanifold of the high dimensional space $\mathcal{X}$.

We employ this strategy by first fitting a GMM to the problem and then utilising the fit as base measure. In Fig. 5 we consider 3 different datasets[5] to test this procedure. As in Fig. 4, fitting is repeated 5 times for random permutations if training and test set. For the 'Thyroid' dataset, one of the 5 fits is excluded, because the importance sampling yielded poor approximation $\text{std}(I) > I \times 10^{-2}$. The training sets contain 400 to 6000 data points with 5 to 9 dimensions. The results for KDE are not reported, since it is always outperformed by the GMM. Fig. 5 demonstrates combining the GMM and VB algorithm results in an improvement of the log test likelihood $\ell_{\text{test}}$ compared to using only GMM. Average relative improvements of $\ell_{\text{test}}$ are 8.9 % for 'Forest Fire', 4.1 % for 'Thyroid', and 1.1 % for 'Wine' dataset.


Figure 5: **Application on higher dimensional data with many data points.** The improvement on log expected test likelihood $\ell_{\text{test}}$ per test point compared to GMM, when using same as base measure $\pi(\boldsymbol{x})$ for the VB inference. **From top to bottom:** 'Forest Fire' dataset [27, 28] (400 training points, 117 test points, 5 dim.), 'Thyroid' dataset [29] ($3 \times 10^3$, 772, 6), 'Wine' dataset [27] ($6 \times 10^3$, 498, 9). Bars mark improvement on average of random permutations of training and test set while points mark single runs.

## 5 DISCUSSION

We have shown how inference for a nonparametric, GP based, density model can be made efficient. In the following we would like to discuss various possible extensions but also limitations of our approach.

**Choice of base measure** As we have shown for applications to real data, the choice of the base measure is quite important, especially for the sampler and for high dimensional problems. While many datasets might favour a normal distribution as base measure, problems with outliers might favour fat tailed densities. In general, any density which can be evaluated on the data space $\mathcal{X}$ and which allows for efficient sampling, is a valid choice as base measure $\pi(\boldsymbol{x})$ in our inference approach for the GP density model. Any powerful density estima-

[5] Only real valued dimensions are considered and for the 'forest fire' dataset dimensions are excluded, where data have more than half 0 entries.

tor which fulfils this condition could provide a base measure which could then potentially be improved by the GP model. It would e.g. be interesting to apply this idea to neural networks [30, 31] based estimators. Other generalisations of our model could consider alternative data spaces $\mathcal{X}$. One might e.g. think of specific discrete and structured sets $\mathcal{X}$ for which appropriate Gaussian processes could be defined by suitable Mercer kernels.

**Big data & high dimensionality** Our proposed Gibbs sampler suffers from cubic scaling in the number of data points and is found to be already impractical for problems with hundreds of observations. This could potentially be tackled by using sparse (approximate) GP methods for the sampler (see [32] for a potential approach). On the other hand, the proposed VB algorithm scales only linearly with the training set size and can be applied to problems with several thousands of observations. The integration of stochastic variational inference into our method could potentially increase this limit [33].

Potential limitations of the GP density model are given by high dimensional problems. If approached naively, the combination of the sparse GP approximation and the numerical integration using importance sampling is expected to yield bad approximations in such cases.[6] If the data is concentrated on a low dimensional submanifold of the high–dimensional space, one could still try to combine our method with other density estimators providing a base measure $\pi(\boldsymbol{x})$ that is adapted to this submanifold, to allow for tractable GP inference.

**Acknowledgements**

CD was supported by the Deutsche Forschungsgemeinschaft (GRK1589/2) and partially funded by Deutsche Forschungsgemeinschaft (DFG) through grant CRC 1294 "Data Assimilation", Project (A06) "Approximative Bayesian inference and model selection for stochastic differential equations (SDEs)".

## References

[1] Carl Edward Rasmussen and Christopher KI Williams. *Gaussian processes for machine learning*, volume 1. MIT press Cambridge, 2006.

[2] Christopher KI Williams and Carl Edward Rasmussen. Gaussian processes for regression. In *Advances in neural information processing systems*, pages 514–520, 1996.

[3] Hannes Nickisch and Carl Edward Rasmussen. Approximations for binary gaussian process classification. *Journal of Machine Learning Research*, 9(Oct):2035–2078, 2008.

[4] Ryan Prescott Adams, Iain Murray, and David JC MacKay. Tractable nonparametric bayesian inference in poisson processes with gaussian process intensities. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 9–16. ACM, 2009.

[5] Cedric Archambeau, Dan Cornford, Manfred Opper, and John Shawe-Taylor. Gaussian process approximations of stochastic differential equations. In *Gaussian Processes in Practice*, pages 1–16, 2007.

[6] Andreas Damianou, Michalis K Titsias, and Neil D Lawrence. Variational gaussian process dynamical systems. In *Advances in Neural Information Processing Systems*, pages 2510–2518, 2011.

[7] Iain Murray, David MacKay, and Ryan P Adams. The gaussian process density sampler. In *Advances in Neural Information Processing Systems*, pages 9–16, 2009.

[8] Jaakko Riihimäki, Aki Vehtari, et al. Laplace approximation for logistic gaussian process density estimation and regression. *Bayesian analysis*, 9(2):425–448, 2014.

[9] Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using pólya–gamma latent variables. *Journal of the American statistical Association*, 108(504):1339–1349, 2013.

[10] Stephen G Walker. Posterior sampling when the normalizing constant is unknown. *Communications in Statistics—Simulation and Computation®*, 40(5):784–792, 2011.

[11] John Frank Charles Kingman. *Poisson processes*. Wiley Online Library, 1993.

[12] Takis Konstantopoulos, Zurab Zerakidze, and Grigol Sokhadze. *Radon–Nikodým Theorem*, pages 1161–1164. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[13] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. In *Readings in Computer Vision*, pages 564–584. Elsevier, 1987.

[14] Scott Linderman. pypolyagamma. `https://github.com/slinderman/pypolyagamma`, 2017.

[6] Potentially in such cases other sparsity methods [34] might be more favourable.

[15] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[16] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[17] Lehel Csató. Gaussian Processes -Iterative Sparse Approximations. *PhD Thesis*, 2002.

[18] Lehel Csató and Manfred Opper. Sparse online gaussian processes. *Neural Computation*, 14(3):641–668, 2002.

[19] Michalis K Titsias. Variational learning of inducing variables in sparse gaussian processes. In *International Conference on Artificial Intelligence and Statistics*, pages 567–574, 2009.

[20] Edward Snelson and Zoubin Ghahramani. Sparse gaussian processes using pseudo-inputs. In *Advances in neural information processing systems*, pages 1257–1264, 2006.

[21] Alexander G de G Matthews, James Hensman, Richard Turner, and Zoubin Ghahramani. On sparse variational methods and the kullback-leibler divergence between stochastic processes. In *Artificial Intelligence and Statistics*, pages 231–239, 2016.

[22] Philipp Batz, Andreas Ruttor, and Manfred Opper. Approximate bayes learning of stochastic differential equations. *arXiv preprint arXiv:1702.05390*, 2017.

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[24] Christian Donner. Sgpd_inference. `https://github.com/christiando/SGPD_Inference`, 2018.

[25] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python . *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[26] David J Hand, Fergus Daly, K McConway, D Lunn, and E Ostrowski. *A handbook of small data sets*, volume 1. cRc Press, 1993.

[27] Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017.

[28] Paulo Cortez and Aníbal de Jesus Raimundo Morais. A data mining approach to predict forest fires using meteorological data. 2007.

[29] Fabian Keller, Emmanuel Muller, and Klemens Bohm. Hics: high contrast subspaces for density-based outlier ranking. In *Data Engineering (ICDE), 2012 IEEE 28th International Conference on*, pages 1037–1048. IEEE, 2012.

[30] Hugo Larochelle and Iain Murray. The neural autoregressive distribution estimator. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 29–37, 2011.

[31] Benigno Uria, Iain Murray, and Hugo Larochelle. A deep and tractable density estimator. In *International Conference on Machine Learning*, pages 467–475, 2014.

[32] Yves-Laurent Kom Samo and Stephen Roberts. Scalable nonparametric bayesian inference on point processes with gaussian processes. In *International Conference on Machine Learning*, pages 2227–2236, 2015.

[33] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

[34] Yarin Gal and Richard Turner. Improving the gaussian process sparse spectrum approximation by representing uncertainty in frequency inputs. In *International Conference on Machine Learning*, pages 655–664, 2015.

## A THE CONDITIONAL POSTERIOR POINT PROCESS

Here we prove that the conditional posterior point process in Equation (13) again is a Poisson process using Campbell's theorem [11, chap. 3]. For an arbitrary function $h(\cdot,\cdot)$ we set $H \doteq \sum_{(\boldsymbol{x},\omega)\in\Pi} h(\boldsymbol{x},\omega)$. We calculate the characteristic functional

$$\begin{aligned}
&\mathbb{E}_{\phi_\lambda}\left[e^H \middle| g,\lambda\right] = \\
&\frac{\mathbb{E}_{\phi_\lambda}\left[\prod_{(\omega,\boldsymbol{x})\in\Pi} e^{f(\omega,-g(\boldsymbol{x}))+h(\boldsymbol{x},\omega)}\middle| g,\lambda\right]}{\exp\left(\int_{\mathcal{X}\times\mathbb{R}^+}\left(e^{f(\omega,-g(\boldsymbol{x}))}-1\right)\phi_\lambda(\boldsymbol{x},\omega)d\omega d\boldsymbol{x}\right)} = \\
&\frac{\exp\left\{\int_{\mathcal{X}\times\mathbb{R}^+}\left(e^{f(\omega,-g(\boldsymbol{x}))+h(\boldsymbol{x},\omega)}-1\right)\phi_\lambda(\boldsymbol{x},\omega)d\omega d\boldsymbol{x}\right\}}{\exp\left(\int_{\mathcal{X}\times\mathbb{R}^+}\left(e^{f(\omega,-g(\boldsymbol{x}))}-1\right)\phi_\lambda(\boldsymbol{x},\omega)d\omega d\boldsymbol{x}\right)} = \\
&\exp\left\{\int_{\mathcal{X}\times\mathbb{R}^+}\left(e^{h(\boldsymbol{x},\omega)}-1\right)e^{f(\omega,-g)}\phi_\lambda(\boldsymbol{x},\omega)d\omega d\boldsymbol{x}\right\} = \\
&\exp\left\{\int_{\mathcal{X}\times\mathbb{R}^+}\left(e^{h(\boldsymbol{x},\omega)}-1\right)\Lambda(\boldsymbol{x},\omega)d\omega d\boldsymbol{x}\right\},
\end{aligned}$$

where the last equality follows from the definition of $\phi_\lambda(\boldsymbol{x},\omega)$ and the tilted Polya–Gamma density. Using the fact that a Poisson process is uniquely characterised by its generating function this shows that the conditional posterior $p(\Pi|g,\lambda)$ is a marked Poisson process.

## B VARIATIONAL LOWER BOUND

The full variational lower bound is given by

$$\begin{aligned}
\mathcal{L}(q) = \sum_{n=1}^{N} &\{\mathbb{E}_Q[\ln\lambda] + \ln\pi(\boldsymbol{x}_n) + \mathbb{E}_Q[f(\omega_n, g(\boldsymbol{x}_n))] \\
&- \ln\cosh\left(\frac{c_n}{2}\right) + \frac{c_n^2}{2}\mathbb{E}_Q[\omega_n]\Big\} \\
&+ \int_{\mathcal{X}}\int_{\mathbb{R}^+}\{\mathbb{E}_Q[\ln\lambda] + \mathbb{E}_Q[f(\omega,-g(\boldsymbol{x}))] - \ln\lambda_1 \\
&- \ln\sigma(-c(\boldsymbol{x})) - \ln\cosh\left(\frac{c(\boldsymbol{x})}{2}\right) - \frac{c(\boldsymbol{x})^2}{2}\omega \\
&- \frac{c(\boldsymbol{x}) - g_1(\boldsymbol{x})}{2} + 1\Big\}\Lambda_1(\boldsymbol{x},\omega)d\omega d\boldsymbol{x} \\
&- \mathbb{E}_Q[\lambda] + \mathbb{E}_Q\left[\ln\frac{p(\lambda)}{q(\lambda)}\right] + \mathbb{E}_Q\left[\ln\frac{p(\boldsymbol{g}_s)}{q(\boldsymbol{g}_s)}\right].
\end{aligned}$$

# Chapter 5

# Journal article: *Inverse Ising problem in continuous time: A latent variable approach*

Published in the journal *Physical Review E* (American Physical Society, United States).

Authors:
Christian Donner[1,2], Manfred Opper[1,2]
[1]: Technische Universität Berlin. [2]: Bernstein Center for Computational Neuroscience Berlin.

Details:
Submitted: September 2016
Accepted: May 2017
DOI: https://doi.org/10.1103/PhysRevE.96.062104
Pubmed-ID: 29347355


Python code on GitHub: https://github.com/christiando/dynamic_ising

PHYSICAL REVIEW E **96**, 062104 (2017)

# Inverse Ising problem in continuous time: A latent variable approach

Christian Donner* and Manfred Opper
*Artificial Intelligence Group, Technische Universität, Marchstr. 23, 10587 Berlin, Germany*
(Received 1 September 2017; published 4 December 2017; corrected 27 December 2017)

We consider the inverse Ising problem: the inference of network couplings from observed spin trajectories for a model with continuous time Glauber dynamics. By introducing two sets of auxiliary latent random variables we render the likelihood into a form which allows for simple iterative inference algorithms with analytical updates. The variables are (1) Poisson variables to linearize an exponential term which is typical for point process likelihoods and (2) Pólya-Gamma variables, which make the likelihood quadratic in the coupling parameters. Using the augmented likelihood, we derive an expectation-maximization (EM) algorithm to obtain the maximum likelihood estimate of network parameters. Using a third set of latent variables we extend the EM algorithm to sparse couplings via L1 regularization. Finally, we develop an efficient approximate Bayesian inference algorithm using a variational approach. We demonstrate the performance of our algorithms on data simulated from an Ising model. For data which are simulated from a more biologically plausible network with spiking neurons, we show that the Ising model captures well the low order statistics of the data and how the Ising couplings are related to the underlying synaptic structure of the simulated network.

DOI: 10.1103/PhysRevE.96.062104

## I. INTRODUCTION

In recent years, the inverse Ising problem, i.e., the reconstruction of couplings and external fields of an Ising model from samples of spin configurations, has attracted considerable interest in the physics community [1]. This is due to the fact that Ising models play an important role for data modeling with applications to neural spike data [2,3], protein structure determination [4], and gene expression analysis [5]. Much effort has been devoted to the development of algorithms for the *static* inverse Ising problem. This is a nontrivial task, because statistically efficient, likelihood-based methods become computationally infeasible by the intractability of the partition function of the model. Hence one has to resort to either approximate inference methods or to other statistical estimators such as pseudolikelihood methods [6] or the interaction screening algorithm [7]. The situation is somewhat simpler for the dynamical inverse Ising problem, which recently attracted attention [8–13]. If one assumes a Markovian dynamics, the exact normalization of the spin transition probabilities allows for an explicit computation of the likelihood if one has a complete set of observed data over time. Nevertheless, the model parameters enter the likelihood in a fairly complex way, and the application of more advanced statistical approaches such as Bayesian inference again becomes a nontrivial task. This is especially true for the continuous time kinetic Ising model where the spins are governed by Glauber dynamics [14]. With this dynamics the likelihood contains an exponential function related to the "nonflipping" times and makes analytical manipulations of the posterior distribution of parameters intractable. However, it is possible to compute the likelihood gradient to find the maximum likelihood estimate (MLE) [15].

In this paper we will show how the likelihood for the continuous time problem can be remarkably simplified by introducing a combination of two sets of auxiliary random variables. The first set of variables are Poisson random variables which "linearize" the aforementioned exponential term that appears naturally in likelihoods of Poisson point-process models [16]. These latent variables are related to previous work, where similar variables have been introduced for sampling the intensity function of an inhomogeneous Poisson process [17]. The second set of variables are the so-called Pólya-Gamma variables, which were introduced into statistics to enable efficient Bayesian inference for logistic regression [18] and which may not be familiar in the physics community. These variables have also been used recently for Monte Carlo–based Bayesian inference of discrete-time Markov models [19], model-based statistical testing of spike synchrony [20], and an expectation-maximization (EM) scheme for logistic regression [21].

With these latent variables the model parameters enter the resulting joint likelihood similarly to simple Gaussian models. We will use this formulation to construct iterative algorithms for a penalized maximum likelihood and for variational Bayes estimators which have simple analytically computable updates. We test our algorithms on artificial data. As an illustrative application we use the Bayes algorithm on data from a simulated recurrent network with conductance-based spiking neurons and show how the model reproduces the statistics of the data and how the obtained Ising parameters reflect the underlying synaptic structure.

The paper is organized as follows: In Sec. II the continuous time kinetic Ising model is introduced followed by a derivation of its likelihood in Sec. III. In Sec. IV we introduce auxiliary latent variables to simplify the likelihood. In Sec. V we develop an EM algorithm for maximum likelihood inference and extend it to L1-regularized likelihood maximization and a variational Bayes approximation. Finally, in Sec. VI we apply our method to simulated data generated from an Ising network and from a network of spiking neurons.

## II. THE MODEL

Following Ref. [15] in this section, we consider a system of $N$ Ising spins $s_i(t) \in \{-1,1\}$ for $i = 1,\ldots,N$. We denote

*Also at Bernstein Center for Computational Neuroscience; christian.donner@bccn-berlin.de

the vector of all spins by $\boldsymbol{s}(t) = (s_1(t), \ldots, s_N(t))^\top$. A spin $i$ is interacting with spin $j$ through a coupling $J_{ij}$. We are not assuming symmetry of these couplings: in general, we have $J_{ij} \neq J_{ji}$. We will also allow for *self-couplings* $J_{ii}$. The total field acting on spin $i$ is given by

$$H_i(t) = \theta_i + \sum_{j=1}^{N} J_{ij} s_j(t), \tag{1}$$

where $\theta_i$ denotes the *external field*. The Glauber dynamics of the spins is defined by asynchronous updates [15] where in a small time interval $\Delta t$, spins $i$ are selected independently with probability $\gamma \Delta t$ for an update; $\gamma > 0$ is the update rate. The updated spins are flipped, i.e., $s_i(t + \Delta t) = -s_i(t)$ with probability

$$P_i^{\text{flip}}(t) = \frac{\exp[-s_i(t) H_i(t)]}{2\cosh[H_i(t)]}. \tag{2}$$

The probability that spin $i$ is not flipped at time $t$ in the interval $\Delta t$ is given by $1 - \gamma \Delta t + \gamma \Delta t [1 - P_i^{\text{flip}}(t)]$. Hence, the total probability of a (time-discretized) temporal sequence $\{\boldsymbol{s}\}_{0:T}$ of spins during a time interval $[0 : T]$ is given by

$$P(\{\boldsymbol{s}\}_{0:T}|\boldsymbol{J}) = \prod_{(i,t)\in F} \left\{ \gamma \Delta t \frac{\exp[-s_i(t) H_i(t)]}{2\cosh[H_i(t)]} \right\} \times \prod_{(i,t)\in NF} \left\{ 1 - \gamma \Delta t + \gamma \Delta t \frac{\exp[s_i(t) H_i(t)]}{2\cosh[H_i(t)]} \right\}. \tag{3}$$

Here $F$ denotes the set of pairs $(i,t)$ where spin $i$ was flipped at time $t$, and $NF$ is the corresponding, complementary set of times and spins where no flips happened. $\boldsymbol{J}$ stands for the parameters of the model: $\boldsymbol{J} \equiv J_{ij}$ for $i, j = 1, \ldots N$ and $\theta_i$ for $i = 1, \ldots, N$.

## III. LIKELIHOOD AND INFERENCE

Our goal is to infer the couplings and external fields from observations of complete spin trajectories over a time interval $[0, T]$. We will consider only *likelihood*-based approaches in this paper. Hence, we need to compute the probability of spin trajectories (3) as a function of parameters, i.e., the so-called likelihood function in continuous time. Taking the limit $\Delta t \to 0$ in (3) and discarding prefactors which contain $\Delta t$ but are irrelevant for inference (being independent of $\boldsymbol{J}$), the complete-data likelihood function [16] is found to be

$$\mathcal{L}(\{\boldsymbol{s}\}_{0:T}|\boldsymbol{J}) = \prod_{(i,t)\in F} \frac{\exp[-s_i(t) H_i(t)]}{2\cosh[H_i(t)]} \times \prod_{i=1}^{N} \exp\left( \gamma \int_0^T \left\{ \frac{\exp[s_i(t) H_i(t)]}{2\cosh[H_i(t)]} - 1 \right\} dt \right). \tag{4}$$

A maximum likelihood estimate of the parameters $\boldsymbol{J}$ can be obtained by a (possibly penalised) gradient ascent approach of this function [15]. However, a Bayesian inference approach does not seem to be feasible from the expression (4). For a Bayesian approach one would introduce a prior density $p(\boldsymbol{J})$ of parameters and would infer statistical properties of $\boldsymbol{J}$ using the posterior density given by

$$p(\boldsymbol{J}|\{\boldsymbol{s}\}_{0:T}) = \frac{\mathcal{L}(\{\boldsymbol{s}\}_{0:T}|\boldsymbol{J}) p(\boldsymbol{J})}{\int \mathcal{L}(\{\boldsymbol{s}\}_{0:T}|\boldsymbol{J}) p(\boldsymbol{J})\, d\boldsymbol{J}}, \tag{5}$$

from which posterior expectations of parameters would have to be calculated by high-dimensional integrals. Due to the complex dependency of the likelihood on the parameters, the application of well-known techniques such as Monte Carlo sampling, e.g., using a Gibbs sampler, or approximate inference methods such as the variational approach [22] would not be trivial. We will show in the next section that the dependency of the likelihood on $\boldsymbol{J}$ can be remarkably simplified by augmenting the system by two sets of auxiliary random variables.

## IV. VARIABLE AUGMENTATION AND TRACTABLE LIKELIHOOD

The two main problems that prevent us from performing efficient analytical inference using Eq. (4) come from two sources: first, the time integral which contains the parameters $\boldsymbol{J}$ appears in an exponential function, and, second, the parameters also appear in the denominators in the hyperbolic cosine function. We will show that both problems can be solved by the introduction of auxiliary variables. We will start with a simplification of the integral.

### A. Poisson variables

We note that fields $H_i(t)$ are piecewise constant functions of time and do not change where no spin is flipped. Hence, the time integral can be calculated analytically. We will order the constant intervals and number them by $n \in \{0, 1, \ldots, n_{\max}\}$. We define $H_i^n$ and $s_i^n$ as the values of the field and spin $i$ between time points $t_n$ and $t_{n+1}$. $t_n$ denotes the time of the $n$th flip time for $n \in \{1, \ldots, n_{\max}\}$, while $t_0 = 0$ and $t_{n_{\max}+1} = T$. Hence, we obtain

$$\int_0^T \frac{\exp[s_i(t) H_i(t)]}{2\cosh[H_i(t)]} dt = \sum_{n=0}^{n_{\max}} \frac{\exp\left(s_i^n H_i^n\right)}{2\cosh\left(H_i^n\right)} (t_{n+1} - t_n). \tag{6}$$

Introducing a set of independent Poisson distributed random variables $\rho_i^n$ for each $i$ and each time slice between $t_{n+1}$ and $t_n$, we obtain the following representation of the second part of the likelihood:

$$\exp\left( \gamma \int_0^T \left\{ \frac{\exp[s_i(t) H_i(t)]}{2\cosh[H_i(t)]} - 1 \right\} dt \right) = \prod_{n=0}^{n_{\max}} \left\{ \sum_{\rho_i^n=0}^{\infty} \left[ \frac{\exp\left(s_i^n H_i^n\right)}{2\cosh\left(H_i^n\right)} \right]^{\rho_i^n} P_{\text{Po}}\left(\rho_i^n \middle| \gamma(t_{n+1} - t_n)\right) \right\}, \tag{7}$$

where

$$P_{\text{Po}}(\rho|\zeta) = e^{-\zeta} \frac{\zeta^\rho}{\rho!} \tag{8}$$

denotes a Poisson distribution with mean parameter $\zeta$. For Eq. (7) we made use of the equality

$$e^{\zeta(x-1)} = \sum_{\rho=0}^{\infty} x^{\rho} P_{\mathrm{Po}}(\rho|\zeta),$$

which is the moment-generating function of the Poisson distribution [23]. Similar variables were used in Ref. [17] to make Poisson-process likelihoods tractable for Monte Carlo sampling.

### B. Pólya-Gamma variables

To get rid of the hyperbolic terms in the denominators, we will use a remarkable representation which was discovered and used in the statistics literature in recent years to simplify Bayesian inference for logistic regression. Reference [18] found a convenient form of writing an inverse hyperbolic cosine as a continuous mixture of Gaussian densities as

$$\cosh^{-b}(x) = \int_0^{\infty} d\omega\, e^{-2\omega x^2} p_{\mathrm{PG}}(\omega|b,0), \tag{9}$$

where $p_{\mathrm{PG}}(\omega|b,0)$ is the *Pólya-Gamma density* with parameter $b$. Surprisingly, the exact form of this distribution is not of importance for our inference algorithm, but only the fact that one can derive its first moments straightforwardly (see Appendix B). Introducing Pólya-Gamma variables $\omega$ into the likelihood (7) yields the representation

$$p(\{\boldsymbol{s}\}_{0:T}|\boldsymbol{J}) = \sum_{\boldsymbol{\rho}} \int \mathcal{L}(\{\boldsymbol{s},\boldsymbol{\rho},\boldsymbol{\omega}\}_{0:T}|\boldsymbol{J})\, d\boldsymbol{\omega}, \tag{10}$$

with the augmented likelihood

$$\begin{aligned}
&\mathcal{L}(\{\boldsymbol{s},\boldsymbol{\rho},\boldsymbol{\omega}\}_{0:T}|\boldsymbol{J}) \\
&= \prod_{(i,t)\in F} \exp\{-s_i(t)H_i(t) - 2[H_i(t)]^2\omega_i(t)\} p_{\mathrm{PG}}(\omega_i(t)|1,0) \\
&\quad \times \prod_{i,n} \Big( \exp\left\{\rho_i^n \left[s_i^n H_i^n - \ln(2)\right] - 2\left(H_i^n\right)^2 \omega_i^n\right\} \\
&\quad \times P_{\mathrm{Po}}\left(\rho_i^n \middle| \gamma(t_{n+1}-t_n)\right) p_{\mathrm{PG}}\left(\omega_i^n \middle| \rho_i^n, 0\right)\Big).
\end{aligned} \tag{11}$$

The advantage of the augmented likelihood over the original one is the fact that the parameters appear at most quadratically in the exponential functions [note that the fields $H_i(t)$ are linear functions of the parameters]. As we will see, the computation of maximum likelihood and related estimators as well as Bayesian inference become considerably facilitated. We will postpone explicit results of Gibbs sampling algorithms to a future publication and discuss applications of the augmented likelihood to penalized maximum likelihood estimation and to a variational Bayes algorithm in this paper.

## V. INFERENCE

### A. EM algorithm

The EM algorithm [24] is a convenient way to maximize the likelihood iteratively with respect to $\boldsymbol{J}$ by using latent variable representations. The algorithm cycles between an E step and an M step and guarantees to increase the likelihood (4) in each step. At iteration $m+1$, in the *E step* one computes the cost function $Q(\boldsymbol{J},\boldsymbol{J}_m)$. It equals the expectation of the logarithm of the augmented likelihood with respect to the distribution of latent variables conditioned on the parameters at the previous iteration $m$:

$$\begin{aligned}
Q(\boldsymbol{J},\boldsymbol{J}_m) \doteq \sum_{\boldsymbol{\rho}} \int d\boldsymbol{\omega}\; p(\boldsymbol{\rho},\boldsymbol{\omega}|\{\boldsymbol{s}\}_{0:T},\boldsymbol{J}_m) \\
\times \ln \mathcal{L}(\{\boldsymbol{s},\boldsymbol{\rho},\boldsymbol{\omega}\}_{0:T}|\boldsymbol{J}).
\end{aligned} \tag{12}$$

For the *M step* we compute an update of the parameters via

$$\boldsymbol{J}_{m+1} = \arg\max_{\boldsymbol{J}} Q(\boldsymbol{J},\boldsymbol{J}_m). \tag{13}$$

The conditional distribution is given by

$$\begin{aligned}
&p(\{\boldsymbol{\rho},\boldsymbol{\omega}\}_{0:T}|\{\boldsymbol{s}\}_{0:T},\boldsymbol{J}) \\
&\quad = p(\{\boldsymbol{\omega}\}_{0:T}|\{\boldsymbol{s},\boldsymbol{\rho}\}_{0:T},\boldsymbol{J}) P(\{\boldsymbol{\rho}\}_{0:T}|\boldsymbol{J},\{\boldsymbol{s}\}_{0:T}),
\end{aligned} \tag{14}$$

where

$$\begin{aligned}
&p(\{\boldsymbol{\omega}\}_{0:T}|\{\boldsymbol{s},\boldsymbol{\rho}\}_{0:T},\boldsymbol{J}) \\
&\quad = \prod_{(i,t)\in F} p_{\mathrm{PG}}(\omega_i(t)|1,2H_i(t)) \prod_{n,i} p_{\mathrm{PG}}\left(\omega_i^n \middle| \rho_i^n, 2H_i^n\right),
\end{aligned} \tag{15}$$

where we defined the *tilted Pólya-Gamma* distribution as

$$p_{\mathrm{PG}}\left(\omega_i^n \middle| b,c\right) = \frac{\exp\left(-\frac{c^2}{2}\omega_i^n\right) p_{\mathrm{PG}}\left(\omega_i^n \middle| b,0\right)}{\cosh^{-b}\left(\frac{c}{2}\right)},$$

and where

$$P(\boldsymbol{\rho}|\boldsymbol{J},\{\boldsymbol{s}\}_{0:T}) = \prod_{n,i} P_{\mathrm{Po}}\left(\rho_i^n \middle| \gamma(t_{n+1}-t_n) \frac{\exp\left(s_i^n H_i^n\right)}{2\cosh\left(H_i^n\right)}\right). \tag{16}$$

The first part of the conditional density is over factorizing Pólya-Gamma variables and the second one over factorizing Poisson random variables. The necessary expectations for the E step follow from simple properties of Poisson random variables and of Pólya-Gamma random variables derived in Appendix B. This results in

$$\begin{aligned}
\langle\omega_i(t)\rangle &= \frac{1}{4H_i(t)} \tanh[H_i(t)], \\
\left\langle\omega_i^n\right\rangle &= \frac{\left\langle\rho_i^n\right\rangle}{4H_i^n} \tanh\left(H_i^n\right), \\
\left\langle\rho_i^n\right\rangle &= (t_{n+1}-t_n)\gamma \frac{\exp\left(s_i^n H_i^n\right)}{2\cosh\left(H_i^n\right)},
\end{aligned} \tag{17}$$

where the brackets $\langle\cdot\rangle$ denote expectations conditioned on $\boldsymbol{J}_m$. Since the augmented log-likelihood is a quadratic form in the parameters $\boldsymbol{J}$, the maximization leads to $N$ systems of linear equations for the vectors $\boldsymbol{J}_{i\cdot} \doteq (\theta_i, J_{i1}, \ldots, J_{iN})^{\top}$ of the form

$$A_i \boldsymbol{J}_{i\cdot} = \boldsymbol{b}_{i\cdot} \tag{18}$$

with

$$b_{ij} = -\sum_{t\in F(i)} s_i(t)s_j(t) + \sum_n \left\langle\rho_i^n\right\rangle s_i^n s_j^n \tag{19}$$

and

$$A_{ijk} = 4\left[\sum_{t\in F(i)} \left\langle\omega_i^t\right\rangle s_k(t)s_j(t) + \sum_n \left\langle\omega_i^n\right\rangle s_k^n s_j^n\right]. \tag{20}$$

Here $F(i)$ is the set of all times that spin $i$ has flipped. As mentioned before, only the first moment of the Pólya-Gamma density is required.

### B. Sparsity via L1 regularization

Assuming a factorizing Laplace distribution over each coupling $J_{ij}$,

$$p(J_{ij}) = \frac{\lambda}{2}\exp(-\lambda|J_{ij}|),$$

will enforce sparsity on the network. $\lambda$ is the scale parameter of this density. On the level of the MAP (maximum *a posteriori*) Bayesian estimator this is equivalent to L1 regularized maximum likelihood estimation. However, the absolute value in the exponent of this prior would prevent us from using the previously described EM procedure directly and allow only for gradient methods similar to Ref. [25]. Fortunately, this problem can again be solved by the introduction of a further auxiliary random variable for each single coupling parameter $J_{ij}$. This follows from the fact that a Laplace distribution can once more be represented as an infinite mixture of Gaussians [26,27],

$$\frac{\lambda}{2}\exp(-\lambda|J|) = \int d\beta \sqrt{\frac{\beta\lambda^2}{2\pi}}\exp\left(-\frac{\beta\lambda^2}{2}J^2\right)p(\beta), \quad (21)$$

with

$$p(\beta) = (\beta/2)^{-2}\exp\left[-1/(2\beta)\right].$$

By extending the augmented likelihood (11) to *sparsity variables* $\{\beta_{ij}\}$ a similar EM algorithm is possible to obtain the L1-regularized ML solution of $\boldsymbol{J}$. The required conditional density factorizes as

$$p(\{\boldsymbol{\rho},\boldsymbol{\omega}\}_{0:T},\boldsymbol{\beta}|\{\boldsymbol{s}\}_{0:T},\boldsymbol{J}) = p(\{\boldsymbol{\rho},\boldsymbol{\omega}\}_{0:T}|\{\boldsymbol{s}\}_{0:T},\boldsymbol{J})\; p(\boldsymbol{\beta}|\boldsymbol{J}), \quad (22)$$

where $p(\boldsymbol{\beta}|\boldsymbol{J}) = \prod_{i,j} p(\beta_{ij}|J_{ij})$ and each factor is a *generalized inverse Gaussian* distribution

$$p(\beta_{ij}|J_{ij}) = p_{\mathrm{GIG}}(\beta_{ij}|a_{ij},1,\nu) = \frac{a_{ij}^{\nu/2}}{2K_\nu(\sqrt{a_{ij}})}\beta_{ij}^{\nu-1}\exp\left(-\frac{a_{ij}\beta_{ij}+1/\beta_{ij}}{2}\right), \quad (23)$$

where $a_{ij} = \lambda^2 J_{ij}^2$, $\nu = -1/2$, and $K_\nu$ is the modified Bessel function of the second kind. The only change in the linear system (18) is in the matrices $A$, which have to be replaced by

$$A_{ijk}^{\mathrm{sparse}} = A_{ijk} + \delta_{i,k}\lambda^2\langle\beta_{ij}\rangle, \quad (24)$$

and where $\langle\beta_{ij}\rangle = (J_{ij}^2\lambda^2)^{-1/2}$ (see Appendix C).

### C. Approximate posterior distribution via variational Bayes

For Bayesian inference we assume the previously discussed Laplace prior over couplings $J_{ij}$ with scaling parameter $\lambda$ and for the external fields $\theta_i$ a Gaussian prior with mean $\mu_\theta$ and precision $\lambda_\theta^2$. To obtain a full posterior distribution including the couplings $\boldsymbol{J}$ we could either sample from the posterior or resort to a variational approach. The latter method is popular in the field of machine learning [22] but has its roots in statistical physics [28]. In our case we assume approximated posterior that has the following factorizing form:

$$p(\boldsymbol{J},\{\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{\beta}|\{\boldsymbol{s}\}_{0:T}) \approx q(\boldsymbol{J},\{\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{\beta}) \equiv q_1(\boldsymbol{J})q_2(\{\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{\beta}), \quad (25)$$

where the two factors $q_1$ and $q_2$ are optimized to minimize the relative entropy (Kullback-Leibler) divergence:

$$D(q;p) = \sum_{\boldsymbol{\rho}}\left[\int q(\boldsymbol{J},\{\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{\beta}) \times \ln\frac{q(\boldsymbol{J},\{\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{\beta})}{p(\boldsymbol{J},\{\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{\beta}|\{\boldsymbol{s}\}_{0:T})}\,d\boldsymbol{\omega}\,d\boldsymbol{\beta}\,d\boldsymbol{J}\right]. \quad (26)$$

This is equivalent to minimizing the *variational free energy*:

$$\mathcal{F}(q;p) = \sum_{\boldsymbol{\rho}}\left[\int q(\boldsymbol{J},\{\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{\beta}) \times \ln\frac{q(\boldsymbol{J},\{\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{\beta})}{p(\{\boldsymbol{s},\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{J},\boldsymbol{\beta})}\,d\boldsymbol{\omega}\,d\boldsymbol{\beta}\,d\boldsymbol{J}\right]. \quad (27)$$

The negative free energy is actually a lower bound on the log marginal likelihood

$$-\mathcal{F}(q;p) \leqslant \int \mathcal{L}(\{\boldsymbol{s}\}_{0:T}|\boldsymbol{J})p(\boldsymbol{J})\,d\boldsymbol{J}, \quad (28)$$

and can be used directly for approximate model selection [22], while in a pure maximum likelihood approach this is not possible.

Minimizing the variational free energy with respect to the factors of our factorizing distribution (25), the optimal factors turn out to be

$$q_1^\star(\boldsymbol{J}) \propto \exp\left(\langle\ln p(\boldsymbol{J},\{\boldsymbol{s},\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{\beta})\rangle_{q_2}\right),$$
$$q_2^\star(\{\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{\beta}) \propto \exp\left(\langle\ln p(\boldsymbol{J},\{\boldsymbol{s},\boldsymbol{\omega},\boldsymbol{\rho}\}_{0:T},\boldsymbol{\beta})\rangle_{q_1}\right),$$

which are obtained by iterative updates [22]. For the posterior at hand we find the optimal factor $q_2$ of the posterior

$$\begin{aligned} q_2^\star(\boldsymbol{\rho},\boldsymbol{\omega},\boldsymbol{\beta}) &= \prod_{(i,t)\in F} q_2(\omega_i(t))\prod_{i,n} q_2\left(\omega_i^n|\rho_i^n\right)q_2\left(\rho_i^n\right)\; q_2(\boldsymbol{\beta}) \\ &= \prod_{(i,t)\in F} p_{\mathrm{PG}}(\omega_i(t)|1,2\sqrt{\langle[H_i(t)]^2\rangle}) \\ &\quad\times \prod_{i,n} p_{\mathrm{PG}}\left(\omega_i^n|\rho_i^n,2\sqrt{\left\langle\left(H_i^n\right)^2\right\rangle}\right) \\ &\quad\times P_{\mathrm{Po}}\left\{\rho_i^n\middle|\gamma(t_{n+1}-t_n)\frac{\exp\left(s_i^n\langle H_i^n\rangle\right)}{2\cosh\left[\sqrt{\left\langle\left(H_i^n\right)^2\right\rangle}\right]}\right\} \\ &\quad\times \prod_{(ij)} p_{\mathrm{GIG}}\left(\beta_{ij}\middle|\langle J_{ij}^2\rangle\lambda^2,1,-1/2\right). \end{aligned} \quad (29)$$

From the fact that the augmented likelihood (11) and the sparsity prior factorize in the components $\boldsymbol{J}_{i\cdot}$ it follows that the optimal posterior $q_1^\star(\boldsymbol{J})$ does so as well. Each of those factors is a Gaussian distribution with covariance and mean given by

$$\Sigma_i = [4A_i + (\tilde{\Sigma}_i)^{-1}]^{-1}, \quad (30)$$

$$\boldsymbol{\mu}_i = \Sigma_i[b_i + (\tilde{\Sigma}_i)^{-1}\tilde{\boldsymbol{\mu}}_i], \quad (31)$$

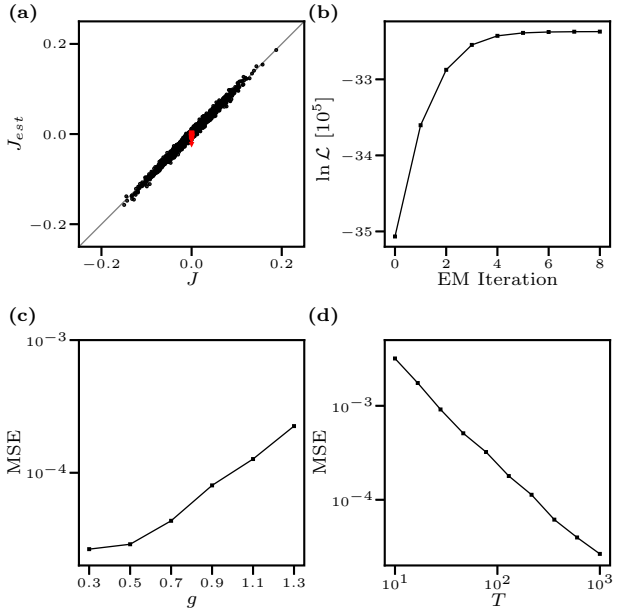



FIG. 1. Inference with EM algorithm on artificial data. (a) True couplings (black dots) and external fields (red triangles) vs inferred ones. (b) The log-likelihood as function of EM iterations. The parameters are set to $N = 40$, $T = 10^3$, and $g = 0.3$ with external fields $\boldsymbol{\theta} = \mathbf{0}$. (c) MSE between $\boldsymbol{J}$ and $\boldsymbol{J}_{\text{est}}$ as a function of scaling factor of the variance $g$ and (d) as a function of data length $T$. If not changed parameters are as in (a).

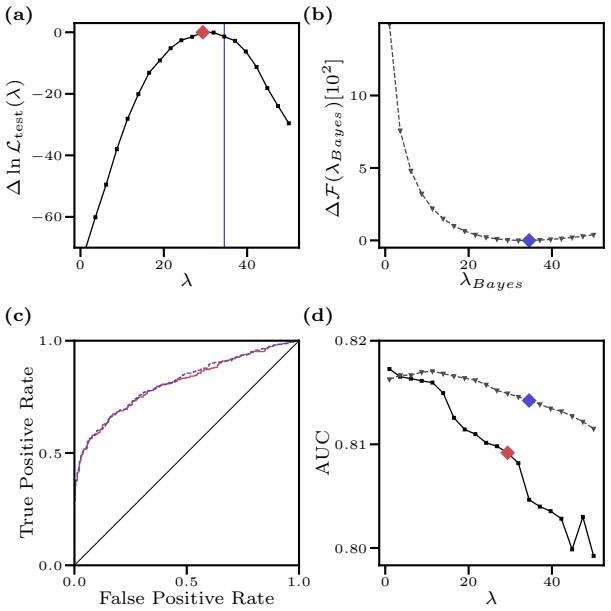



FIG. 2. Inference of sparse couplings with EM and variational Bayes. Artificial data ($T = 50, N = 25, g = 0.3$) are generated, but each coupling is set to 0 with probability $p_{\text{sparse}} = 1/2$. (a) Difference in likelihood (with respect to the likelihood obtained with optimal $\lambda^\star$) of couplings $\boldsymbol{J}_{\text{est}}$ inferred by EM as a function of regularization parameter $\lambda$. Likelihood $\mathcal{L}_{\text{test}}$ is computed on unseen test data ($T = 50$). The optimal parameter is $\lambda^\star = 29.4$ (red diamond). The vertical line marks the variational estimation $\lambda^\star_{\text{Bayes}}$. (b) Difference in free energy $\mathcal{F}$ (with respect to the likelihood obtained with estimate of optimal $\lambda^\star_{\text{Bayes}}$) of the variational Bayes algorithm. The optimal parameter is $\lambda^\star_{\text{Bayes}} = 34.5$ (blue diamond). (c) ROC curves for the $\lambda^\star$ (EM, solid red line) and $\lambda^\star_{\text{Bayes}}$ (Bayes, dashed blue line), respectively. (d) The AUC for different parameters $\lambda$ for the EM result (solid black line with squares) and the variational Bayes algorithm (dashed gray line with triangles). Diamonds mark the optimal $\lambda^\star$ and the estimate $\lambda^\star_{\text{Bayes}}$.

where $\tilde{\Sigma}_i$ is a diagonal matrix with $diag(\tilde{\Sigma}_i^{-1}) = (\lambda_\theta^2, \lambda^2\langle\beta_{i1}\rangle, \ldots, \lambda^2\langle\beta_{iN}\rangle)$. The prior mean is defined as $\tilde{\boldsymbol{\mu}}_i = (\mu_\theta, 0, \ldots, 0)^\top$. Similar to the EM algorithm, we have a variational step, where $q_2$ is optimized, given $q_1$ and a second one, optimizing $q_1$ given $q_2$. The variational step updating $q_2$ differs from E step in the sense that here expectations over the terms with the couplings $\boldsymbol{J}$ are required and not only the pointwise estimate (see Appendix D). The variational M step is similar to the EM algorithm, where the expectations for $A$ and $b$ are computed with respect to $q_2$.

The Python code of the algorithms discussed here is publicly available [29].

## VI. RESULTS

We test the EM algorithm on artificial data generated with random couplings $J_{ij}$ from a Gaussian distribution with mean 0 and variance $g^2/N$, where scaling factor $g = 0.3$. With external fields $\boldsymbol{\theta} = \mathbf{0}$ and update rate $\gamma = 100$ data is generated with a Gillespie algorithm [16] (see Appendix A).

### A. Maximum likelihood

In Fig. 1 the inference results for the EM algorithm are shown. Figures 1(a) and 1(b) present a single fit with $N = 40$ spins and data length $T = 10^3$. The inferred couplings $\boldsymbol{J}_{\text{est}}$ agree well with the true couplings $\boldsymbol{J}$. The logarithm of the likelihood (4) converges well after eight EM iterations. The mean squared error (MSE) increases with increasing scaling of the coupling variance $g$ [Fig. 1(c)] and decreases linearly on a log-log scale with increasing data length $T$ [Fig. 1(d)].

### B. L1 regularization and variational Bayes

Regularization becomes particular important once little data are at hand. To test this we generate couplings as before for a network of $N = 25$ spins, but a coupling is set to 0 with probability of $p_{\text{sparse}} = 0.5$. Generated data have length $T = 50$. We run the L1-regularized EM algorithm with different values of $\lambda$ and define the optimal $\lambda^\star$, whose MLE $\boldsymbol{J}_{\text{est}}$ maximizes the likelihood $\mathcal{L}_{\text{test}}$ on unseen test data ($T = 50$) generated by the true Ising parameters $\boldsymbol{J}$ [see Fig. 2(a)]. For inference by the variational algorithm on the same training data we estimate the optimal $\lambda^\star_{\text{Bayes}}$ by taking the value that minimizes the free energy (27) [Fig. 2(b)]. Note that the Bayesian algorithm requires no test data for this estimate.

Next we try to find the nonzero couplings from our fitting results. For the L1-penalized MLE an estimated coupling is considered as nonzero if $|J_{ij}^{\text{est}}| \geqslant z$, where the $z$ is an arbitrary threshold. To make use of the additional information of uncertainty, for the variational Bayes couplings are considered to be nonzero if $|\langle J_{ij}\rangle_{q_1}| \geqslant z\sqrt{(\Sigma_i)_{(jj)}}$. The classification of

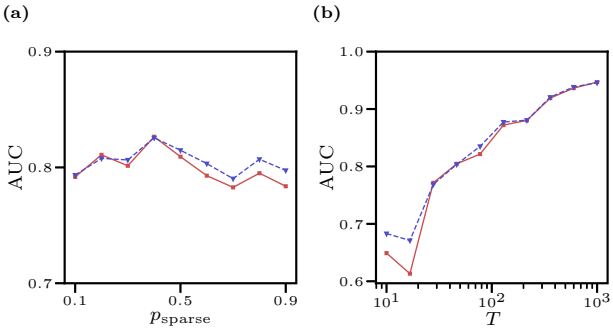



FIG. 3. The classification of nonzero couplings depending on sparsity of couplings and data length. (a) The AUC depending on the sparsity, i.e., the probability of a true coupling being 0, and in (b) depending on length of training data $T$. Results for EM shown by the red solid and the variational Bayes algorithm by the blue dashed line. If not changed, parameters are as in Fig. 2.

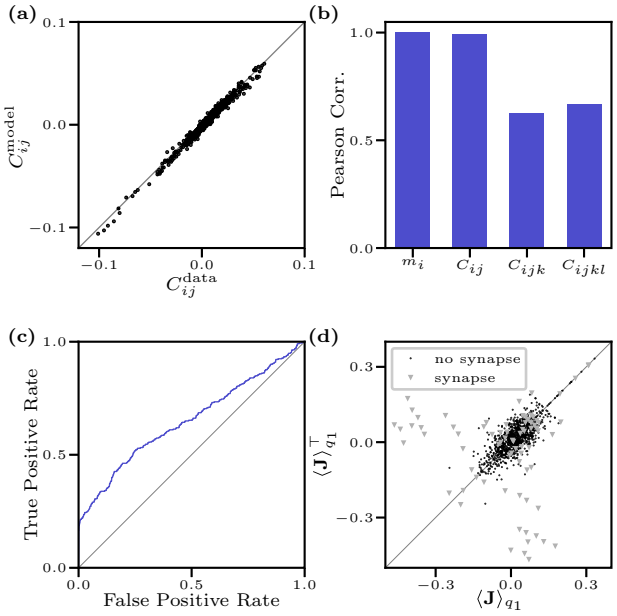



FIG. 4. Model fitted to data from a simulated recurrent network. (a) Second-order correlation $C_{ij}$ of the original data vs data sampled from mean couplings $\langle \boldsymbol{J} \rangle_{q_1}$ obtained via variational Bayes. (b) Pearson correlation between first- and fourth-order correlations of real sampled data. (c) ROC curve for identifying synapses with posterior over couplings $\boldsymbol{J}$ (AUC $= 0.65, \lambda^{\star}_{\text{Bayes}} = 16.5$). (d) Mean couplings $\langle \boldsymbol{J} \rangle_{q_1}$ of the variational posterior vs transpose. Couplings between neurons connected by a synapse are marked with gray triangles.

nonzero couplings is quantified by plotting the false positive rate (proportion of zero couplings that are misclassified as nonzero) versus the true positive rate (proportion of zero couplings that are correctly classified as nonzero) for a varying threshold $z \in [0, \infty]$. This is the Receiver-Operator characteristic (ROC) curve [see Fig. 2(c) for $\lambda^{\star}$ and $\lambda^{\star}_{\text{Bayes}}$ respectively]. As a measure of classification performance we use the area under the ROC curve (AUC), which is 1 for perfect classification and $1/2$ at chance level. Figure 2(d) shows that performance for the EM and the variational Bayes algorithm differ only marginally. For both algorithms the AUC is approximately constant, when repeating the same data generating and fitting procedure as before, but with varying sparsity $p_{\text{sparse}}$ [Fig. 3(a)]. When increasing the length of training data $T$ the AUC increases as expected [Fig. 3(b)]. For subsequent analysis we will focus on the variational Bayes algorithm.

### C. Inference of biophysical network

As an application of our algorithm we fit our model to data generated from a more biologically plausible network. We simulate a recurrent network of 1000 leaky integrate-and-fire neurons (800 excitatory and 200 inhibitory neurons) receiving Poisson input (see Appendix E and Ref. [30]). The synapses connect neurons randomly, are conductance-based, and vary in strengths and delays. The network is simulated for $T = 1000$ s. Spike times of 30 excitatory and 10 inhibitory neurons are used for fitting the kinetic Ising model, where neuron $i$ is considered as "active" for 10 ms ($= \gamma^{-1}$) after each spike $s_i(t) = 1$ and "inactive" otherwise [$s_i(t) = -1$]. The two questions we address here are (1) how well does the fitted model reproduce the statistics of the recorded data and (2) how are the synapses reflected in the estimated coupling parameters $\boldsymbol{J}$?

For the first question we compare data obtained from the spiking network with data sampled from the fitted the kinetic Ising model $\langle \boldsymbol{J} \rangle_{q_1}$ ($T = 1000$ s). To compare the original data with the Ising model data the (second-order) correlations from these data are computed as

$$C_{ij} = \frac{1}{T} \int_0^T [s_i(t) - m_i][s_j(t) - m_j]\, dt, \qquad (32)$$

where the mean is given by $m_i = \int_0^T s_i(t)\, dt/T$.

The results are compared in Fig. 4(a), and we find good agreement of original data and Ising samples. Furthermore, we compute the higher order correlations $C_{ijk}$ and $C_{ijkl}$ of the data and calculate the Pearson correlation coefficient between correlations from the original and the sampled data [see Fig. 4(b)]. The first two correlations ($m_i$ and $C_{ij}$) yield a Pearson correlation close to 1. Interestingly the Pearson correlation coefficient is strongly positive for $C_{ijk}$ and $C_{ijkl}$ as well, indicating that the Ising model also carries information about higher order correlations in the data.

As before we try to identify synapses in the simulated network by ROC curve analysis [Fig. 4(c)]. The classification yields an AUC $= 0.65$ ($\lambda^{\star}_{\text{Bayes}} = 16.5$). Even though there is information about the synapses, many more nonzero couplings are estimated that do not directly reflect synapses in the network. This is possibly caused by the fact that the network is only partially observed and the kinetic Ising model compensates for this part with more nonzero couplings.

Previous work has indicated for the kinetic Ising model in discrete time [15,31], that for experimental data recorded *in vivo* the estimated couplings $\boldsymbol{J}$ show a symmetric signature: $J_{ij} \approx J_{ji}$. This is particularly interesting for the Ising model in

continuous time, since for the model with symmetric couplings the stationary distribution is given by the maximum entropy equilibrium model [14] and potentially justifies the use of static Ising models for such data. As an indicator for symmetry we plot the mean of the variational posterior obtained from the recurrent network versus its transpose [Fig. 4(d)]. We observe that many couplings are indeed close to the diagonal, while some show large deviations from it. However, those with strong deviations correspond to the couplings which reflect synapses in the underlying network. Hence, the approximately symmetric part is not caused by synapses, but either by our data transformation to fit the Ising model or by the fact that we only partially observe the network.

## VII. DISCUSSION

In this paper we have presented efficient algorithms for inferring the couplings of a continuous time kinetic Ising model defined by Glauber dynamics. Using a combination of two auxiliary latent variable sets the complete data log-likelihood becomes a simple quadratic function in the couplings. A third set of auxiliary variables allows us to deal with sparse couplings, equivalent to an L1-penalized likelihood without resorting to gradient-based algorithms [25]. Using this representation we derive an EM algorithm for (penalized) maximum likelihood estimation of the couplings with explicit analytical updates. This leads to a guaranteed increase of the likelihood in each iteration. The computational complexity is similar to a Newton-Raphson method for optimizing the original log-likelihood, since the Hessian matrix requires a similar inverse of the summed data covariances [15]. However, our algorithm does not require any tuning of a step size.

We have extended our latent variable approach to a Bayesian scenario but have restricted ourselves to a fast variational Bayes approximation. However, it is straightforward to develop a Monte Carlo Gibbs sampler for the latent variable structure. This would require drawing samples from Pólya-Gamma density rather than computing only its mean. We have tested our inference algorithms on simulated data demonstrating fast convergence of the method. The variational Bayes approximation allows us to perform model selection, yielding hyperparameters which achieve close to optimal likelihoods on test data. As an application of our approach we have investigated the quality of the kinetic Ising model to describe data which were generated from a more realistic, biologically inspired *integrate and fire* neural network model which is only partially observed. We have shown that the kinetic Ising model reproduces low order statistics of the data well. However, the partial observation of neurons prohibits a safe identification of synapses in terms of the Ising coupling parameters. It would be interesting to see if the performance of a kinetic Ising model on such data could be improved by including explicit unobserved neurons and their couplings in the model [32]. We expect that our latent variable approach would facilitate statistical inference for such an extended model and provide alternatives to current approximate inference methods [33–36]. We are currently working on an extension of our inference approach by including time-dependent model parameters which makes the model more realistic and which has been shown of importance for biological data analysis [12,37].

Finally, our latent variable approach should also be applicable to other inference problems for point process models; e.g., a combination with Gaussian process priors should allow for nonparametric approximate inference of rate functions for inhomogeneous Poisson processes [17]. Models with similar point-process likelihoods are common in neuroscience [38–40], for modeling seismic activity [41], analyzing social network analysis [42], etc.

## ACKNOWLEDGMENT

C.D. was supported by the Deutsche Forschungsgemeinschaft (GRK1589/2).

## APPENDIX A: GENERATING DATA

To generate artificial data for the kinetic Ising model in continuous time we can make use of the Gillespie algorithm [16]. Having a coupling matrix $\boldsymbol{J}$ for $N$ spins and an initial data vector $\boldsymbol{s}(0)$ data are generated as follows: (1) We draw the next update time $t'$ from a exponential distribution with mean $(\gamma \times N)^{-1}$, (2) we draw a spin $i$ with probability $1/N$, and finally (3) we flip spin $i$ at time $t'$ according to Eq. (2) and set $t \leftarrow t'$. These three steps are repeated until $t \geqslant T$.

## APPENDIX B: PROPERTIES OF PÓLYA-GAMMA DISTRIBUTION

The Pólya-Gamma density [18] allows us to represent the inverse hyperbolic cosine function as an infinite Gaussian mixture

$$\cosh^{-b}(c/2) = \int_0^\infty d\omega \exp\left(-\frac{c^2}{2}\omega\right) p_{\mathrm{PG}}(\omega|b,0). \tag{B1}$$

Furthermore, we define the *tilted Pólya-Gamma distribution* as

$$p_{\mathrm{PG}}(\omega|b,c) \propto e^{-c^2/2\omega} p_{\mathrm{PG}}(\omega|b,0). \tag{B2}$$

From Eqs. (B1) and (B2) we obtain the moment generating function

$$\langle e^{\omega t}\rangle = \frac{\cosh^b(c/2)}{\cosh^b\left(\sqrt{\frac{c^2/2-t}{2}}\right)}. \tag{B3}$$

By differentiating (B3) at $t = 0$ the analytical form of the expectation of $\omega$ is obtained:

$$\langle\omega\rangle = \frac{b}{2c}\tanh\left(\frac{c}{2}\right). \tag{B4}$$

## APPENDIX C: LATENT VARIABLE REPRESENTATION OF LAPLACE DISTRIBUTION

The Laplace distribution can written as an infinite mixture of Gaussians [26,27]

$$\frac{\lambda}{2}\exp(-\lambda|x|) = \int_0^\infty \sqrt{\frac{\beta\lambda^2}{2\pi}}\exp\left(-\frac{\beta\lambda^2}{2}x^2\right)p(\beta)d\beta, \tag{C1}$$

with

$$p(\beta) = (\beta/2)^{-2}\exp\left(-\frac{1}{2\beta}\right). \tag{C2}$$

By inspection we find the conditional density

$$p(\beta|x) = p_{\mathrm{GIG}}(\beta|x^2\lambda^2, 1, -1/2), \tag{C3}$$

where $p_{\mathrm{GIG}}$ is a generalized inverse Gaussian distribution defined as

$$p_{\mathrm{GIG}}(\beta|a,b,\nu) = \frac{(a/b)^{\nu/2}}{2K_\nu(\sqrt{ab})}\beta^{\nu-1}\exp\left[-(a\beta - b/\beta)/2\right], \tag{C4}$$

and $K_\nu$ is the modified Bessel function of the second kind. The expectations of $\beta$ are

$$\langle\beta\rangle = \frac{K_{1/2}(\sqrt{x^2\lambda^2})}{\sqrt{x^2\lambda^2}K_{-1/2}(\sqrt{x^2\lambda^2})} = \frac{1}{\sqrt{x^2\lambda^2}}, \tag{C5}$$

where the Bessel functions cancel due to $K_\nu(\sqrt{x^2\lambda^2}) = K_{-\nu}(\sqrt{x^2\lambda^2})$.

## APPENDIX D: VARIATIONAL BAYES

In the variational Bayes algorithm the updates in the step updating $q_2$ involve the expectations $\langle H_i^{t,n}\rangle_{q_1}$ and $\langle (H_i^{t,n})^2\rangle_{q_1}$ instead of only the pointwise MLE in the E step of the EM algorithm. The required expectations are

$$\begin{aligned}
\langle\omega_i(t)\rangle &= \frac{1}{4\sqrt{\langle(H_i(t))^2\rangle}}\tanh[\sqrt{\langle[H_i(t)]^2\rangle}],\\
\langle\omega_i^n\rangle &= \frac{\langle\rho_i^n\rangle}{4\sqrt{\langle\left(H_i^n\right)^2\rangle}}\tanh\left[\sqrt{\langle\left(H_i^n\right)^2\rangle}\right],\\
\langle\rho_i^n\rangle &= (t_{n+1}-t_n)\gamma\frac{\exp\left(s_i^n\langle H_i^n\rangle\right)}{2\cosh\left[\sqrt{\langle\left(H_i^n\right)^2\rangle}\right]}.
\end{aligned} \tag{D1}$$

The free energy (27) that is minimized in the variational Bayes algorithm is easy to calculate since we immediately see that the terms involving $p_{\mathrm{PG}}(\omega_i(t)|1,0)$, $p_{\mathrm{PG}}(\omega_i^n)|\rho_i^n,0)$, $P_{\mathrm{Po}}(\rho_i^n|\gamma(t_{n+1}-t_n))$ and $p(\beta)$ appear in the nominator as well as in the denominator and cancel out. The free energy at a minimum is

$$\begin{aligned}
\mathcal{F}(q^\star;p) = &\sum_{(i,t)\in F}\ln\frac{2\cosh[\sqrt{\langle[H_i(t)]^2\rangle}]}{\exp[-s_i(t)\langle H_i(t)\rangle]}\\
&+\sum_{i,n}\gamma(t_{n+1}-t_n)\left\{1-\frac{\exp\left(s_i^n\langle H_i^n\rangle\right)}{2\cosh\left[\sqrt{\langle\left(H_i^n\right)^2\rangle}\right]}\right\}\\
&+\sum_{i,j}\ln\left[\frac{\sqrt{2\pi}\langle J_{ij}^2\rangle^{-1/4}}{(2\sqrt{\lambda})^3K_{-1/2}\left(\sqrt{\lambda^2\langle J_{ij}^2\rangle}\right)}\right]\\
&-\sum_i\left\langle\ln\mathcal{N}\left(\theta_i\middle|\mu_\theta,\lambda_\theta^{-2}\right)\right\rangle+\langle\ln q_1(\boldsymbol{J})\rangle,
\end{aligned} \tag{D2}$$

where all expectations are taken over the variational posterior $q^\star$. Note the similarity of the first two summands and the likelihood (4).

## APPENDIX E: SIMULATED NETWORK OF SPIKING NEURONS

We simulate a spiking network similar to the one described in Ref. [30], Fig. 3. The network consisted of three recurrently connected populations of neurons: 800 input ($X$) neurons, 800 excitatory ($E$), and 200 inhibitory ($I$) neurons. The input neurons do not get any input and generate Poisson spikes independently with a rate of 10 Hz. For the conductance-based integrate-and-fire neuron $i$ in the population $\alpha\in\{E,I\}$ the dynamics of the membrane potential $V_i^\alpha$ are described by the differential equation

$$C_m\frac{dV_i^\alpha}{dt} = -g_L\left(V_i^\alpha - V_L\right) + \sum_{\beta\in\{X,E,I\}} I_i^{\alpha,\beta}(t), \text{ if } V_i^\alpha < V_{th}, \tag{E1}$$

where the membrane capacitance is set to $C_m = 0.25$ nF and the leak conductance $g_L = 16.7$ nS. The resting potential is $V_L = -70$ mV, and the firing threshold $V_{th} = -50$ mV. After each spike the membrane potential was reset to $V_R = -60$ mV. $E$ and $I$ neurons have a 2 and 1 ms refractory period, respectively. $I_i^{\alpha\beta}$ is the input current neuron $i$ receives from population $\beta$.

The neurons are connected with probability $p_{\mathrm{connect}} = 0.2$, and the connections consist of conductance-based synapses (for details see the Supplementary Material of Ref. [30]). We draw the conductances for the synapses from a uniform distribution with mean $g^{\alpha\beta}$ and standard deviation $0.5g^{\alpha\beta}$. As in Ref. [30] we set $g^{EE} = 2.4$ nS, $g^{EI} = 40$ nS, $g^{IE} = 4.8$ nS, $g^{II} = 40$ nS, and $g^{EX} = g^{IX} = 5.4$ nS.

For generating data we simulated the network for $T = 1000$ s and recorded the spike times of a randomly selected subpopulation (100 excitatory and 40 inhibitory neurons). From those, the 30 excitatory and 10 inhibitory neurons with the highest firing rates are selected as data for fitting the kinetic Ising model.

To preprocess the data for the Ising model we follow the argument of Ref. [15]. The update rate $\gamma$ can be interpreted as the inverse of the width of a neuron's autocorrelation function, which is typically found to be 10 ms. Hence we set $\gamma = 10^2$ Hz and consider a neuron as "active" for 10 ms after each spike.

[1] H. C. Nguyen, R. Zecchina, and J. Berg, Adv. Phys. **66**, 197 (2017).
[2] E. Schneidman, M. J. Berry II, R. Segev, and W. Bialek, Nature (London) **440**, 1007 (2006).
[3] Y. Roudi, J. Tyrcha, and J. Hertz, Phys. Rev. E **79**, 051915 (2009).
[4] M. Weigt, R. A. White, H. Szurmant, J. A. Hoch, and T. Hwa, Proc. Nat. Acad. Sci. USA **106**, 67 (2009).

[5] T. R. Lezon, J. R. Banavar, M. Cieplak, A. Maritan, and N. V. Fedoroff, Proc. Nat. Acad. Sci. USA **103**, 19033 (2006).
[6] L. Bachschmid-Romano and M. Opper, J. Stat. Mech.: Theory Exp. (2017) 063406.
[7] M. Vuffray, S. Misra, A. Lokhov, and M. Chertkov, in *Advances in Neural Information Processing Systems*, edited by D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (Curran Associates Inc., 2016), pp. 2595–2603.
[8] Y. Roudi and J. Hertz, J. Stat. Mech.: Theory Exp. (2011) P03031.
[9] M. Mézard and J. Sakellariou, J. Stat. Mech.: Theory Exp. (2011) L07001.
[10] Y. Roudi and J. Hertz, Phys. Rev. Lett. **106**, 048702 (2011).
[11] H.-L. Zeng, E. Aurell, M. Alava, and H. Mahmoudi, Phys. Rev. E **83**, 041135 (2011).
[12] J. Tyrcha, Y. Roudi, M. Marsili, and J. Hertz, J. Stat. Mech.: Theory Exp. (2013) P03005.
[13] D. Soudry, S. Keshri, P. Stinson, M.-h. Oh, G. Iyengar, and L. Paninski, arXiv:1309.3724 (2013).
[14] R. J. Glauber, J. Math. Phys. **4**, 294 (1963).
[15] H.-L. Zeng, M. Alava, E. Aurell, J. Hertz, and Y. Roudi, Phys. Rev. Lett. **110**, 210601 (2013).
[16] D. Wilkinson, *Stochastic Modelling for Systems Biology*, Chapman & Hall/CRC Mathematical & Computational Biology (Taylor & Francis, Philadelphia, 2006).
[17] R. P. Adams, I. Murray, and D. J. MacKay, in *Proceedings of the 26th Annual International Conference on Machine Learning* (ACM, Montreal, Quebec, Canada, 2009), pp. 9–16.
[18] N. G. Polson, J. G. Scott, and J. Windle, J. Am. Stat. Assoc. **108**, 1339 (2013).
[19] S. Linderman, M. Johnson, and R. P. Adams, in *Advances in Neural Information Processing Systems*, edited by C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (Curran Associates Inc., 2015), pp. 3456–3464.
[20] J. G. Scott, R. C. Kelly, M. A. Smith, P. Zhou, and R. E. Kass, J. Am. Stat. Assoc. **110**, 459 (2015).
[21] J. G. Scott and L. Sun, arXiv:1306.0040 (2013).
[22] C. M. Bishop, *Pattern Recognition and Machine Learning* (Springer, New York, 2006).
[23] J. Kingman, *Poisson Processes*, Oxford Studies in Probability (Clarendon Press, Oxford, 1992).
[24] A. P. Dempster, N. M. Laird, and D. B. Rubin, J. R. Stat. Soc. B **39**, 1 (1977).
[25] H. L. Zeng, J. Hertz, and Y. Roudi, Phys. Scr. **89**, 105002 (2014).
[26] F. Girosi, *Models of Noise and Robust Estimation* (Massachusetts Institute of Technology, Cambridge, MA, 1991).
[27] M. Pontil, S. Mukherjee, and F. Girosi, in *International Conference on Algorithmic Learning Theory*, edited by H. Arimura, S. Jain, and A. Sharma (Springer, New York, 2000), pp. 316–324.
[28] R. Feynman, *Statistical Mechanics: A Set of Lectures*, Advanced Books Classics (Avalon Publishing, 1998).
[29] Python code: https://github.com/christiando/dynamic_ising.git.
[30] A. Renart, J. De La Rocha, P. Bartho, L. Hollender, N. Parga, A. Reyes, and K. D. Harris, Science **327**, 587 (2010).
[31] J. Hertz, Y. Roudi, and J. Tyrcha, arXiv:1106.1752 (2011).
[32] Y. Roudi and G. Taylor, Curr. Opin. Neurobiol. **35**, 110 (2015).
[33] J. Tyrcha and J. Hertz, Mathematical Biosciences and Engineering: MBE **11**, 149 (2014).
[34] B. Dunn and Y. Roudi, Phys. Rev. E **87**, 022127 (2013).
[35] L. Bachschmid-Romano and M. Opper, J. Stat. Mech.: Theory Exp. (2014) P06013.
[36] C. Battistin, J. Hertz, J. Tyrcha, and Y. Roudi, J. Stat. Mech.: Theory Exp. (2015) P05021.
[37] C. Donner, K. Obermayer, and H. Shimazaki, PLoS Comput. Biol. **13**, e1005309 (2017).
[38] D. R. Brillinger, Biol. Cybern. **59**, 189 (1988).
[39] L. Paninski, Netw., Comput. Neural Syst. **15**, 243 (2004).
[40] K. W. Latimer, E. Chichilnisky, F. Rieke, and J. W. Pillow, in *Advances in Neural Information Processing Systems*, edited by Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberge (2014), pp. 954–962.
[41] Y. Ogata, Ann. Inst. Stat. Math. **50**, 379 (1998).
[42] Q. Zhao, M. A. Erdogdu, H. Y. He, A. Rajaraman, and J. Leskovec, in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (ACM, Sydney, NSW, Australia, 2015), pp. 1513–1522.

# Chapter 6

# Conjugacy by augmentation: Additional models & potential extensions

In the previous chapters we have demonstrated that the augmentation scheme described in chapter 2 is applicable to 3 different models, and that the resulting inference algorithms are much faster than previously proposed ones.

As already mentioned the augmentation scheme derived in chapter 2 is equivalent to combining the Pólya–Gamma (Polson et al. 2013) and Poisson process augmentation (Adams et al. 2009). Independently of the presented work the same augmentation scheme has been utilised by Lindon (2018), who implemented an efficient Gibbs sampler for one–dimensional problems of the sigmoidal Gaussian Cox process addressed in chapter 3. Another recent work (Gonçalves and Gamerman 2018) chose a scaled Gaussian cumulative density function as link function, which allows for Gibbs sampling only using the Poisson process augmentation discussed in chapter 2. However, sampling from the conditional posterior of the GP is not straightforward.

We would like to emphasise, that the models discussed in chapters 3–5 are only an exemplary subset of possible applications. For example, a natural extension of Poisson processes are self–exciting point processes, also known as Hawkes' processes (Hawkes 1971). Preceding events of such a process increase the likelihood of following events and are of interest in modelling of financial markets (Embrechts et al. 2011), seismic activity (Ogata 1998) and neuronal data (Linderman and Adams 2015). So called *linear* Hawkes' processes have the likelihood as in Eq (2), chapter 2, with an intensity function

$$\Lambda_Z(t) = \lambda_0 + \sum_{t' \in \mathcal{H}_t} f(t')\phi(t - t'),$$

where $\lambda_0 \in \mathbb{R}^+$ is the baseline intensity, and $\mathcal{H}_t$ is the set of past events at time $t$. $\phi$ is a non–negative memory kernel and $f(t) : [0, T) \to \mathbb{R}^+$ the time–dependent excitation amplitude. By choosing $f(t) = c\sigma(g(t))$ being a scaled sigmoid and $g(t)$ a GP, we can again utilise the augmentations developed previously. For *non–linear* Hawkes processes the intensity is given by

$$\Lambda_Z(t) = f\left(\lambda_0 + \sum_{t' \in \mathcal{H}_t} g(t')\phi(t - t')\right),$$

where $\lambda_0$ and $g(t)$ are not required to be positive any more, as long as $f : \mathbb{R} \to \mathbb{R}^+$. Surprisingly, when choosing $f(\cdot) = c\sigma(\cdot)$ and $g$ being a GP, the inference problem is even simpler than for the

linear case and is closely related to the inference of the kinetic Ising model discussed in chapter 5.

Additionally, for a specific likelihood for multi–class GP classification we achieve conjugacy of the model with the 3 augmentation schemata used in chapter 4. For details see appendix A, part III. Furthermore, heteroscedastic GP regression problems (Lázaro-Gredilla and Titsias 2011) can be addressed, where the variance is space–dependent and its inverse is modelled by the scaled sigmoid having a GP as argument. We expect that the methods developed in this work can be used to perform inference on many additional models.

Admittedly, introducing the sigmoid link function in these models can be regarded as artificial in order to utilise the presented augmentations. As discussed in chapter 3 other link functions, e.g. exponential or squared, do not require (neither allow for) a similar augmentation scheme. However, they come with other disadvantages. For the variational approach with the exponential link function the approximate posterior's variance is uncoupled from the data (Lloyd et al. 2014). Models with the squared link function are limited to certain priors and domains. Furthermore, we experienced that inference results for models with the squared link function depend on how the domain $\mathcal{X}$ is scaled. For stability of the algorithm from Lloyd et al. (2014) this scaling had to be adjusted. In contrast, we did not experience such effects for the scaled sigmoid link function.

The Poisson process augmentation in Eq (7), chapter 2, can be invoked for all bounded link functions. However, the property of the sigmoid function $\sigma(z) = 1 - \sigma(-z)$ made the subsequent Pólya-Gamma representation possible. In general, even if one has a Gaussian representation of the link function one requires in addition such a representation for the 'complement' function, whereas for the sigmoid function we can use the same representation.

Interestingly, a direct variational lower bound has been derived for the sigmoid link function without making use of Pólya–Gamma augmentation (Jaakkola and Jordan 2000; Mackay and Gibbs 2000). This bound is equivalent to the variational one obtained with the Pólya–Gamma augmentation (Wenzel et al. 2018). Palmer et al. (2006) discusses a class of functions for which similar variational lower bounds can be derived. This function class is broad and interesting, because the derivation does not require knowledge (or even existence) of the augmentation densities, e.g. the Pólya–Gamma density. With the hindsight of the present work we can derive a lower bound for the likelihood in Eq (2), chapter 2 without making use of the marked Poisson process augmentation (see appendix B, part III). However, also here one requires $\sigma(z) = 1 - \sigma(-z)$ and we would not be able to derive the variational bound without this property.

The sampling scheme that was utilised in chapter 4 yields fast converging Markov chains, where the samples quickly become uncorrelated. For such schemes knowing the augmentation density is essential. The Schönberg theorem (Ressel 1976) shows, that translation and rotation invariant kernels can be written as scale Gaussian mixture models and that the augmentation density exists. Finding link functions, that are composed of these kernels (Mercer 1909) and have the property $f(x) = 1 - f(-x)$ would allow for an augmentation scheme described in chapter 2. Sampling the required densities, however, is still an issue.

We demonstrated superior efficiency of the newly derived variational mean–field algorithms compared to more traditional methods, that assume an approximate Gaussian posterior and optimise the parameters by gradient methods (Hensman et al. 2015b). Because the latter approach directly maximises the lower bound on the model evidence $p(\mathcal{D})$, it finds the (locally) optimal variational Gaussian posterior. As we have seen empirically in chapter 3, the variational posterior obtained with the augmented model and the original model perform similar in terms of test likelihood, indicating that the gap between the two posteriors is relatively small. Potential improvements of the augmented variational posterior could be obtained by pertubative corrections (Opper et al. 2015) as was done by Wenzel et al. (2018). Alternatively, one could use the here proposed variational mean–field methods to find a posterior that is close to an optimum, and then use this posterior to initialise gradient methods (Hensman et al. 2015b) in order to find the optimal variational Gaussian posterior.

As we have shown in the previous chapters, the variational algorithm only requires $\sim 10$ iterations to converge. Despite empirical evidence we know little of why it converges quickly. For each update of the Gaussian model parameters we need to solve a quadratic problem, which is convex. Previous work addressed variational inference for non–conjugate models as well by either reformulating the problem into locally convex subproblems (Khan et al. 2013), or making use of 'partial' conjugacy of models (Khan and Lin 2017). Similar to the results presented here the optimisation procedures converge relatively fast and in just a few iterative steps. However, using these methods for point–process models without the augmentations presented here is still challenging, because the double integral, i.e. the expectation over a random process and the integration over the observed domain, has to be computed.

# Part II

# Inference of models for non-stationary spiking data

# Chapter 7

# Statistical modelling of spiking data: A brief introduction

The neurosciences are no exception to the nowadays trend of collecting increasingly larger and complex datasets. While this is true for neuroscience at all levels (molecular, single neuron, network and behavioural level), the availability of massive recordings of neuronal activity was especially favoured by the parallel advance in electrophysiology and optogenetic techniques (Ahrens et al. 2013; Einevoll et al. 2012). Such data are recorded in-vivo under increasingly complex experimental paradigms, to relate cellular activity to the organism's behaviour (Ishiyama and Brecht 2016; Stensola et al. 2015). Even though these data are recorded in tightly controlled experimental settings, the animal is affected by multiple factors (e.g. mood, attentional state, internal dynamics etc.) which are commonly unknown to the researchers and thus hard to control. This poses a challenge for the analysis of these data, because not accounting for those phenomena might result in wrong conclusions. Hence, semi– or unsupervised inference methods for data analysis are required, which are capable to extract these unknowns from the recorded data.

Attempts to model spiking activity dates back to Hodgkin and Huxley (1952), describing a single neuron dynamics by a set of ordinary differential equations. Since then a range of models have been proposed, which vary in their complexity and level of abstraction (Abbott and Kepler 1990). For statistical description of spike train data, likelihood–based approaches have been suggested, i.e. models for which a data likelihood can be derived (Cunningham and Yu 2014; Pillow et al. 2008; Schneidman et al. 2006). Even though (or because) undoubtedly oversimplifying the picture, these models have attracted great interest, since they allow projecting recorded data in low dimensional parameter spaces.

Despite the fact that models of likelihood based approaches can be conveniently fitted to data, limitations and weaknesses of these models need to be taken into account. Modelling always requires assumptions about the underlying system. If those are not chosen carefully, outcomes might be misleading. For example not accounting for non–stationarity can lead to spurious positive results in correlation analysis (Brody 1999). Furthermore, because of their phenomenological nature, models can describe statistical properties of observed data but often do not provide a satisfying explanation for these. An example are higher order correlations observed in spike train data and quantified by such phenomenological models (Ohiorhenuan et al. 2010). It was shown that simple statistical models can reproduce such higher order correlations (Macke et al. 2011; Shimazaki et al. 2015); These models, however, do not provide any mechanistic explanation of the phenomenon. Recently, Shomali et al. (2018) used a minimal mechanistic integrate–and–fire (I&F) model to investigate the underlying mechanisms of the experimental observations.

Considering models, that account for specific properties of the data might prevent misinterpretation

and provide straightforward experimental hypotheses. However, this usually comes with increasing model complexity, which in turn requires more complicated fitting procedures and more data. Choosing the 'best' model for recorded spike train data usually results in finding a compromise between those two extremes.

While analysing in–vivo data that are recorded from behaving animals non–stationarity has to be accounted for. In fact, it has been shown that for resting animals (Tsodyks et al. 1999) and even for neurons in cell–cultures (Sasaki et al. 2007) the stationary assumption can be violated. In the following, we address two different models which account for temporal variability of spike–train data recorded from multiple neurons. These models are based on two different levels of abstraction, where one is more conventional and the other one tries to minimise the gap between model and the biological system.

## Outline of part II

First, in chapter 8 we consider a non–stationary extension of the model presented in chapter 5. The model assumes that observed spike–train data are generated by a kinetic Ising model, a simple dynamics for binary data considering couplings among neurons. To account for non–stationarity, we make the simplifying assumption that the model parametrisation (i.e. effective couplings) intermittently jumps into different states. For the time between jumps, the parameters are constant. The number of model's states are finite. Previously described augmentation schemes combined with variational inference (Donner and Opper 2017) allow for fitting this fully Bayesian model efficiently to spike data. Furthermore, we show that this method does not require explicit binning of data, an assumption which is required for the majority of likelihood–based models and might affect the interpretation of results.

For the second model, presented in chapter 9, we assume a minimal biophysically plausible spiking mechanism, namely the integrate-and-fire (I&F) class (Gerstner and Kistler 2002). Interesting from the inference perspective is that for this class of physiologically constrained models it is possible to derive a likelihood semi analytically, by solving the first passage time problem (Ladenbauer et al. 2018; Mullowney and Iyengar 2008). These likelihoods have been used relatively little so far because optimisation is thought of as being demanding compared to more phenomenological models. However, Ladenbauer et al. (2018) showed that inference for these models is possible in the stationary case. In this work, we consider a non–stationary scenario, where a population of I&F neurons is driven by a time–dependent stochastic input current. We show, that for this model a Hidden Markov model can be derived, for which the likelihood can be efficiently evaluated and optimised.

# Chapter 8

# Unpublished article: *Bayesian network inference from non-stationary spiking data*

Authors:
Christian Donner[1,2], Manfred Opper[1,2]
[1]Technische Universität Berlin. [2]Bernstein Center for Computational Neuroscience Berlin.

**Chapter 8** This chapter comprises the unpublished manuscript, which is authored by myself (CD), and Prof. Manfred Opper (MO).

Contributions:
CD and MO conceived and designed the work. CD derived the inference algorithms and developed the Python code. CD performed the numerical experiments. CD wrote the manuscript.

Python code on GitHub: https://github.com/christiando/MJP_ising_inference.git

# Bayesian network inference from non-stationary spiking data

**Christian Donner** christian.donner@bccn-berlin.de
**Manfred Opper** manfred.opper@tu-berlin.de
*Artificial Intelligence Group*
*Technische Universität Berlin*
*Berlin, Germany*

## Abstract

We propose a model for non–stationary, correlated spiking data recorded from multiple neurons in continuous time. The correlated activity is modelled by a kinetic Ising model, which considers effective couplings between the observed neurons. To account for the non–stationarity in spiking data, the parametrisation of the Ising model is assumed to follow a Markov jump process, which can assume a finite number of states. We derive an efficient variational inference scheme for this model using a structured mean–field assumption. The resulting unsupervised algorithm accurately recovers the Markov jump process, the dynamic coupling structure of the network, and the probabilities of the Markov jump process states. Finally, we demonstrate practicality on multi-unit recordings from monkey V4. The model recovers the relevant features of the behavioural task and additionally unveils patterns of activity uncorrelated to the experimental paradigm.

**Keywords:** Non–stationary spiking data, Markov jump process, kinetic Ising model, monkey V4

## 1. Introduction

Technical advances in the field of extracellular recordings of ensembles of neurons (Stevenson and Kording, 2011) require novel and adequate analysis techniques to uncover how information is processed in the central nervous system. These techniques should account for the dynamic and correlated nature of these data (i.e. *spike trains*). The frequently made stationarity assumption is strongly violated, when data are recorded in-vivo from behaving animals. Even spike trains recorded from an animal at rest, i.e. spontaneous activity (Tsodyks et al., 1999), or recorded in–vitro (Sasaki et al., 2007) exhibit qualitative changes over time. Appropriate statistical description of these data thus requires flexible models.

This problem has been addressed by introducing latent hidden states, that follow a certain dynamics over the recording time. Different works consider either a continuous dynamics of the latent state (Yu et al., 2009; Lawhern et al., 2010; Zhao and Park, 2017), or piecewise constant and intermittently jumping states (Abeles et al., 1995; Escola et al., 2011; Putzky et al., 2014). For the latter type it is often additionally assumed, that latent dynamics can revisit a hidden state multiple times (Abeles et al., 1995; Putzky et al., 2014; Escola et al., 2011; Stimberg et al., 2012). These approaches are closely related to Dirichlet–processes (Teh, 2011). Despite the simplification introduced by these assumptions, experimental studies show that the approximations are plausible in some scenarios (Sasaki

et al., 2007; Latimer et al., 2015; Ponzi and Wickens, 2010). However, most of these works consider discrete time and hence require the binning of data recorded in continuous time.

To describe neuronal activity a popular model class is generalised linear models (GLMs) for Point processes (Pillow et al., 2008; Lawhern et al., 2010; Gerwinn et al., 2010). These models assume that observed spiking activity is a probabilistic process. The rate of this process is parametrised in various ways and usually depends on observed factors, e.g. stimulus, activity of other neurons, etc. Inference for GLMs is performed in several ways to obtain the maximum a–posteriori (MAP) estimate (Pillow et al., 2008; Tyrcha and Marsili, 2013), or a full posterior of the model parameters (Gerwinn et al., 2010). The latter (Bayesian) approach requires some kind of approximation, because the GLM likelihoods are of a form, for which the posterior solution is intractable. For time–discretised data the GLM (Bernoulli) likelihood can be rendered in a favourable form by variable augmentation (Polson et al., 2013). This form allows to efficiently sample the posterior (Linderman et al., 2016), or to obtain the MAP estimate by a fast expectation–maximisation (EM) algorithm (Scott and Pillow, 2012; Scott and Sun, 2013). However, in the continuous time limit this augmentation is not practical anymore, because GLM likelihoods involve an exponential whose argument contains an integral over time. Fortunately, recent work provides extended augmentation schemes (Donner and Opper, 2017, 2018) to deal with such likelihoods.

A specific case of a GLM in discrete time is the kinetic Ising model considering couplings to other observed neurons. Proposed in the 70's (Little, 1974) to describe computations of the brain it was recently utilised for statistical description of correlated spiking data (Tyrcha and Marsili, 2013; Dunn et al., 2015; Marre et al., 2009). An interesting extension of this model is the kinetic Ising model with *asynchronous updates* (Zeng et al., 2013), which allows to study the transition from discrete to continuous time. In the continuous time limit this model becomes a Markov jump process, where neurons can switch between on– and off–states. In statistical physics this model is known for describing dynamics of interacting spins, the so–called *Glauber dynamics* (Glauber, 1963). While the kinetic Ising model with asynchronous updates does not belong to the popular class of GLMs, it contains as subclass the equilibrium Ising model (Glauber, 1963), which has gained attention in the neuroscience community in recent years (Schneidman et al., 2006; Shlens and Field, 2006; Mana et al., 2018).

In this work we combine the kinetic Ising model with asynchronous updates with a latent state space dynamics in order to describe non–stationary continuous time spiking data of multiple neurons. First we describe the generative model for the discrete time case and obtain a new favourable form of the model likelihood by variable augmentation (section 2). This allows us to derive an efficient inference algorithm in section 3, that infers the couplings of different latent states, the time points the system spends in each state, and how likely are those states to appear in a fully Bayesian framework via variational methods. In section 4, we consider the continuous time limit of the model and the resulting inference algorithm. The model parameters can be correctly recovered on an artificial dataset. Finally, in section 5, we demonstrate practicality on a dataset of 40 multi– and single unit activity recorded in monkey V4.

## 2. Data and Generative Model

**Data** Binary spike train data $\boldsymbol{s}_{0:T}$ are represented in a $(T\Delta^{-1}) \times N$ dimensional matrix, where $N$ is the number of neurons observed, and $T$ is the recording length discretised in bins of width $\Delta$. Furthermore, we define the set of time points as $\mathcal{T} = \{\Delta, 2\Delta, \ldots, T\}$. At each time point $t \in \mathcal{T}$ the observed data of the network can be represented by a vector $(s_{t,1}, \ldots, s_{t,N})^\top$, where $s_{t,i} = 1$ if a neuron $i$ is active in the interval $t \in [t, t+\Delta)$ and $-1$ otherwise.

**Observation model** At each time point $t \in \mathcal{T}$ the data are sampled from a kinetic Ising model with asynchronous update Zeng et al. (2013) having time-dependent external fields $\boldsymbol{\theta}(t)$ and coupling matrix $\boldsymbol{J}_t$. It can be modelled a doubly stochastic process, where at each time point $t$ the activity of each neuron $i$ is *updated* with a probability $\Delta\gamma_s$. Only if the neuron is updated its state is flipped $s_{t+\Delta,i} = -s_{t,i}$ with probability

$$\mathbb{P}^{\text{flip}}_{t,i} = \frac{\exp\left(-s_{t,i}H_{t,i}\right)}{2\cosh\left(H_{t,i}\right)}, \tag{1}$$

where $H_{t,i} = \theta_{t,i} + \sum_{j=1}^{N} J_{t,ij}s_{t,j}$. $\theta_{t,i}$ denotes the external field of neuron $i$ and $J_{t,ij}$ is the coupling from neuron $j$ to neuron $i$ at time $t$. $\mathbb{P}$ and $\mathbb{Q}$ denote probabilities throughout this work, while $p$ and $q$ are densities with respective measures $P$ and $Q$. For notational convenience the vector form $H_{t,i} = \boldsymbol{J}_{t,i}^\top \boldsymbol{s}_t$ is introduced, where $\boldsymbol{J}_{t,i} = (J_{t,i0}, \ldots, J_{t,iN})^\top$, and $\boldsymbol{s}_t = (s_{t,0}, \ldots, s_{t,N})^\top$. In this notation the external fields are $J_{t,i0} = \theta_{t,i}$ and the data comprise a hidden neuron which is always active denoted by $s_{t,0} = 1$ for $t \in \{0\} \cup \mathcal{T}$. Furthermore, we define $\boldsymbol{J}_t = (\boldsymbol{J}_{t,1}, \ldots, \boldsymbol{J}_{t,N})$.

**Time dependence of the model parameters** We consider a sequence of latent states $z_{0:T} = (z_0, \ldots, z_T)$, where $z_t \in 1, \ldots, K$. Each latent state $k$ has a corresponding coupling matrix $\boldsymbol{J}_k$. If at time $t$ the latent state $z_t = k$, then $\boldsymbol{J}_t = \boldsymbol{J}_k$. Hence, we define the function $H_{t,i}(z) = \sum_{k=1}^{K} \delta_{z_t,k}\boldsymbol{J}_{k,i}^\top \boldsymbol{s}_t$ and $P^{\text{flip}}_{t,i}(z)$ according to (1). $\delta_{x,y}$ is the Kronecker delta. The likelihood of the data given the couplings of all states $\boldsymbol{J}_{1:K}$ and the latent state sequence $z_{0:T}$ is

$$\mathbb{P}(\boldsymbol{s}_{0:T}|\boldsymbol{J}_{1:K}, z_{0:T}) = \prod_{(t,i)\in\mathcal{F}} \Delta\gamma_s \mathbb{P}^{\text{flip}}_{t,i}(z) \prod_{(t,i)\in\mathcal{NF}} \left(1 - \Delta t\gamma_s \mathbb{P}^{\text{flip}}_{t,i}(z)\right), \tag{2}$$

where $\mathcal{F}$ and $\mathcal{NF}$ are the sets of $(t,i)$ pairs of flips and non flips, respectively. Furthermore, we define the sets $\mathcal{F}_i$ and $\mathcal{NF}_i$ containing only the (non) flip times of neuron $i$.

**Priors** The prior dynamics over the state variables is assumed to be a Markov chain. At each time point the latent state might switch with probability $\Delta\gamma_z$, where $\gamma_z$ is the *state switching rate*. If the latent state switches, the new state is chosen out of $K$ states according to a multinomial *state distribution* $\mathbb{P}(z_{t+\Delta} = k|\text{state switch at } t) = \pi_k$. Hence, the transition probability of the Markov chain is

$$\mathbb{P}(z|z', \pi_{1:K}) = \begin{cases} \Delta\gamma_z\pi_z \text{ if } z \neq z' \\ 1 - \Delta\gamma_z\left(1 - \pi_z\right) \approx \exp\left(-\Delta\gamma_z\left(1 - \pi_z\right)\right) \text{ else,} \end{cases}$$

where $\pi_{1:K} = (\pi_1, \ldots, \pi_K)^\top$ and the approximation holds for $\Delta\gamma_z << 1$ being exact in the limit of $\Delta \to 0$. The probability of a sequence $z_{0:T}$ is

$$\mathbb{P}(z_{0:T}|\gamma_z, \pi_{1:K}) = \prod_{t\in\mathcal{T}} \mathbb{P}_0(z_t|z_{t-\Delta}, \pi_{1:K})\mathbb{P}(z_0). \tag{3}$$

We define $\mathbb{P}_0(z_0 = k) = K^{-1}$ as the initial uniform state distribution at $t = 0$.

Because we expect sparse coupling structures in neuronal networks, we assume a Laplace prior for each of the couplings $J_{k,ij}$ for $j = 1, \ldots, N$ and $k = 1, \ldots, K$ with mean $\boldsymbol{\mu}_J = 0$ and scaling parameter $\sigma_J/\sqrt{N}$. The external fields $\theta_{k,i}$ have a *Gaussian prior* with mean $\boldsymbol{\mu}_\theta$ and variance $\sigma_\theta$. The state probabilities $\pi_{1:K}$ are distributed according to a *Dirichlet prior* with concentration parameters $\alpha/K$ for all states $k = 1, \ldots, K$. We avoid dealing with infinite Dirichlet–process priors by assuming a finite, but large enough $K$. Furthermore, we assume an exponential distribution as prior with mean 1 (which means if $\Delta$ is in units of seconds, we expect 1 state switch per second) over the switching rate $\gamma_z$.

**Data augmentation** Inference for the proposed model is difficult because of the non–conjugacy of likelihood Eq (2) to the priors. To render the likelihood and the prior into an alternate, conjugate Gaussian form we utilise the augmentation scheme developed by Donner and Opper (2017). Along these lines we make use of the well-known Pólya–Gamma representation (Polson et al., 2013) and rewrite the flip probability (1)

$$\mathbb{P}^{\text{flip}}_{t,i}(z) = \int_0^\infty \exp\left(-s_{t,i}H_{t,i}(z) - 2(H_{t,i}(z))^2\omega_{t,i} - \ln 2\right) p(\omega_{t,i})d\omega_{t,i}.$$

For notational convenience, we define

$$p^{\text{flip}}_{t,i}(z, \omega) \doteq \exp\left(-s_{t,i}H_{t,i}(z) - 2(H_{t,i}(z))^2\omega_{t,i} - \ln 2\right) p(\omega_{t,i}).$$

We write the factors of the second product in Eq (2) as an expectation over a binary variable $\rho_{t,i}$

$$(1 - \Delta\gamma_s\mathbb{P}^{\text{flip}}_{t,i}(z)) = (1 - \Delta\gamma_s + \Delta\gamma_s\mathbb{P}^{\text{noflip}}_{t,i}(z)) = \sum_{\rho_{t_i}\in\{0,1\}} \left(\mathbb{P}^{\text{noflip}}_{t,i}(z)\right)^{\rho_{i,t}} \text{Ber}(\rho_{i,t}|\Delta\gamma_s),$$

where $\text{Ber}(\rho|\Delta\gamma)$ denotes a Bernoulli distribution over random variable $\rho = 1$ with probability $\Delta\gamma$. Furthermore, we have

$$\mathbb{P}^{\text{noflip}}_{t,i}(z) = 1 - \mathbb{P}^{\text{flip}}_{t,i}(z) = \frac{\exp\left(s_{t,i}H_{t,i}(z)\right)}{2\cosh\left(H_{t,i}(z)\right)}.$$

By applying once more the Pólya–Gamma augmentation we get

$$(1 - \Delta\gamma_s\mathbb{P}^{\text{flip}}_{t,i}(z)) = \sum_{\rho_{t_i}\in\{0,1\}} \int_{\mathbb{R}^+} p^{\text{noflip}}_{t,i}(z, \rho, \omega)d\omega,$$

where

$$p^{\text{noflip}}_{t,i}(z, \rho, \omega) = \left(\exp\left(s_{t,i}H_{t,i}(z) - 2(H_{t,i}(z))^2\omega_{t,i} - \ln 2\right) p(\omega_{t,i})\right)^{\rho_{t,i}} \text{Ber}(\rho_{t,i}|\Delta\gamma_s).$$

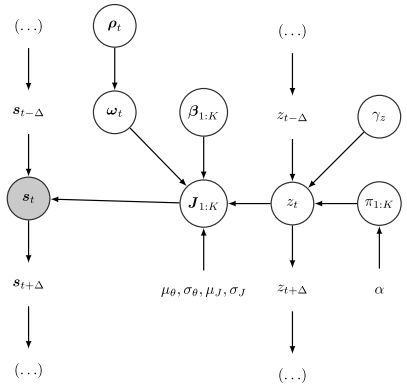



Figure 1: **Graphical model for data at time t.** Grey circle denotes observed data, and white circles random variables. Hyperparameters, data, and random variables at the neighbouring time–steps are depicted without circles.

This allows us to rewrite the joint augmented likelihood of Eq 2 as

$$p(\boldsymbol{s}_{0:T}, \omega_{\mathcal{F}}, (\rho,\omega)_{\mathcal{NF}} | \boldsymbol{J}_{1:K}, z_{0:T}) = \prod_{(t,i)\in\mathcal{F}} \Delta\gamma_s p^{\text{flip}}_{t,i}(z,\omega) \prod_{(t,i)\in\mathcal{NF}} p^{\text{noflip}}_{t,i}(z,\rho,\omega), \tag{4}$$

where $\omega_{\mathcal{F}}$ and $(\rho,\omega)_{\mathcal{NF}}$ are the sets of augmentation variables at the flip and non flip times, respectively.

For the prior over the couplings $J_{k,ij}$ we invoke the fact that the Laplace density can be rewritten as a scale Gaussian mixture model (Gao, 2008)

$$p(J_{k,ij}) = \frac{1}{2\sigma_J} \exp\left(-\frac{|J_{k,ij}|}{\sigma_J}\right) = \int \sqrt{\frac{\beta_{k,ij}}{2\pi\sigma_J^2}} \exp\left(-\frac{\beta_{k,ij}\,(J_{k,ij})^2}{2\sigma_J^2}\right) \eta(\beta_{k,ij}) d\beta_{k,ij},$$

where $\eta(\beta_{ij})$ is an *inverse gamma distribution* with shape parameter being 1 and scale parameter $\frac{1}{2}$. The new set of variables $\beta$ is named *sparsity variables*, and we define $\boldsymbol{\beta}_k$ being the sparsity variables for $\boldsymbol{J}_k$.

**Joint distribution** With the augmentation we have a model, that has a Gaussian form in terms of the couplings $\boldsymbol{J}_{1:K}$. The joint distribution of the model is

$$\begin{aligned} p(\boldsymbol{s}_{0:T}, \omega_{\mathcal{F}}, (\rho,\omega)_{\mathcal{NF}}, \boldsymbol{\beta}_{1:K}, z_{0:T}, \gamma_z, \pi_{1:K}, \boldsymbol{J}_{1:K} | \boldsymbol{\vartheta}) &= p(\boldsymbol{s}_{0:T}, \omega_{\mathcal{F}}, (\rho,\omega)_{\mathcal{NF}} | \boldsymbol{J}_{1:K}, z_{0:T}) \\ &\times \mathbb{P}(z_{0:T} | \pi_{1:K}, \gamma_z) p(\pi_{1:K} | \alpha) \\ &\times p(\boldsymbol{J}_{1:K}, \boldsymbol{\beta}_{1:K} | \boldsymbol{\mu}_\theta, \sigma_\theta, \mu_J, \sigma_J) p(\gamma_z), \end{aligned} \tag{5}$$

where the set of hyperparameters is denoted by $\boldsymbol{\vartheta} \doteq \{\boldsymbol{\mu}_\theta, \sigma_\theta, \gamma_s, \alpha, \boldsymbol{\mu}_J, \sigma_J\}$. Note, that by marginalising over the sets of augmented variables $\omega_{\mathcal{F}}, (\rho,\omega)_{\mathcal{NF}}, \boldsymbol{\beta}_{1:K}$ one obtains the joint distribution of the original model again. A graphical representation of the full model can be seen in Fig 1.

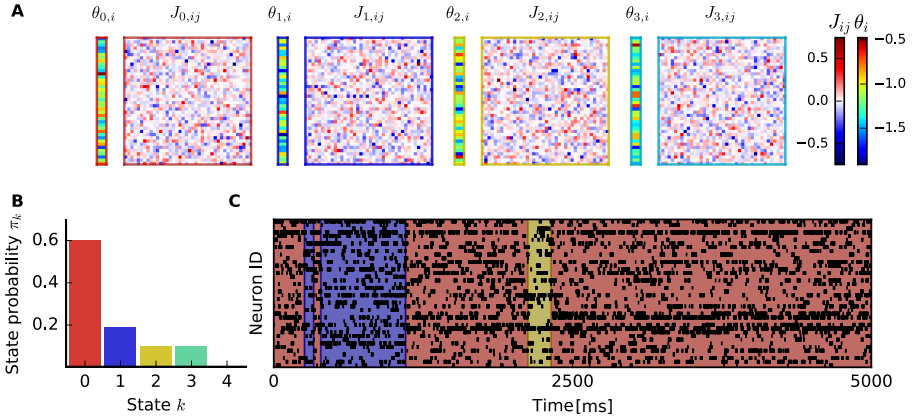



Figure 2: **Data from generative model. A** The kinetic Ising model parametrisations of the 4 most likely states, that generate the data. The vector on the left denote the external fields $\boldsymbol{\theta}_k$ and the matrices the couplings $\boldsymbol{J}_k$. The frame colour indicates the state identity. **B** State distribution sampled from a stick-breaking process with $\alpha = 1$. **C** Periods of the generated data, where each row is the activity of one neuron. Black indicates a neuron being in the active state. Colours indicate different states the data at a given time are sampled from.

**Continuous time limit** It is possible to obtain the continuous time limit $\Delta \to 0$ for the original and the augmented model in Eq (5). The prior of sequence $z_{0:T}$ in Eq (3), will become a Markov jump process density. The limit of the kinetic Ising model likelihood (2) and its augmented counterpart in Eq (4) has been already derived byDonner and Opper (2017). We refrain from taking the limit at this point and rather will derive an inference algorithm for the discrete time model, which provides efficient iterative updates for an approximate posterior. In the end we show that for each update the continuous time limit exists.

## 3. Variational Inference

While various efficient inference algorithms could be derived for the augmented model (see section 6), we resort to a hypothetical fast and fully Bayesian, but approximate variational approach. For solving the inference problem we make the structured mean–field assumption of the form

$$\begin{aligned} p(\omega_{\mathcal{F}}, (\rho,\omega)_{\mathcal{NF}}, \boldsymbol{\beta}_{1:K}, z_{0:T}, \gamma_z, \pi_{1:K}, \boldsymbol{J}_{1:K} | \boldsymbol{s}_{0:T}, \boldsymbol{\vartheta}) &\approx q(\omega_{\mathcal{F}}, (\rho,\omega)_{\mathcal{NF}}, \boldsymbol{\beta}_{1:K}, z_{0:T}, \gamma_z, \pi_{1:K}, \boldsymbol{J}_{1:K}) \\ &\doteq q_1(\omega_{\mathcal{F}}, (\rho,\omega)_{\mathcal{NF}}, \boldsymbol{\beta}_{1:K}, z_{0:T}) \\ &\quad \times q_2(\boldsymbol{J}_{1:K}, \pi_{1:K}) q_3(\gamma_z). \end{aligned} \tag{6}$$

We identify the *variational lower bound* on the logarithm of the marginal likelihood as

$$\mathcal{L}(q) = \mathbb{E}_Q\left[\ln \frac{p(\boldsymbol{s}_{0:T}, \omega_{\mathcal{F}}, (\rho,\omega)_{\mathcal{NF}}, z_{0:T}, \gamma_z, \pi_{1:K}, \boldsymbol{J}_{1:K}, \boldsymbol{\beta}_{1:K}|\boldsymbol{\vartheta})}{q(\omega_{\mathcal{F}}, (\rho,\omega)_{\mathcal{NF}}, \boldsymbol{\beta}_{1:K}, z_{0:T}, \gamma_z, \pi_{1:K}, \boldsymbol{J}_{1:K})}\right] \leq \ln p(\boldsymbol{s}_{0:T}|\boldsymbol{\vartheta}),$$

where $\mathbb{E}_Q[\cdot]$ is the expected value with respect to the corresponding variational posterior measure of the density in Eq (6). We make use of the fact, that the optimal factors of a mean–field posterior $\prod_x q_x(Y_x)$ are given by

$$\ln q_x(Y_x) = \mathbb{E}_{Q_{\setminus x}}\left[\ln p(\boldsymbol{s}_{0:T}, Y_x, Y_{\setminus x}|\boldsymbol{\vartheta})\right] + \text{const.}\,, \tag{8}$$

where the expectation is with respect to the posterior measure over all variables $Y_{\setminus x}$ except $Y_x$. We derive updates for each of the three variational factors in Eq (6) given the other two factors.

### 3.1 First factor

From the joint likelihood in Eq (5), the variational posterior in Eq (6), and Eq (8) we derive the factorisation

$$q_1(\omega_{\mathcal{F}}, (\rho,\omega)_{\mathcal{NF}}, \boldsymbol{\beta}_{1:K}, z_{0:T}) = q(\omega_{\mathcal{F}}|z_{0:T})q((\rho,\omega)_{\mathcal{F}}|z_{0:T})q(z_{0:T})q_1(\boldsymbol{\beta}_{1:K}),$$

meaning that $\omega_{\mathcal{F}}$ and $(\rho,\omega)_{\mathcal{NF}}$ are conditionally independent given $z_{0:T}$. The sparsity variables are independent of the rest $\boldsymbol{\beta}_{1:K}$.

**Augmented variables at the flip times** The conditional distribution turns out to be a product of tilted Pólya–Gamma densities

$$q_1(\omega_{\mathcal{F}}|z_{0:T}) = \prod_{(t,i)\in\mathcal{F}} p_{\mathrm{PG}}(\omega_{t,i}|1, c_{t,i}(z)) \propto \prod_{(t,i)\in\mathcal{F}} \exp\left(-\frac{[c_{t,i}(z)]^2}{2}\omega_{i,t}\right) p_{\mathrm{PG}}(\omega_{t,i}|1,0),$$

where $c_{t,i}(z) = 2\sqrt{\mathbb{E}_{\mathcal{Q}_{\setminus 1}}\left[(H_{t,i}(z))^2\right]}$. This defines the conditional expectation

$$\mathbb{E}_Q\left[\omega_{i,t}|z_t\right] = \frac{1}{2c_{t,i}(z)}\tanh\left(\frac{c_{t,i}(z)}{2}\right).$$

For future derivations we define

$$\hat{p}^{\text{flip}}_{t,i}(z) = \frac{\exp\left(-s_{t,i}\mathbb{E}_{Q_{\setminus 1}}\left[H_{t,i}(z)\right]\right)}{2\cosh\left(\frac{c_{t,i}(z)}{2}\right)}.$$

**Augmented variables at non-flips** For the conditional distribution of the $(\rho,\omega)_{t,i}$ pairs in $\mathcal{NF}$ we obtain a product of conditional Pólya–Gamma and Bernoulli distributions

$$q((\rho,\omega)_{\mathcal{NF}}|z_{0:T}) = \prod_{(t,i)\in\mathcal{NF}} [p_{\mathrm{PG}}(\omega_{t,i}|1, c_{t,i}(z))]^{\rho_{t,i}}\,\mathrm{Ber}(\rho_{t,i}|\mathbb{P}^{(\rho)}_{t,i}(z_t)), \tag{9}$$

where

$$\mathbb{P}^{(\rho)}_{t,i}(z) = \frac{\Delta\Lambda_{t,i}(z)}{(1-\Delta\gamma_s)+\Delta\Lambda_{t,i}(z)} \text{ with } \Lambda_{t,i}(z) = \gamma_s \frac{\exp\left(s_{t,i}\mathbb{E}_{Q_{\setminus 1}}\left[H_{t,i}(z)\right]\right)}{2\cosh\left(\frac{c_{t,i}(z)}{2}\right)}.$$

A random process as in Eq (9) can be easily sampled, by first sampling the $\rho$ sequence and then the variables $\omega_{t,i}$, where $\rho_{t,i} = 1$. The expectation over a pair of variables is given by

$$\mathbb{E}_Q\left[\omega_{t,i}\rho_{t,i}|z_t\right] = \mathbb{E}_Q\left[\omega_{t,i}|z_t\right]\mathbb{P}^{(\rho)}_{t,i}(z).$$

**State labels over time** Next we derive the variational posterior over the sequence of latent states $z_{0:T}$. With Eq (8) and marginalising out $\omega_{\mathcal{F}}$ and $(\omega,\rho)_{\mathcal{NF}}$, we obtain

$$\mathbb{Q}(z_{0:T}) \propto \prod_{t\in\mathcal{T}} \mathbb{P}_1(z_t|z_{t+\Delta}) \exp\left(-U_t(z_t, z_{t-\Delta})\right)\mathbb{P}(z_0), \tag{10}$$

where we defined a new effective transition matrix

$$\mathbb{P}_1(z|z',\Delta) = \delta_{z,z'} + \Delta\varphi(z|z'),$$

with effective rates being

$$\begin{aligned}\ln\varphi(z|z') &= \mathbb{E}_{Q_{\setminus 1}}\left[\ln\pi_k\right] + \mathbb{E}_{Q_{\setminus 1}}\left[\ln\gamma_z\right], \text{ if } z \neq z' \text{ and}\\ \varphi(z|z) &= -\sum_{z'\neq z}\varphi(z'|z).\end{aligned}$$

The $U$-function is given by

$$U_t(z,z') = U_t^{\text{data}}(z) + \tilde{U}(z,z'),$$

where the first term is the new effective negative data log–likelihood

$$U_t^{\text{data}}(z) = -\sum_{i=1}^{N}\left(\frac{\mathbf{1}_{\mathcal{F}}((t,i))}{\Delta}\ln\hat{p}^{\text{flip}}_{t,i}(z) + \Lambda_{t,i}(z)\right),$$

with $\mathbf{1}_{\mathcal{X}}(x)$ being 1 if $x \in \mathcal{X}$ and 0 otherwise. The second term

$$\tilde{U}(z,z') = \delta_{z,z'}\left(\frac{1}{\Delta}\ln(1+\Delta\varphi(z|z')) + \mathbb{E}_{Q_{\setminus 1}}\left[\gamma_z\right]\left(1-\mathbb{E}_{Q_{\setminus 1}}\left[\pi_z\right]\right)\right).$$

is a discrepancy term of the probability staying in the same state caused by the mean–field assumption in Eq (6). If the term would not involve expectations it would vanish (by noting that $\ln(1-\Delta\varphi(z_{t+\Delta}|z_t)) \approx \Delta\varphi(z_{t+\Delta}|z_t)$ for $\Delta\varphi(z_{t+\Delta}|z_t) << 1$). With Eq (10) we see immediately, that the posterior represents a Markov chain of the form

$$\mathbb{Q}(z_{0:T}) = \prod_{t\in\mathcal{T}}\mathbb{Q}_t(z_t|z_{t-\Delta})\mathbb{Q}_0(z_0), \tag{11}$$

where the factors have to be determined. Following standard procedures for hidden Markov models (Bishop, 2006) or minimising the Kullback–Leibler divergence (see appendix B) we can derive the transition probabilities

$$\mathbb{Q}_t(z|z') \propto \mathbb{P}_1(z|z', \Delta) \exp\left(-\tilde{U}(z, z')\Delta\right) r_t(z).$$

The $r_t$ factors are so-called backward messaged, that can be solved recursively by

$$r_t(z) = \sum_{z'} \mathbb{P}_1(z'|z, \Delta) \exp\left(-U_{t+\Delta}(z', z)\Delta\right) r_{t+\Delta}(z'), \tag{12}$$

where we initialise $r_T(z) = 1$ for $z = \{1, \ldots, K\}$. With these results we actually are able to solve for the factors of the Markov Chain in Eq (11), by first solving the backward equations and then obtaining the marginals by forward iterating Eq (11). However, we would like to have forward iterations being independent of the backward messages, such that we can parallelise their computation. In order to do so, we define the marginals as $\mathbb{Q}_t(z) \doteq f_t(z) \times r_t(z)$, where $f_t(z)$ are the forward messages. Consequently, we can derive iterative updates

$$f_t(z) = \sum_{z'} \mathbb{P}_1(z|z', \Delta) \exp\left(-U_t(z, z')\Delta\right) f_{t-\Delta}(z'), \tag{13}$$

which is independent of $r_t$ and where $f_0(z) = \mathbb{P}(z_0)$.

**Sparsity variables** Each sparsity variable $\beta_{k,ij}$ is independent of all other latent variables in the variational posterior density and hence can be separately updated. The approximate posterior density is *generalised inverse Gaussian density*

$$q(\beta_{k,ij}) = \frac{(a_{k,ij})^{-1/4}}{2K_{-1/2}\left(\sqrt{a_{k,ij}}\right)} (\beta_{k,ij})^{-\frac{3}{2}} \exp\left(-\frac{a_{k,ij}\beta_{k,ij} + 1/\beta_{k,ij}}{2}\right),$$

where $a_{k,ij} = \mathbb{E}_Q\left[\left(J_{k,ij} - \mu_J\right)^2\right] / \sigma_J^2$, and $K_y(x)$ is the modified Bessel function. The expectation over the sparsity variables are $\mathbb{E}_Q\left[\beta_{k,ij}\right] = (a_{k,ij})^{-1/2}$.

### 3.2 Second factor

For the second factor has a factorising form

$$q_2(\boldsymbol{J}_{1:K}, \pi_{1:K}) = q(\pi_{1:K}) \prod_{k=1}^{K} \prod_{i=1}^{N} q(\boldsymbol{J}_{k,i}),$$

meaning that state labels $\pi_{1:K}$ are independent of the couplings $\boldsymbol{J}_{1:K}$. Furthermore, the couplings $\boldsymbol{J}_{k,i}$ of each state $k$ and neuron $i$ are independent of all other couplings.

**Couplings** Due to the augmented model and the mean field assumption we are able to derive the posterior density for each vector $\boldsymbol{J}_{k,i} = \left(J_{i0}^k, \ldots, J_{iN}^k\right)^\top$ in a closed form. We

identify them being Gaussian densities with covariance matrix

$$\Sigma_{k,i} = \left( 4 \sum_{t \in \mathcal{T}} \left[ \mathbf{1}_{\mathcal{F}_i}(t) \mathbb{E}_{Q_{\setminus 2}} \left[ \delta_{z_t,k} \omega_{t,i} \right] + \mathbf{1}_{\mathcal{NF}_i}(t) \mathbb{E}_{Q_{\setminus 2}} \left[ \delta_{z_t,k} \rho_{t,i} \omega_{t,i} \right] \right] C_{t-\Delta} + \left( \tilde{\Sigma}_{k,i} \right)^{-1} \right)^{-1}, \tag{14}$$

where we defined the data covariance matrix $C_t = \boldsymbol{s}_t \boldsymbol{s}_t^\top$ and $\tilde{\Sigma}_{k,i}$ is a diagonal matrix with $\mathrm{diag}(\tilde{\Sigma}_{k,i}) = \left( \sigma_\theta^2, \frac{(\sigma_J)^2}{\mathbb{E}_Q[\beta_{k,i1}]}, \ldots, \frac{(\sigma_J)^2}{\mathbb{E}_Q[\beta_{k,iN}]} \right)$. For the mean we get

$$\boldsymbol{\mu}_{k,i} = \Sigma_i^k \left( \sum_{t \in \mathcal{T}} \left[ -\mathbf{1}_{\mathcal{F}_i}(t) \mathbb{E}_{Q_{\setminus 2}} \left[ \delta_{z_t,k} \right] + \mathbf{1}_{\mathcal{NF}_i}(t) \mathbb{E}_{Q_{\setminus 2}} \left[ \delta_{z_t,k} \rho_{t,i} \right] \right] C_{t-\Delta,i} + \left( \tilde{\Sigma}_{k,i} \right)^{-1} \boldsymbol{\mu}_J \right), \tag{15}$$

with $C_{t,i} = s_{t,i} \boldsymbol{s}_t$. Note, that those equations are analogue to the results obtained for the variational posterior by Donner and Opper (2017). They differ only because Eq (14) and (15) involve expectations of $\delta_{z_t,k}$, which act as 'weighting factors' for states $k$ at time point $t$.

**State probabilities** The variational posterior density of state probabilities cannot be obtained in close form. For this reason we further approximate $q(\pi_{1:K})$ with a Dirichlet distribution having concentration parameters $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K)^\top$. We maximise the variational lower bound (Eq 7) by differentiating it with respect to parameters $\boldsymbol{\alpha}$ and update them using a simple *gradient ascent* algorithm. Once convergence is achieved the distribution of state probabilities is given by

$$q(\pi_{1:K}) = \frac{\Gamma\left( \sum_{k=1}^K \alpha_k \right)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^K \pi_k^{\alpha_k - 1},$$

where $\Gamma(x)$ is the gamma function.

### 3.3 Third factor

**Switching rate** For the switching rate $\gamma_z$ the variational posterior density is a gamma density

$$q_3(\gamma_z) = \frac{\nu^\kappa}{\Gamma(\kappa)} \gamma_z^{\kappa - 1} \exp\left( -\nu \gamma_z \right), \tag{16}$$

with shape parameter and scale parameters

$$\begin{aligned} \kappa &= \sum_{t \in \mathcal{T}} \sum_z \left( 1 - \mathbb{Q}_t(z|z) \right) \mathbb{Q}_{t-\Delta}(z) + 1 \\ \nu &= \sum_{t \in \mathcal{T}} \sum_z \mathbb{Q}_t(z|z) \mathbb{Q}_{t-\Delta}(z) (1 - \mathbb{E}_Q[\pi_z]) \Delta + 1. \end{aligned}$$

**Variational lower bound** Since the assumed posterior density $q$ is now completely known and all required expectations can be derived analytically, it is straightforward to evaluate the variational lower bound in Eq 7 to check convergence after each update of the variational posterior.

## 4. The continuous time limit

We now study the limit $\Delta \to 0$ of the updates for the variational posterior density derived in section 3.

**First factor** The posterior densities of $\omega_{\mathcal{F}}$ and the sparsity variables $\boldsymbol{\beta}_{1:K}$ do not depend on $\Delta$ and consequently remain unchanged.

For the augmentation variables at the non flip times $(\rho, \omega)_{\mathcal{NF}}$ the variational posterior density in Eq (9) for $\Delta \to 0$ becomes

$$q(\Pi_{1:N}|z_{0:T}) \propto \prod_{i=1}^{N} \prod_{(t,\omega)\in\Pi_i} \Lambda_{t,i}(z,\omega) \exp\left(-\int_{\mathcal{T}\times\mathbb{R}^+} \Lambda_{t,i}(z,\omega) d\omega dt\right), \tag{17}$$

$\Pi_i = \{(t,\omega)\}$ is a random point set on the space $\mathcal{T} \times \mathbb{R}^+$ where $\rho_{t,i} = 1$. Eq (17) is proportional to a density of a *Poisson process* with respect to another homogeneous Poisson process on the same space (Konstantopoulos et al., 2011). Actually Eq (17) is an instance of a *marked Poisson process* (Kingman, 1993), where $\omega$ are the 'marks' at times with $\rho_{t,i} = 1$. $\Lambda_{t,i}(z_t, \omega) = p_{\mathrm{PG}}(\omega|1, c_{t,i})\Lambda_{t,i}(z_t)$ is the intensity of this Poisson process (see appendix A and Donner and Opper (2018)). The expectation over the Poisson process is (Kingman, 1993)

$$\mathbb{E}_Q\left[\sum_{(t,\omega)\in\Pi_i} h(t,\omega)\middle| z_{0:T}\right] = \int_{T\times\mathbb{R}^+} h(t,\omega)\Lambda_{t,i}(z,\omega)dt, \tag{18}$$

where the expectation is conditioned on the trajectory $z_{0:T}$. Note, that this result is similar to the one obtained by Donner and Opper (2017). The difference is for Donner and Opper (2017) the Poisson process rate was piecewise constant and hence the expectation can be written as sum over Poisson densities. Due to the expectation over the trajectory $z_{0:T}$ in Eq. (17) this is not the case in the current work.

For the recursive forward–backward Eqs (12) and (13) we show that they become ordinary differential equations in the continuous time limit $\Delta \to 0$. The forward equations are

$$\partial_t f_t(z) = \sum_{z'\neq z} (\varphi(z|z') f_t(z') - \varphi(z'|z) f_t(z)) - U_t(z) f_t(z),$$

and the backward equations

$$\partial_t r_t(z) = \sum_{z'\neq z} \varphi(z'|z)(r_t(z) - r_t(z')) + U_t(z) r_t(z).$$

The limit of the $U$-function is

$$\begin{aligned} U_t(z) = \lim_{\Delta\to 0} U_t(z,z') = -\sum_{i=1}^{N} \left( \sum_{(t',i')\in\mathcal{F}} \delta(t-t')\delta_{i,i'} \ln \hat{p}_{t,i}^{\mathrm{flip}}(z) + \Lambda_{t,i}(z) \right) \\ + \left(\varphi(z|z) + \mathbb{E}_{Q_{\setminus 1}}[\gamma_z]\left(1 - \mathbb{E}_{Q_{\setminus 1}}[\pi_z]\right)\right), \end{aligned}$$

with $\delta(x)$ being the Dirac delta function. For the infinitesimal transition probabilities we derive

$$\mathbb{Q}_t(z|z') \approx \delta_{z,z'} + dt g_t(z|z'),$$

with the transition rates

$$g_t(z|z') = \varphi(z|z')\frac{r_t(z)}{r_t(z')}.$$

This results are equivalent to the one, which were obtained by Opper and Sanguinetti (2008). For detailed derivation of the forward–backward equations see appendix B. While the differential equations above are exact in the limit, for practical integration schemes the discrete update (Eqs (12) and (13)) rules should be preferred, because they consider also the non–linear terms in $\Delta$, which guarantee $f_t$ and $r_t$ being non–negative.

**Second factor** The limit for the updates of the Gaussian posterior over $\boldsymbol{J}_{1:K}$ is straightforward, since the sums become integrals. For the covariance matrix in Eq (14) becomes

$$\Sigma_{k,i} = \left( 4 \int_{\mathcal{T}} \sum_{t' \in \mathcal{F}_i} \delta(t-t') C_t \mathbb{E}_{Q_{\setminus 2}} \left[\omega_{t,i}\right] q_t(k) dt \right.$$
$$\left. + 4 \int_{\mathcal{T}\times\mathbb{R}^+} C_t q_t(k) \Lambda_{t,i}(k,\omega) d\omega dt + \left(\tilde{\Sigma}_{k,i}\right)^{-1} \right)^{-1},$$

and equivalently the mean in Eq (15) is

$$\boldsymbol{\mu}_{k,i} = \Sigma_i^k \left( - \int_{\mathcal{T}} \sum_{t' \in \mathcal{F}_i} \delta(t-t') C_{t,i} q_t(k) dt + \int_{\mathcal{T}} C_{t,i} q_t(k) \Lambda_{t,i}(k) dt + \left(\tilde{\Sigma}_{k,i}\right)^{-1} \boldsymbol{\mu}_J \right).$$

For the update of state probabilities $\pi_{1:K}$ nothing changes, except that gradients of the lower bound in Eq (7) involve expectations over Poisson process defined by Eq (18) instead of Bernoulli variables.

**Third factor** The posterior over the state switching rate $\gamma_z$ remains a gamma density in the limit, where the parameters of Eq (16) become $\kappa = \int_{\mathcal{T}} \sum_z g_t(z|z) \mathbb{Q}_t(z) dt + 1$ and $\nu = \int_{\mathcal{T}} \sum_z \mathbb{Q}_t(z) dt (1 - \mathbb{E}_Q[\pi_z]) + 1$.

## 5. Results

To evaluate whether the derived variational inference algorithm a good approximate posterior we first generate data for $N = 40$ neurons with the generative model and compare the inference results with the ground truth. Finally, we fit the model to spiking data recorded from monkey V4 area in–vivo while the subject performed a perceptual task.

**Artificial Data** We first generate artificial data for with $N = 40$ neurons (see Fig 2). The state probability distribution in Fig 2 **B** is generated via a *stick-breaking process* with $\alpha = 1$. Note, that this generative process does not assume a finite number of states $K$. The external fields $\boldsymbol{\theta}_k$ are drawn from a Gaussian $\mathcal{N}(-1.2, 0.25)$, the couplings from a Laplace distribution ($\boldsymbol{\mu}_J = 0, \sigma_J = 0.5/\sqrt{N}$). The empirical probability of the resulting data for a neuron being active is $\sim 0.15$ which would correspond to an average firing rate of $\sim 15$ Hz. The switching rate is set to $\gamma_z = 3$ Hz and $\sim 16.5$ min of data are generated. Mainly 4 states are present the data, whose parameters $\boldsymbol{J}_k$ are shown in Fig 2 **A** and state probabilities are shown in Fig 2 **B**. 5s of example data are shown Fig 2 **C**. An active state of a neuron is denoted by black. The colouring shows at what times which state generated the data.

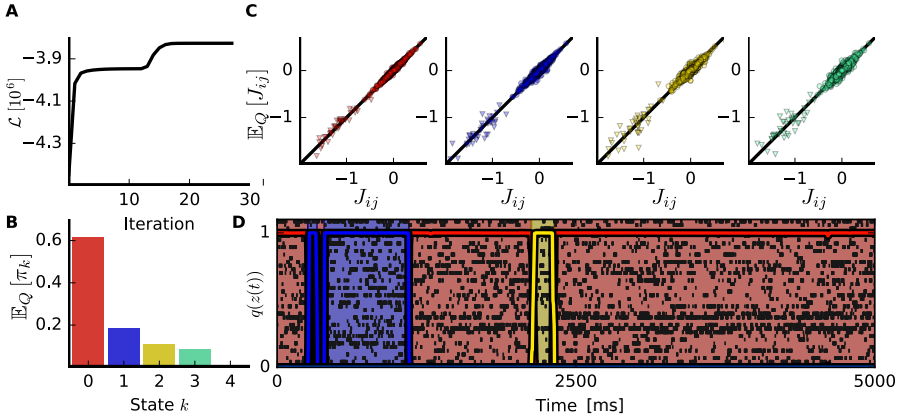



Figure 3: **Variational inference of the model from artificial data.** The data are the same as shown in Fig 2. **A** The lower bound as the number of iterations of the algorithm (outer loop iterations). **B** The inferred state probabilities. **C** The true couplings plotted against the inferred ones for each of the 4 states. The triangles indicate the external fields, and circles inditcate the couplings. **D** The inferred state label distribution $q(z_t)$ (solid lines, colour indicates state identity) with the spiking data and true labels again (coloured area) below.

**Practical implementation** Before discussing the inference results, we describe the practical realisation of the inference algorithm. For the initialisation we assume that all states are equal probable at all times, i.e. $\mathbb{Q}(z(t) = k) = 1/K$ for $t \in \mathcal{T}$. The mean for the couplings and external fields is set to the analytic solution for the assumption, that the network is uncoupled and the data are stationary. The standard deviation for the external fields is set to $\sigma_\theta = 1$, resulting in a relatively broad prior. To find the optimal value of the scaling parameter, the model is fit once with $\sigma_J \in \{0.01/\sqrt{N}, 0.1/\sqrt{N}, 1/\sqrt{N}\}$ and the result with the maximal lower bound is chosen. $\alpha$ is set to 0.5.

After initialisation, we define an inner and an outer loop. In the inner loop, all coupling relevant parameters (i.e. coupling $\boldsymbol{J}_{1:K}$, augmentation variables at flip times $\omega_{\mathcal{F}}$, and the latent Poisson process $\Pi_{1:N}$) are updated until the mean of couplings converges. Then in an outer loop the posterior over the of state labels $z_{0:t}$ is updated $q(z_{0:T})$, followed by the state probabilities $q(\pi_{1:K})$, and the switching rate $q(\gamma_z)$. Then the inner loop is repeated. When lower bound converges, the procedure stops. However, to check for convergence we compare the current value of lower bound and the one from 10 iterations before. This is necessary, because the algorithm often encounters a plateau, before finding a good configuration of state labels $z_{0:T}$ (see Fig 3**A**).

**Inference on artificial data** For the fit to artificial data, we consider an integration step of $\Delta = 1$ ms. The inference results are shown in Fig 3. The lower bound is increasing for each iteration (see Fig 3**A**) and the mean of the inferred variational posterior den-

sity for the state probabilities $\pi_{1:K}$ corresponds to the underlying values (compare Fig 2**B** with 3**B**). Furthermore, the couplings of all 4 states are recovered well (Fig 3**C**) and the state labels $z_{0:T}$ in the data are inferred accurately (Fig 3**D**). The mean of the switching rate $\mathbb{E}_Q[\gamma_z] = 3.095$ Hz is close to the ground truth. The fit with $\sigma_J = 0.1/\sqrt{N}$ yields the maximal lower bound.

**V4 Monkey Data** After validating the inference algorithm we analyse spiking data recorded from monkey V4. The data comprise activity from 40 multi– and single units recorded over 500 trials a 3 s. During each trial a either 0° or 90° drifting grating was presented to the monkey for 2 s. The experiment was performed at the University of Pittsburgh. All experimental procedures were approved by the University of Pittsburgh Institutional Animal Care and Use Committee, and were performed in accordance with the United States' National Institutes of Health (NIH) *Guide for the Care and Use of Laboratory Animals*. (For details see Snyder et al. (2015)). Exemplary data are shown in Fig 4 **A**. For the model inference all 500 trials are concatenated, i.e. the lack of data between trials is ignored.

The inference results are shown in Fig 4. In Fig 4**A** top and centre panel, we see that state switches appear shortly after stimulus on and offset (trial averaged probability of state switching $\mathbb{E}_{\text{trials}}[\sum_z (1 - \Delta g_t(z|z))\mathbb{Q}_t(z)]$ peaks at 56 ms, not shown here). Stimulus specific states can be identified by investigation of the averaged state label probabilities $q(z_t)$ over 90° and 0° trials, respectively (Fig 4 **A** Bottom). However, there are differences between trials even under same stimulus condition. Some trials show few state label switches during the stimulus phase (e.g. Fig 4 **A** Top). In contrast, other trials yield several switches (e.g. Fig 4 **A** Centre). Note, that these differences across trials can only be seen because the model allows for cross–trial variability.

Since the states are defined by the couplings and external fields we show the posterior mean of parameters for 4 different exemplary states (Fig 4**B**). The 1$^{\text{st}}$ (blue) state from the left is mainly inferred for periods before and after the stimulus (Fig 4**A**). However, brief irregularly appearing periods during stimulus presentation with collective sparse firing are also assigned to this state. The 2$^{\text{nd}}$ (dark green) state frequently appears shortly after stimulus onset. The last to states are stimulus selective (3$^{\text{rd}}$ (lavender) for 90° and 4$^{\text{th}}$ (orange) for 0°). The inferred mean of the inferred posterior over the couplings $J_{k,ij}$ shows sparse connectivity structures among all states (i.e. most $J_{k,ij} \approx 0$). However, for the 1$^{\text{st}}$ (blue) state we observe more positive couplings than the last two (lavender and orange) stimulus states indicating more uncorrelated activity with the stimulus present.

To show why considering single–trial analysis is important we focus on the stimulus period (from 200 ms to 2 s, to exclude the transient at stimulus onset). We align the population averaged activity data during this period to the onsets inferred for each of the 4 states shown in Fig 4 **B**. A state onset is defined as the time point a state becomes the most likely one. We average over all inferred state onsets, resulting in the state–specific average population activity (Fig 4**C** Top). For the 1$^{\text{st}}$ state we observe a strong drop in the fraction of active units after stimulus onsets. The 2$^{\text{nd}}$ state shows a drop before the state onset (likely caused by the frequently preceding 1$^{\text{st}}$ state) followed by a strong increase in population activity. The stimulus states show a brief activity increase. Additionally, the

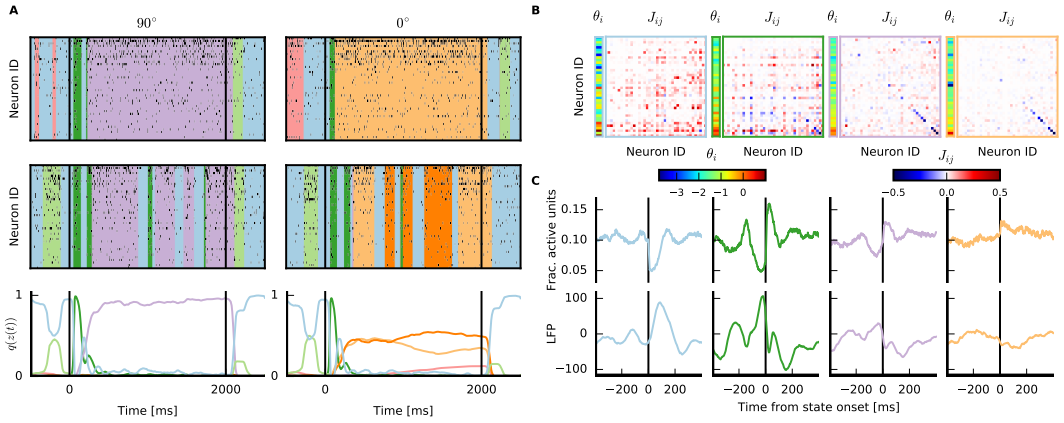



Figure 4: **Model fit to V4 monkey data** 500 trials recorded in–vivo while the monkey performed a perceptual task. Spiking data of 40 units are analysed. **A** Left: Two individual trials where the stimulus is a 90° drifting grating. Colours indicate the identity of the most likely state $z_t$ at each time point. The lower row shows the state label probabilities $q(z_t)$ averaged over all 90° trials. Right: Same as left for trails where stimulus is a 0° drifting grating. **B** Posterior mean of external fields and couplings for 4 different states. From left to right: The first (blue) state is prominent during non-stimulus periods and during stimulus periods when the network exhibits collectively sparse activity. The second (dark green) state appears most frequently after stimulus onset. The last two states are stimulus selective ($3^{\text{rd}}$ (lavender) for 90° and $4^{\text{th}}$ (orange) for 0°). **C** Average population activity (top) and the LFP (bottom) aligned with the state onsets. Only periods that appeared between 0.2 s and 2 s after stimulus onsets are considered.

averaged local field potential (LFP) - recorded simultaneously with the spike data - shows a strong signature of the 1st state as well as for the 2nd state (Fig 4**C** Bottom).

## 6. Discussion

To statistically describe non–stationary data recorded from multiple neurons we presented a model, which considers couplings among recorded neurons and does not require discretisation of the observed data. The proposed variational algorithm infers hidden states in an unsupervised manner, and hence forms a flexible analysis framework for these data.

We utilised the augmentation scheme of (Donner and Opper, 2017) to render the data likelihood of the kinetic Ising model into a Gaussian form with respect to the model parameters $\boldsymbol{J}_{1:K}$. Furthermore, the Laplace prior is written as a Gaussian mixture model. This allowed for the derivation of an efficient variational mean–field algorithm, where, apart from the state–probabilities, all updates can be computed analytically.

Furthermore, the augmented form of the model allows for alternative inference schemes. For example, an empirical Bayes approach is possible, where one derives an expectation–maximisation (EM) algorithm (Dempster et al., 1977). For this we consider only the point estimate of the couplings $\boldsymbol{J}_{1:K}$, the state probabilities $\pi_{1:K}$, and the state jump rate $\gamma_z$. This allows us to define the well–known Q-function for the model, which can iteratively be maximised. In the E–step one computes analytically the posterior over state–labels $z_{0:T}$ and augmented variables $\omega_{\mathcal{F}}, (\rho, \omega)_{\mathcal{NF}}, \boldsymbol{\beta}_{1:K}$. In the M–step the Q–function is optimised with respect to the variables $\boldsymbol{J}_{1:K}, \pi_{1:K}, \gamma_z$. For the couplings $\boldsymbol{J}_{1:K}$ one has only to solve linear system of equations. Since $\pi_{1:K}, \gamma_z$ are coupled to each other, one requires some numerical optimisation method to maximise the Q-function, e.g. gradient ascent or second order methods. This algorithm is then guaranteed to converge to a (local) maximum of the non–augmented (penalised) model likelihood.

Alternatively the augmentations allow for a Markov chain Monte Carlo scheme. The variables $\boldsymbol{J}_{1:K}, \omega_{\mathcal{F}}, (\rho, \omega)_{\mathcal{NF}}, \gamma_z$ can be sampled directly from their conditional posterior. For the sparsity variables $\boldsymbol{\beta}_{1:K}$ the conditional posterior is – as in the variational case – a generalised inverse Gaussian, which can be sampled via efficient rejection sampling (Atkinson, 1982). Another option is to assume a spike–and–slab prior, which can be sampled efficiently (Linderman et al., 2016). For the state probabilities $\pi_{1:K}$ one could in principle assume a Chinese restaurant process prior, such that the number of states $K$ does not need to be fixed. Then the state labels $z_{0:T}$ and probabilities $\pi_{1:K}$ could be sampled by a MCMC procedure (Stimberg et al., 2012).

The augmentation scheme utilised in this work can be applied to generalised linear models (GLMs) with point process likelihood having an intensity function of the form $\Lambda_t = \gamma\sigma(h_t(\cdot))$, where $\sigma(\cdot)$ is the sigmoid link function and $h_t(\cdot)$ depends on observed covariates and linearly on the model parameters. For models with such likelihoods and Gaussian priors, tractable posteriors over the model parameters can be obtained efficiently (Donner and Opper, 2018). The GLM with sigmoid link function has been shown to be advantageous for describing spiking activity compared to the more classical choice $\Lambda_t = \exp(h_t(\cdot))$ (Capone et al., 2018). The consideration of continuous time allows to avoid the question of choosing the bin–size for time discretisation. Consequently, optimisation of this parameter is not required, as in contrast to more traditional paradigms. Admittedly, while preserving the

temporal precision of the data, the difficult choice of bin size is shifted to the question of how long a neuron is considered to be 'active' after a spike. This can be circumvented with GLMs that have parametrised history kernels. Here we focused on the kinetic Ising model because it allows interpolating between discrete and continuous time (Zeng et al., 2013).

The methodology presented here can also be used for models with continuous dynamics. The augmentation scheme can straightforwardly be applied to the *Gaussian process factor analysis* (GPFA) (Yu et al., 2009). To preserve tractability of the model Yu et al. (2009) assumed, that the observed square root spike–count is Gaussian distributed, while a Poisson distribution would be a more natural choice. While the treatment of Poisson likelihoods for discrete time (Nam, 2015) and Point process likelihoods for continuous time (Duncker and Sahani, 2018) were already addressed, the augmentations developed in Donner and Opper (2017) and the present work probably allow for efficient inference schemes for the GPFA model.

Non–stationary models as the one presented here allow for statistical description of single–trial data. In experiments with repetitive trial structure we have demonstrated that our model can uncover variations across trials. Not taking these into account might lead to spurious results Brody (1999).

In conclusion, this work represents a step towards investigating recorded neural networks in continuous time when knowledge of the underlying dynamics is scarce.

## Appendix A. Continuous time limit for posterior of latent variables at non flip times

We want to derive the limit $\Delta \to 0$ of the variational posterior density

$$q((\rho,\omega)_{\mathcal{NF}}|z_{0:T}) = \prod_{i=1}^{N} \prod_{t\in\mathcal{NF}_i} [p_{\mathrm{PG}}(\omega_{t,i}|1, c_{t,i}(z))]^{\rho_{t,i}} \,\mathrm{Ber}(\rho_{t,i}|\mathbb{P}_{t,i}^{(\rho)}(z)).$$

First, we note that in the limit $\Delta \to 0$ becomes the whole space $\mathcal{NF}_i \to \mathcal{T}$. We define a set $\Pi_i = \{(t,\omega)\}$ containing all the time points where $\rho_{t,i} = 1$ and write

$$q((\rho,\omega)_{\mathcal{NF}}|z_{0:T}) = \prod_{i=1}^{N} \prod_{(t,\omega)\in\Pi_i} p_{\mathrm{PG}}(\omega_{t,i}|1, c_{t,i}(z))\mathbb{P}_{t,i}^{(\rho)}(z) \prod_{(t,\omega)\notin\Pi_i} (1-\mathbb{P}_{t,i}^{(\rho)}(z)).$$

The continuous time limit of this expression is

$$q(\Pi_{1:N}|z_{0:T}) \propto \prod_{i=1}^{N} \Lambda_{t,i}(z,\omega) \exp\left(-\int_{\mathcal{T}\times\mathbb{R}^+} \Lambda_{t,i}(z,\omega) d\omega dt\right),$$

which is proportional to a product of Poisson process densities over the space $\mathcal{T}\times\mathbb{R}^+$, where the densities are defined with respect to a homogeneous Poisson process measure (Konstantopoulos et al., 2011).

## Appendix B. Derivation of forward–backward equations for a Markov jump process

In this section we show how to derive the forward–backward equations for a Markov jump process. As we will show, the final results are the same as in Opper and Sanguinetti (2008). While Opper and Sanguinetti (2008) consider the limit $\Delta \to 0$ from the start we derive the forward–backward equations first for a Markov chain and then take the limit in the end. In practice, a finite $\Delta$ needs to be chosen and the subsequent derivations yield a preferable integration scheme. As shown the variational posterior density has the form

$$\mathbb{Q}(z_{0:T}) \propto \tilde{\mathbb{Q}}(z_{0:T}) = \prod_{t\in\mathcal{T}} \mathbb{P}_1(z_t|z_{t+\Delta}) \exp\left(-U(z_t, z_{t-\Delta})\Delta\right) \mathbb{P}(z_0),$$

We want to determine the Markov chain factors

$$\mathbb{Q}(z_{0:T}) = \prod_{t\in\mathcal{T}} \mathbb{Q}_t(z_t|z_{t-\Delta})\mathbb{Q}_0(z_0).$$

The objective function we minimise is the Kullback–Leibler divergence between $\mathbb{Q}$ and the unnormalised $\tilde{\mathbb{Q}}$ as

$$\begin{aligned}\mathrm{D}_{\mathrm{KL}}\left(\mathbb{Q}\|\tilde{\mathbb{Q}}\right) = \mathbb{E}_{\mathbb{Q}}\left[\ln\frac{\mathbb{Q}}{\tilde{\mathbb{Q}}}\right] &= \sum_{t\in\mathcal{T}}\sum_{z,z'} \mathbb{Q}_t(z|z')\mathbb{Q}_{t-\Delta}(z) \ln\frac{\mathbb{Q}_t(z|z')}{\mathbb{P}_1(z|z')\exp\left(-\tilde{U}_t(z,z')\Delta\right)} \\ &\quad + \sum_{t\in\mathcal{T}}\sum_{z} \mathbb{Q}_t(z)U_t^{\mathrm{data}}(z)\Delta + \sum_{z} \mathbb{Q}_0(z)\ln\frac{\mathbb{P}_0(z)}{\mathbb{Q}_0(z)}.\end{aligned}$$

The factors of the Markov chain need to fulfil the marginalisation constraints

$$1 = \sum_{z'} \mathbb{Q}_t(z'|z) \tag{19}$$

$$\mathbb{Q}_t(z) = \sum_{z'} \mathbb{Q}_t(z|z')\mathbb{Q}_{t-\Delta}(z'). \tag{20}$$

For the moment we only consider only the constraint in Eq (20) and will take care of the one in Eq (19) during the derivations. The constraint optimisation problem is

$$\mathcal{L}_z = \mathrm{D}_{\mathrm{KL}}\left(\mathbb{Q}\|\tilde{\mathbb{Q}}\right) + \sum_{t\in\mathcal{T}}\sum_{z} \lambda_t(z)\left(\mathbb{Q}_t(z) - \sum_{z'} \mathbb{Q}_t(z|z')\mathbb{Q}_{t-\Delta}(z')\right), \tag{21}$$

where $\lambda_t(z)$ are the Lagrangian multipliers. Taking the derivative with respect to $\mathbb{Q}_t(z|z')$, setting it to 0, and considering the normalisation constraint Eq (19) we derive

$$\mathbb{Q}_t(z|z') = \frac{\mathbb{P}_1(z|z')\exp\left(-\tilde{U}_t(z,z')\Delta + \lambda_t(z)\right)}{\sum_z \mathbb{P}_1(z|z')\exp\left(-\tilde{U}_t(z,z')\Delta + \lambda_t(z)\right)}. \tag{22}$$

We take the derivative with respect to $\mathbb{Q}_t(z_t)$ of Eq (21) and equate it to 0

$$\begin{aligned}\frac{\partial\mathcal{L}_z}{\partial\mathbb{Q}_t(z)} = &\sum_{z'} \mathbb{Q}_{t+\Delta}(z'|z)\left(-\ln\mathbb{P}_1(z|z') + \tilde{U}_{t+\Delta}(z',z)\Delta + \ln\mathbb{Q}_{t+\Delta}(z'|z) - \lambda_{t+\Delta}(z')\right)\\ &+ \lambda_t(z) + U_t^{\mathrm{data}}(z_t)\Delta = 0.\end{aligned}$$

By inserting (22) into this result and defining $r_t(z) = e^{\lambda_t(z)}$ we get the backward equations

$$r_t(z) = \sum_{z'} \mathbb{P}_1(z'|z,\varphi_{1:K},\Delta)\exp\left(-U_{t+\Delta}(z',z)\Delta\right) r_{t+\Delta}(z'). \tag{23}$$

This allows us already to solve of the inferring the distribution over the Markov Chain since we can first recursively calculate $r_t$. We then solve for the marginals by iterating forward

$$\mathbb{Q}_t(z) = \sum_{z'} \mathbb{Q}_t(z|z')\mathbb{Q}_{t-\Delta}(z'). \tag{24}$$

Since we consider quite long Markov chains, we would like to decouple these forward–backward iterations, such that they can be parallelised. In order to do so, we define $\mathbb{Q}_t(z) \propto f_t(z) \times r_t(z)$. Substituting this into Eq (24), and with Eqs (23) and (22) we derive the forward equations

$$f_t(z) = \sum_{z'} \mathbb{P}_1(z|z',\varphi_{1:K},\Delta)\exp\left(-U_t(z,z')\Delta\right) f_{t-\Delta}(z'), \tag{25}$$

which do not depend on $r_t$.

**Continuous time limit** For $\Delta \to 0$ the backward and forward equations (23) and (25) become differential equations of the form

$$\partial_t r_t(z) = \sum_{z' \neq z} \varphi(z'|z)(r_t(z) - r_t(z')) + U_t(z) r_t(z),$$
$$\partial_t f_t(z) = \sum_{z' \neq z} (\varphi(z|z') f_t(z') - \varphi(z'|z) f_t(z)) - U_t(z) f_t(z),$$

where $U_t(z) = \lim_{\Delta \to 0} U_t(z, z')$. For these results we neglected the higher order term $\mathcal{O}(\Delta^2), \mathcal{O}(\Delta^3)$, etc. The limit for the transition probability in Eq (22) becomes

$$\mathbb{Q}_t(z|z') \approx \delta_{z,z'} + dt g_t(z|z'), \tag{26}$$

where

$$g_t(z|z') = \phi(z|z') \frac{r(z)}{r(z')}.$$

Interestingly, the *master equation* of a Markov jump process can be derived from Eq (26) and the limiting case of the constrains in Eqs (19) and (20) being

$$\partial_t \mathbb{Q}_t(z_t) = \sum_{z' \neq z} g_t(z|z') \mathbb{Q}_t(z') - g_t(z'|z) \mathbb{Q}_t(z).$$

This constraint is considered from the start by Opper and Sanguinetti (2008). While from a theoretical point of view the limiting equations come out very elegant, for practical reasons, an integration scheme using Eqs (23) and (25) is preferred, because these equations consider also higher order terms of the differential equations of the finite integration step size $\Delta$ and ensure, that $r_t$ and $f_t$ are always non–negative.

## References

M. Abeles, H. Bergman, I. Gat, I. Meilijson, E. Seidemann, N. Tishby, and E. Vaadia. Cortical activity flips among quasi-stationary states. *Proceedings of the National Academy of Sciences*, 92(19):8616–8620, 1995. ISSN 0027-8424. doi: 10.1073/pnas.92.19.8616. URL `http://www.pnas.org/cgi/doi/10.1073/pnas.92.19.8616`.

A. C. Atkinson. The Simulation of Generalized Inverse Gaussian and Hyperbolic Random Variables. *SIAM Journal on Scientific and Statistical Computing*, 3(4):502–515, dec 1982. ISSN 0196-5204. doi: 10.1137/0903033. URL `http://epubs.siam.org/doi/10.1137/0903033`.

Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, 2006. ISBN 9780387310732.

Carlos D. Brody. Correlations Without Synchrony. *Neural Computation*, 11(7):1537–1551, oct 1999. ISSN 0899-7667. doi: 10.1162/089976699300016133. URL `http://www.mitpressjournals.org/doi/10.1162/089976699300016133`.

Cristiano Capone, Guido Gigante, and Paolo Del Giudice. Spontaneous activity emerging from an inferred network model captures complex spatio-temporal dynamics of spike data. *Scientific Reports*, 8(1):17056, dec 2018. ISSN 2045-2322. doi: 10.1038/s41598-018-35433-0. URL http://www.nature.com/articles/s41598-018-35433-0.

Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. ISSN 0035-9246. doi: 10.1.1.133.4884. URL http://www.jstor.org/stable/2984875.

Christian Donner and Manfred Opper. Inverse Ising problem in continuous time: A latent variable approach. *Physical Review E*, 96(6):062104, dec 2017. ISSN 24700053. doi: 10.1103/PhysRevE.96.062104. URL https://link.aps.org/doi/10.1103/PhysRevE.96.062104.

Christian Donner and Manfred Opper. Efficient Bayesian Inference of Sigmoidal Gaussian Cox Processes. *Journal of Machine Learning Research*, 19(67):1–34, 2018. ISSN 15337928. URL http://www.jmlr.org/papers/v19/17-759.htmlhttp://arxiv.org/abs/1808.00831.

Lea Duncker and Maneesh Sahani. Temporal alignment and latent Gaussian process factor inference in population spike trains. *bioRxiv*, page 331751, may 2018. doi: 10.1101/331751. URL https://www.biorxiv.org/content/early/2018/05/27/331751.

Benjamin Dunn, Maria Mørreaunet, and Yasser Roudi. Correlations and Functional Connections in a Population of Grid Cells. *PLoS Computational Biology*, 11(2):e1004052, feb 2015. ISSN 15537358. doi: 10.1371/journal.pcbi.1004052. URL https://dx.plos.org/10.1371/journal.pcbi.1004052.

Sean Escola, Alfredo Fontanini, Don Katz, and Liam Paninski. Hidden Markov Models for the Stimulus-Response Relationships of Multistate Neural Systems. *Neural Computation*, 23(5):1071–1132, may 2011. ISSN 0899-7667. doi: 10.1162/NECO_a_00118. URL http://www.mitpressjournals.org/doi/10.1162/NECO_a_00118.

Junbin Gao. Robust L1 principal component analysis and its Bayesian variational inference. *Neural Computation*, 20(2):555–572, feb 2008. ISSN 08997667. doi: 10.1162/neco.2007.11-06-397. URL http://www.mitpressjournals.org/doi/10.1162/neco.2007.11-06-397.

Sebastian Gerwinn, Jakob H Macke, and Matthias Bethge. Bayesian inference for generalized linear models for spiking neurons. *Frontiers in Computational Neuroscience*, 4:12, may 2010. ISSN 16625188. doi: 10.3389/fncom.2010.00012. URL http://journal.frontiersin.org/article/10.3389/fncom.2010.00012/abstract.

Roy J. Glauber. Time-dependent statistics of the Ising model. *Journal of Mathematical Physics*, 4(2):294–307, feb 1963. ISSN 00222488. doi: 10.1063/1.1703954. URL http://aip.scitation.org/doi/10.1063/1.1703954.

John F C Kingman. *Poisson processes*. Clarendon Press, 1993. ISBN 9780198536932. URL https://global.oup.com/academic/product/poisson-processes-9780198536932?cc=de&lang=en&.

Takis Konstantopoulos, Zurab Zerakidze, and Grigol Sokhadze. RadonNikodým Theorem. In *International Encyclopedia of Statistical Science*, pages 1161–1164. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. doi: 10.1007/978-3-642-04898-2_468. URL http://link.springer.com/10.1007/978-3-642-04898-2_468.

Kenneth W Latimer, Jacob L Yates, Miriam L R Meister, Alexander C Huk, and Jonathan W Pillow. Single-trial spike trains in parietal cortex reveal discrete steps during decision-making. *Science (New York, N.Y.)*, 349(6244):184–7, jul 2015. doi: 10.1126/science.aaa4056. URL http://www.ncbi.nlm.nih.gov/pubmed/26160947.

Vernon Lawhern, Wei Wu, Nicholas Hatsopoulos, and Liam Paninski. Population decoding of motor cortical activity using a generalized linear model with hidden states. *Journal of Neuroscience Methods*, 189(2):267–280, jun 2010. ISSN 01650270. doi: 10.1016/j.jneumeth.2010.03.024. URL http://linkinghub.elsevier.com/retrieve/pii/S0165027010001585.

Scott W. Linderman, Ryan P. Adams, and Jonathan W. Pillow. Bayesian latent structure discovery from multi-neuron recordings. In *Advances in neural information processing systems*, pages 2002–2010, 2016. doi: 10.2307/633674. URL http://papers.nips.cc/paper/6185-bayesian-latent-structure-discovery-from-multi-neuron-recordings.

W. A. Little. The existence of persistent states in the brain. *Mathematical Biosciences*, 19(1-2):101–120, feb 1974. ISSN 00255564. doi: 10.1016/0025-5564(74)90031-5. URL https://www.sciencedirect.com/science/article/pii/0025556474900315.

PierGianLuca Porta Mana, Vahid Rostami, Emiliano Torre, and Yasser Roudi. Maximum-entropy and representative samples of neuronal activity: a dilemma. may 2018. URL http://arxiv.org/abs/1805.09084.

O. Marre, S. El Boustani, Y. Frégnac, and A. Destexhe. Prediction of spatiotemporal patterns of neural activity from pairwise correlations. *Physical Review Letters*, 102(13): 138101, apr 2009. ISSN 00319007. doi: 10.1103/PhysRevLett.102.138101. URL https://link.aps.org/doi/10.1103/PhysRevLett.102.138101.

Hooram Nam. Poisson Extension of Gaussian Process Factor Analysis for Modelling Spiking Neural Populations, 2015. URL https://pdfs.semanticscholar.org/fc4c/e9761aea889a3a733278796f78544e5b8634.pdf.

Manfred Opper and Guido Sanguinetti. Variational inference for Markov jump processes, 2008. URL http://papers.nips.cc/paper/3296-variational-inference-for-markov-jump-processes.

Jonathan W. Pillow, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M. Litke, E. J. Chichilnisky, and Eero P. Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999, 2008. ISSN 00280836. doi: 10.1038/nature07140.

Nicholas G. Polson, James G. Scott, and Jesse Windle. Bayesian inference for logistic models using Pólya-Gamma latent variables. *Journal of the American Statistical Association*, 108

(504):1339–1349, dec 2013. ISSN 1537274X. doi: 10.1080/01621459.2013.829001. URL http://www.tandfonline.com/doi/abs/10.1080/01621459.2013.829001.

Adam Ponzi and Jeff Wickens. Sequentially Switching Cell Assemblies in Random Inhibitory Networks of Spiking Neurons in the Striatum. *Journal of Neuroscience*, 30(17):5894–5911, 2010. ISSN 0270-6474. doi: 10.1523/JNEUROSCI.5540-09.2010. URL http://www.jneurosci.org/cgi/doi/10.1523/JNEUROSCI.5540-09.2010.

P. Putzky, F. Franzen, G. Bassetto, and J. H. Macke. A Bayesian model for identifying hierarchically organised states in neural population activity. In *Advances in Neural Information Processing Systems*, pages 3095–3103, 2014. URL http://papers.nips.cc/paper/5338-a-bayesian-model-for-identifying-hierarchically-organised-states-in-neural-population-activity.

T. Sasaki, N. Matsuki, and Y. Ikegaya. Metastability of Active CA3 Networks. *Journal of Neuroscience*, 27(3):517–528, jan 2007. ISSN 0270-6474. doi: 10.1523/JNEUROSCI.4514-06.2007. URL http://www.jneurosci.org/cgi/doi/10.1523/JNEUROSCI.4514-06.2007.

E Schneidman, MJ Berry, R Segev, and W Bialek. Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature*, 440(7087):1007, 2006. URL http://www.nature.com/nature/journal/v440/n7087/abs/nature04701.html.

James Scott and Jonathan W Pillow. Fully Bayesian inference for neural models with negative-binomial spiking. *Advances in Neural Information Processing Systems*, 25:1898–1906, 2012. ISSN 10495258. URL http://papers.nips.cc/paper/4567-fully-bayesian-inference-for-neural-models-with-negative-binomial-spiking.

James G. Scott and Liang Sun. Expectation-maximization for logistic regression. may 2013. URL http://arxiv.org/abs/1306.0040.

J Shlens and GD Field. The structure of multi-neuron firing patterns in primate retina. *The Journal of Neuroscience*, 26(32):8254–8266, 2006. URL http://www.jneurosci.org/content/26/32/8254.short.

Adam C. Snyder, Michael J. Morais, Cory M. Willis, and Matthew A. Smith. Global network influences on local functional connectivity. *Nature Neuroscience*, 18(5):736–743, 2015. ISSN 15461726. doi: 10.1038/nn.3979. URL http://www.nature.com/neuro/journal/v18/n5/abs/nn.3979.html.

Ian H. Stevenson and Konrad P. Kording. How advances in neural recording affect data analysis. In *Nature Neuroscience*, volume 14, pages 139–142, 2011. ISBN 1097-6256. doi: 10.1038/nn.2731. URL http://www.nature.com/neuro/journal/v14/n2/abs/nn.2731.html.

Florian Stimberg, Andreas Ruttor, and Manfred Opper. Bayesian Inference for Change Points in Dynamical Systems with Reusable Statesa Chinese Restaurant Process Approach. *Artificial Intelligence and Statistics*, 22(1):1117–1124, 2012. ISSN 1938-7228. URL http://proceedings.mlr.press/v22/stimberg12/stimberg12.pdf.

Yee W. Teh. Dirichlet Process. In *Encyclopedia of Machine Learning*, pages 280–287. Springer US, Boston, MA, 2011. doi: 10.1007/978-0-387-30164-8_219. URL http://www.springerlink.com/index/10.1007/978-0-387-30164-8_219.

M Tsodyks, T Kenet, A Grinvald, and A Arieli. Linking spontaneous activity of single cortical neurons and the underlying functional architecture. *Science*, 286(5446):1943–1946, 1999. URL http://science.sciencemag.org/content/286/5446/1943.short.

Joanna Tyrcha and Matteo Marsili. The Effect of Nonstationarity on Models Inferred from Neural Data. *Journal of Statistical Mechanics: Theory and Experiment*, 2013(03): P03005, 2013. URL http://iopscience.iop.org/article/10.1088/1742-5468/2013/03/P03005/meta.

B. M. Yu, J. P. Cunningham, G. Santhanam, S. I. Ryu, K. V. Shenoy, and M. Sahani. Gaussian-Process Factor Analysis for Low-Dimensional Single-Trial Analysis of Neural Population Activity. In *Journal of Neurophysiology*, volume 102, pages 614–635, 2009. ISBN 0022-3077. doi: 10.1152/jn.90941.2008. URL http://jn.physiology.org/cgi/doi/10.1152/jn.90941.2008.

Hong-Li Zeng, Mikko Alava, Erik Aurell, John Hertz, and Yasser Roudi. Maximum likelihood reconstruction for Ising models with asynchronous updates. *Physical Review Letters*, 110(21):210601, 2013.

Yuan Zhao and Il Memming Park. Variational latent gaussian process for recovering single-trial dynamics from population spike trains. *Neural Computation*, 29(5): 1293–1316, may 2017. ISSN 1530888X. doi: 10.1162/NECO_a_00953. URL http://www.mitpressjournals.org/doi/10.1162/NECO_a_00953.

# Chapter 9

# Unpublished article: *Inferring the collective dynamics of neuronal populations from single-trial spike trains using mechanistic models*

Authors:
Christian Donner[1,2], Manfred Opper[1,2], Josef Ladenbauer[2,3]
[1]Technische Universität Berlin. [2]Bernstein Center for Computational Neuroscience Berlin. [3]École Normale Supérieure, Paris.

**Chapter 9** This chapter comprises the unpublished manuscript, which is authored by myself (CD), Prof. Manfred Opper (MO), and Dr. Josef Ladenbauer (JL).

Contributions:
CD and JL conceived and designed the work with help of MO. CD derived the inference algorithms and developed the Python code. CD performed the numerical experiments. CD wrote the manuscript with substantial contribution from JL.

Python code on GitHub: https://github.com/christiando/doubly_stochastic_lif_inference.git

# Inferring the collective dynamics of neuronal populations from single-trial spike trains using mechanistic models

**Christian Donner** christian.donner@bccn-berlin.de
**Manfred Opper** manfred.opper@tu-berlin.de
*Artificial Intelligence Group*
*Technische Universität Berlin*
*Berlin, Germany*
**Josef Ladenbauer** josef.ladenbauer@gmail.com
*Laboratoire de Neurosciences Cognitives et Computationnelles*
*École Normale Supérieure - PSL Research University*
*Paris, France*

## Abstract

Multi-neuronal spike-train data recorded in–vivo often exhibit rich dynamics as well as considerable variability across cells and repetitions of identical experimental conditions (trials). Efforts to characterise and predict the population dynamics and the contributions of individual neurons require model-based tools. Abstract statistical models allow for principled parameter estimation and model selection, but possess only limited interpretive power because they typically do not incorporate prior biophysical constraints. Here we present a statistically principled approach based on a population of doubly-stochastic integrate-and-fire neurons, taking into account basic biophysics. This model class comprises an idealised description for the dynamics of the neuronal membrane voltage in response to fast independent and slower shared input fluctuations. To efficiently estimate the model parameters and compare different model variants we compute the likelihood of observed single-trail spike trains by leveraging analytical methods for spiking neuron models combined with inference techniques for hidden Markov models. This allows us to reconstruct the shared input variations, classify their dynamics, obtain precise spike rate estimates, and quantify how individual neurons couple to the low-dimensional overall population dynamics, all from a single trial. Extensive evaluations based on simulated data show that our method correctly identifies the dynamics of the shared input process and accurately estimates the model parameters. Validations on ground truth recordings of neurons in–vitro demonstrate that our approach successfully reconstructs the dynamics of hidden inputs and yields improved fits compared to a typical phenomenological model. Finally, we apply the method to a neuronal population recorded in–vivo, for which we assess the contributions of individual neurons to the overall spiking dynamics. Altogether, our work provides statistical inference tools for a class of reasonably constrained, mechanistic models and demonstrates the benefits of this approach to analyse measured spike train data.

**Keywords:** non–stationary spiking activity, leaky integrate and fire neuron, hidden Markov model, Ornstein–Uhlenbeck process, Markov jump process

## 1. Introduction

Cortical computations are represented in the collective spiking activity of multiple neurons. The growing interest to uncover how these neural populations process and transform (complex) incoming information into decisions and motor actions has brought about cell-resolving activity measurements at an increasing pace and scale. Much information appears to be contained in population dynamics of low dimensionality (Shadlen and Newsome, 1998; Luczak et al., 2009; Mante et al., 2013; Aljadeff et al., 2016). To interpret the measured spike trains statistical methods based on generative models can be very powerful, providing also the opportunity to make predictions for unobserved conditions. Often parametric phenomenological models are fitted to these data and used for analysis. Typical examples include models assuming Poisson data (Brillinger, 1988; Latimer et al., 2015; Macke et al., 2015), Ising and generalised linear models (Gaudino et al., 2014; Shimazaki et al., 2012; Chichilnisky, 2001; Truccolo, 2004; Pillow et al., 2008; Aljadeff et al., 2016). Furthermore, explicit dimension reduction methods that extract a low dimensional trajectory of observed spiking data, were proposed (Cunningham and Yu, 2014; Pandarinath et al., 2018). Models of that kind are a flexible model class, can be efficiently fit, and are well suited to capture statistical properties of the data. A drawback of these models is, however, that their parameters do not directly relate to the underlying biophysics, which limits their interpretive power.

Mechanistically more detailed models describe the dynamics of the neuronal membrane voltage, which is typically not observed. The most classical, and probably best known model is of the Hodgkin–Huxley type (Hodgkin and Huxley, 1952), which contains numerous parameters and a set of partial differential equations, that make fitting already challenging knowing the membrane voltage trace (Lueckmann et al., 2017; Meliza et al., 2014). An alternative, prominent class of idealised models are of the *integrate-and-fire* (I&F) type. These models have become state-of-the-art for describing neural activity in in–vivo like conditions (i.e. not knowing the internal voltage dynamics)(Jolivet et al., 2008; Badel et al., 2008; Gerstner and Naud, 2009; Harrison et al., 2015; Pozzorini et al., 2015; Teeter et al., 2018) and have been applied in a multitude of studies on neural network dynamics. While biophysically more faithful than purely phenomenological models, these spiking neuron models are also more complex to fit, particularly in the presence of unknown, noisy and correlated inputs, having only access to the spike times as typical for in–vivo data.

To account for the collective stochastic spiking dynamics observed in–vivo we consider a population of doubly-stochastic I&F neurons. Their hidden inputs contain fast independent fluctuations that give rise to spiking variability across neurons and trials, and slower shared variations due to common drive, which dominate the low-dimensional overall population dynamics. We develop a statistically principled approach to fit this type of model to single-trial spike trains and allow for a quantitative comparison between different model variants, including simpler phenomenological models, according to established criteria. Specifically, we efficiently compute the likelihood of a given spike train by exploiting analytical methods for spiking neuron models (Ladenbauer et al., 2018) combined with inference techniques for hidden Markov models (Rabiner, 1989). Based on simulated data we then evaluate this approach extensively in terms of reconstruction of the true dynamics, classification of their type (continuous and jumpy input dynamics), and estimation of the model parameters.

We next validate the method using in-vitro ground truth recordings of cortical neurons stimulated by current signals with additive noise (Pozzorini et al., 2013). We successfully reconstruct the signal dynamics and demonstrate improved fitting performance compared to a typical Poisson model. Finally, we apply our approach to extracellular recordings from macaque primary visual cortex in vivo (Kohn and Smith, 2016). We demonstrate how model selection and parameter inference allow us to characterise the overall population dynamics and the mechanistic contributions of individual neurons (Okun et al., 2015) systematically.

## 2. Results

Our results are structured as follows. We explain the modelling/inference setup in section 2.1. In the following two sections we outline our inference methods and evaluate them exhaustively based on simulated (ground truth) data. For reasons of clarity and comprehensibility we focus on single neurons in section 2.2 and consider neuronal populations in section 2.3. In section 2.4 we validate our approach based on single cell recordings in–vitro. In section 2.5 we apply our methods to population spike train data from extracellular recordings in-vivo. The python code with examples is publicly available at (Donner, 2018).

### 2.1 Statistical modelling with I&F neurons

We consider having observed cell-resolved spike trains (ordered sets of spike times) from a population of $N$ neurons. Such data are obtained, for example, from extracellular multi-electrode recordings upon pre-processing that includes spike sorting (Einevoll et al., 2012). Each neuron (with index $i$) of this population shall be described by the classical leaky I&F model (Brunel and Van Rossum, 2007) with membrane time constant $\tau_{\mathrm{m}}$, receiving a compound synaptic input given by a Gaussian white noise process with mean

$$\mu_i(t) = C_i x(t) + \bar{\mu}_i \tag{1}$$

and standard deviation $\sigma_i$. This process effectively models synaptic bombardment impinging upon the neuron's membrane voltage (see, e.g., (Brunel and Van Rossum, 2007; Mattia and Del Giudice, 2002; Renart et al.; Kim and Shinomoto, 2012; Augustin et al., 2017)). The input includes shared variations $x(t)$ caused by a common, time-varying drive, and independent, rapid fluctuations with strength $\sigma_i$. $\bar{\mu}_i$ is a cell-specific offset. The common input component affects each neuron to a (possibly) different extent via "coupling" strengths $C_i$, and may be thought of a signal that carries information.

Since the dynamics of the common input are not known we describe $x(t)$ by a stochastic process. In particular, we consider two qualitatively different Markov models, an *Ornstein-Uhlenbeck process* (OUP) and a *Markov jump process* (MJP), which gives rise to two model variants. For the OUP $x(t)$ varies continuously with time constant $\tau$, whereas for the MJP $x(t)$ is piece-wise constant intermittently jumping to different values with rate $\tau^{-1}$. The stationary probability density of both processes is standard normal and their autocorrelation functions are identical. Our mechanistic model thus incorporates reasonable biological constraints and features rich dynamics in a minimal way without overwhelming complexity (in view of the available data). For further details on the generative model see section 4.1. In sections 2.4 and 2.5 we further consider a typical, simpler model, that describes the spike train of neuron $i$ by a Poisson process with rate $\exp(\mu_i(t))$, where $\mu_i(t)$ is given by Eq (1).

We would like to point out that not all model parameters need to be estimated. The membrane voltage can be scaled such that the remaining parameters of interest for estimation are those for the input together with the membrane time constant (Ladenbauer et al., 2018). Here we focus on inferring the input dynamics, because it predominantly determines the overall dynamics of the population (Ostojic and Brunel, 2011; Augustin et al., 2017).

### 2.2 Inference for single neurons

**Outline of inference approach** It is instructive to first consider a single neuron. For improved readability we omit the neuron index $i$ in this section; the parameters of interest for inference are thus $\vartheta := \{C, \bar{\mu}, \sigma, \tau_\mathrm{m}\}$ and $\tau$. We collect the measured spike train data in the increasing sequence of spike times $t_{0:K} := (t_0, \ldots, t_K)$ and define the $k$-th interspike interval (ISI) by $s_k := t_k - t_{k-1}$ ($K$ is the number of observed ISIs). For notational ease below we use that $s_k$, the duration of the $k$-th ISI, implicitly contains information about the start time and end time of the interval. As spike emission in the leaky I&F model is a renewal process the likelihood/probability of observing a given spike train from the model is completely determined by the likelihoods/probabilities of observing the constituent ISIs. Justified by the assumption that the process $x(t)$ is rather slow — i.e., $\tau$ is large compared to the ISIs — it is approximated for each ISI by one value, $x(t) = x_k$ for ISI $s_k$. In this way we obtain a hidden Markov model with latent states $x_{0:K} := (x_0, \ldots, x_K)$, where $x_0$ is the value of the process at $t = 0$ and $x_k$ corresponds to time $t = t_k$ for $k \geq 1$. The observations are contained in the sequence $s_{1:K} := (s_1, \ldots, s_K)$. The joint likelihood of observing a given spike train and realisation of $x(t)$, approximated by sequence $x_{1:K}$, is given by

$$p(s_{1:K}, x_{0:K}|\vartheta, \tau) = \prod_{k=1}^{K} p(s_k|\mu_k, \vartheta) p(x_k|x_{k-1}, \tau) p(x_0), \tag{2}$$

with effective mean input $\mu_k = Cx_k + \bar{\mu}$. $p(s_k|\mu_k, \vartheta)$ is the ISI probability density of a leaky I&F neuron subject to Gaussian white noise input, which can be accurately evaluated by solving a Fokker-Planck partial differential equation (Ostojic and Brunel, 2011). This solution can be obtained numerically in efficient ways (Ladenbauer et al., 2018) (see Methods). $p(x_k|x_{k-1}, \tau)$ is the transition probability density for the hidden process, from state $x_{k-1}$ to state $x_k$, which depends on the time constant $\tau$ and on the time duration between states $x_{k-1}$ and $x_k$. To keep notation simple the latter duration is not indicated explicitly. The transition density can be explicitly expressed for either model variant (OUP or MJP). $p(x_0)$ is the prior distribution of the process $x(t)$ at $t = 0 < t_0$ (prior to $t_0$), for which we assume its stationary distribution, a standard normal (see Methods section 1). Note that in Eq (2) we have used the renewal property of leaky I&F neurons. The likelihood of $s_{1:K}$ is obtained from Eq (2) by marginalising with respect to $x_{0:K}$,

$$p(s_{1:K}|\vartheta, \tau) = \int p(s_{1:K}, x_{0:K}|\vartheta, \tau) dx_{0:K},$$

which can be efficiently achieved by forward filtering (see Methods).

The ability to evaluate the likelihood $p(s_{1:K}|\vartheta, \tau)$ with high accuracy and low computational effort allows us to locate its maximum with respect to the parameters of interest,

which yields the parameter estimates. For maximisation, we use an established simplex-based method (Nelder and Mead, 1965) (see section 4.3).

We then compare the fitting performance of either model variant (OUP and MJP) using the maximised likelihood values. In this way we identify the best model and classify the dynamics of the slow input variations. Specifically, we use the log-likelihood ratio (LLR) for comparison, equivalently expressed as the difference of the logarithm of maximal likelihoods,

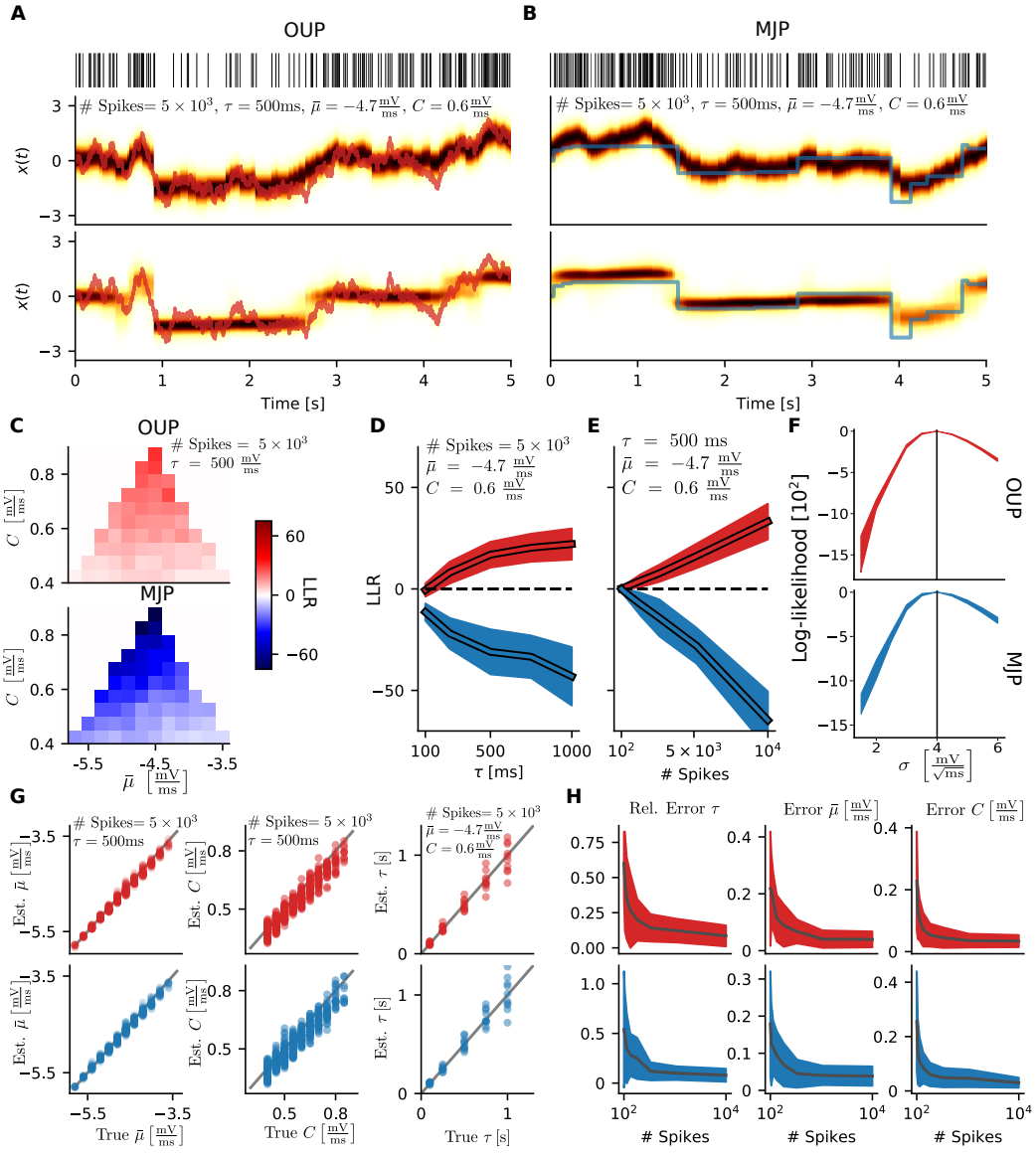
$$\mathrm{LLR}(\mathrm{OUP},\mathrm{MJP}) := \log \max_{\vartheta,\tau} p_{\mathrm{OUP}}(s_{1:K}|\vartheta,\tau) - \log \max_{\vartheta,\tau} p_{\mathrm{MJP}}(s_{1:K}|\vartheta,\tau),$$

where subscripts OUP and MJP indicate the respective process. Positive values of the LLR indicate that the OUP model variant is favoured, while negative ones indicate that the MJP model variant better describes the data.

Finally, in order to reconstruct the time series of the process $x(t)$ and estimate the spike rate time series of the neuron we infer $p(x_k|s_{1:K},\vartheta,\tau)$, the probability density over $x_k$ given $s_{1:K}$ and the parameter values, for each $k$. This density is computed through backward smoothing in addition to the above-mentioned forward filtering (which completes the established forward-backward algorithm for hidden Markov models (Rabiner, 1989), see section 4.5). The reconstructed time series of $x(t)$ is then given by the sequence of expected values $(\langle x_0\rangle,\ldots,\langle x_K\rangle)$ calculated using these densities. Additionally, the spike rate time series, can be obtained by the sequence $(r_0,\ldots,r_K)$ of expected inverse mean ISIs, where the mean ISI for the $k$-th interval is calculated using $p(s|\mu_k,\vartheta)$ and the expectation is with respect to $p(x_k|s_{1:K},\vartheta,\tau)$ (see section 4.5). Note that $r_k$ represents the instantaneous spike rate estimate at the time that corresponds to $x_k$.

We compare the doubly-stochastic I&F model to a Poisson process whose principal parameter, the (non-negative) rate, is given by $\lambda(t)=\exp(\mu(t))$, with $\mu(t)=Cx(t)+\bar{\mu}$ (cf. Eq (1)) where $x(t)$ is an OUP or MJP as described above. We would like to remark that the exponential function for the mapping between input and (Poisson) output rate is a common choice, see e.g. (Pillow et al., 2008; Macke et al., 2015; Pandarinath et al., 2018). Since the ISIs according to a (stationary) Poisson process with rate $\lambda$ are distributed according to a exponential distribution with mean $\lambda^{-1}$, we calculate the corresponding distribution analytically and follow the methodology described above. For the Poisson model we use an inference method that is analogous to the one described above for I&F neurons with only two differences: the ISI density is explicitly expressed by $p(s_k|\mu_k,\vartheta)=\lambda_k^{-1}\exp(-s_k\lambda_k)$ with $\lambda_k=\exp(\mu_k)$, $\mu_k=Cx_k+\bar{\mu}$, and $\vartheta$ only contains $C$ and $\bar{\mu}$.

**Evaluation on simulated data from generative model** To test the proposed method we simulate data according to the generative model and maximise the logarithm of the likelihood with respect to the model parameters $\vartheta$. If not stated otherwise, model parameters for generating the data are fixed to $\bar{\mu}=-4.7\ \frac{\mathrm{mV}}{\mathrm{ms}}$, $C=0.6\ \frac{\mathrm{mV}}{\mathrm{ms}}$, $\tau_m=10$ ms, $\sigma=4\ \frac{\mathrm{mV}}{\sqrt{\mathrm{ms}}}$, $\tau=500$ ms, and the data contain $5\times10^3$ spikes. Throughout this work, we consider the membrane time constant to be fixed to a biophysical plausible value $\tau_\mathrm{m}=10$ ms. Changing $\tau_\mathrm{m}$ hardly influences the likelihood and can be compensated by optimising the white noise amplitude $\sigma$ (see appendix and Ladenbauer et al. (2018)). Exemplary data together with the inferred posterior density of the common input process $x(t)$ are shown in Fig 1 **A** and **B**. The slow common input process $x(t)$ is generated as an OUP and MJP (solid lines), respectively. The inferred posterior recovers the shared process well from the spike data.


A
OUP
B
MJP
# Spikes= 5 × 10^3, τ = 500ms, μ̄ = −4.7 mV/ms, C = 0.6 mV/ms
x(t)
Time [s]
C
OUP
# Spikes = 5 × 10^3
τ = 500 mV/ms
MJP
C [mV/ms]
μ̄ [mV/ms]
LLR
D
# Spikes = 5 × 10^3
μ̄ = −4.7 mV/ms
C = 0.6 mV/ms
LLR
τ [ms]
E
τ = 500 ms
μ̄ = −4.7 mV/ms
C = 0.6 mV/ms
LLR
# Spikes
F
Log-likelihood [10^2]
OUP
MJP
σ [mV/√ms]
G
# Spikes= 5 × 10^3
τ = 500ms
Est. μ̄ [mV/ms]
Est. C [mV/ms]
Est. τ [s]
True μ̄ [mV/ms]
True C [mV/ms]
True τ [s]
H
Rel. Error τ
Error μ̄ [mV/ms]
Error C [mV/ms]
# Spikes

Note, that only a small fraction of the whole data is visualised here. For both examples, we also infer the model with the wrong process prior and see that in those cases the basic dynamics is recovered, but clearly less accurate as compared to fits with the correct prior dynamics.

Is the less accurate recovery of the process $x(t)$ reflected in the obtained likelihood values? To examine this systematically we simulate several datasets as in Fig 1 **A** and **B**, but with varying values of the parameters $\bar{\mu}$, $C$ and $x(t)$ either being generated as OUP (Fig 1 **C**, top) or MJP (bottom). We maximise the likelihood for each dataset under both process assumptions and compute the LLR(OUP, MJP), which should be positive for data generated with an OUP process and negative for MJP data, as is indeed the case (Fig 1 **C**). The colouring of each triangle entry depicts the average LLR of 10 spike train simulations with same $\bar{\mu}, C$ values. We consider only $\bar{\mu}, C$-pairs, for which the neuron's expected firing rate is $>$ 1Hz and $<$ 110Hz for at least 98% of the simulation time. This can be determined by investigating the quantiles of the stationary distribution of $\mu(t)$, which is $\mathcal{N}(\bar{\mu}, C^2)$. The mean firing rate is a deterministic function of $\mu(t)$ and $\sigma$ (Richardson, 2008).

---

Figure 1 *(facing page)*: **Fitting single neuron model.** If parameters are not explicitly varied their values are $\bar{\mu} = -4.7\frac{\text{mV}}{\text{ms}}, C = 0.6\frac{\text{mV}}{\text{ms}}, \tau = 500\text{ms}, \sigma = 4\frac{\text{mV}}{\sqrt{\text{ms}}}$ and the number of observed spikes is $5 \times 10^3$. Results shown in red are fits to data generated with $x(t)$ being an OUP and blue for an MJP prior. **A**: Example trace with an OUP as true slow common input process $x(t)$ (red trace). Top: Observed spike data. Centre: Fit with OUP prior for $x(t)$. Bottom: Fit with MJP prior. Colour map depicts the inferred posterior density oover process $x_k$. **B**: Same as **A** where true $x(t)$ is a MJP realisation (blue trace). **C**: LLR for data generated with different model parameters $\bar{\mu}, C$ (see text). Top: Data are generated with $x(t)$ being OUP. Bottom: True $x(t)$ is a MJP. The colouring of each entry in the pyramid is the average over fits to 10 datasets. **D**: LLR for several data examples as a function of the time constant $\tau$. Areas denote mean $\pm$ std. obtained from 10 fits to different datasets. **E**: LLR as a function of number of observed spikes. **F**: Maximal likelihood (shifted such that the maximum value is at 0) obtained by fitting the model with different values of white noise parameter $\sigma$. **G**: True versus estimated model parameters. Left and centre: $\bar{\mu}$ and $C$, respectively. All simulations from **C** are combined. Right: The time constant parameters $\tau$, where the other parameters $\bar{\mu}, C$ are fixed (data from **D**). **H** Left: The relative error of the time constant parameters $\tau$ as a function of number of observed spikes used for fitting. Centre and right: The absolute error of $\bar{\mu}, C$ as a function of number of observed spikes, respectively.

In Fig 1**C** we observe, that the LLR favours the correct dynamics increasingly for stronger couplings $C$. Increasing the offset $\bar{\mu}$ has only a minor effect. In Fig 1**D**, we investigate how the process time constant $\tau$ affects the LLR. Slower processes (i.e. higher values of $\tau$) allow for better discrimination between the OUP and MJP prior. For even larger $\tau$ we expect again a decline in separability of the two processes, because the same stationary process is approached for $\tau \to \infty$. We observe a slight bias towards the MJP process, in particular for small values of $\tau$. The reason might be caused our assumption of a stationary process between two consecutive spikes ($x(t) \approx x(t_k)$ for $t \in [t_{k-1}, t_k)$). This piecewise constant assumption is closer to the MJP prior than the OUP. The LLR increases as more spikes are observed (Fig 1**E**). A few hundred spikes are sufficient to identify the correct dynamics.

After having established that the process $x(t)$ and the correct dynamics can be recovered, we investigate whether the parameters of the model are recovered accurately (Fig 1**F**–**H**). In Fig 1**F** the likelihood (average ± standard deviation) for 10 different datasets is shown as a function of the white noise amplitude $\sigma$. The likelihood is strongly influenced by this parameter and is maximal for the correct white noise std., $\sigma = 4\ \frac{\mathrm{mV}}{\sqrt{\mathrm{ms}}}$. In Fig 1**H** the true parameters from Fig 1**C** and **D** are plotted against the estimated ones (Top with the OUP as generating process and bottom with the MJP). The offset $\bar{\mu}$ are recovered with high precision. The couplings are also well captured, even tough with higher variance and a small bias towards values smaller than the true ones. The time constant $\tau$ is recovered well with higher precision for faster processes. With increasing data length the estimation error decreases (Fig 1 **H**, mean ± standard deviation for data from **E**). However, for $\sim 10^3$ spikes and more, the error saturates and does not approach 0. This might be because of the discretisation of process $x(t)$, which limits the precision of the parameter estimate that can be obtained. The error is defined as the absolute difference between true and estimated parameters. The relative error is the absolute error divided by the true value.

### 2.3 Inference for neuronal populations

**Outline of inference approach** We now move to the full population as described in section 2.1. We collect the observed (cell-resolved) ISIs of all $N$ neurons in the sequence $s_{1:K} := (s_1, \ldots, s_K)$ which is ordered according to the last spike time of each ISI. Similarly as in section 2.2 (single neuron) we effectively approximate $x(t)$ for each neuron and each ISI by one value, which is justified by the slowness of the process. Specifically, it is approximated by the sequence $x_{0:K} := (x_0, \ldots, x_K)$ where $x_k$ is the value of the process at the last spike time of ISI $s_k$ for $k \geq 1$ and $x_0$ corresponds to time $t = 0$. Note that the sequence can contain multiple values within each ISI. This approximation allows us to extend straightforwardly the approach from the previous section to a neuronal population. We summarise neuron-specific parameters as $\vartheta_i := \{C_i, \bar{\mu}_i, \sigma_i, \tau_\mathrm{m}\}$ and $\vartheta := \{\vartheta_1, \ldots, \vartheta_N\}$. The joint likelihood of the data and a realisation of process $x(t)$ can then be expressed as

$$p(s_{1:K}, x_{0:K}|\vartheta, \tau) = \prod_{k=1}^{K} p(s_k|\mu_{k,i_k}, \vartheta_{i_k}) p(x_k|x_{k-1}, \tau) p(x_0), \tag{3}$$

with effective mean input $\mu_{k,i_k} = C_{i_k} x_k + \bar{\mu}_{i_k}$, where $i_k$ indicates the neuron corresponding to the $k$-th ISI $s_k$ in the sequence $s_{1:K}$. Note that for the transition probability density

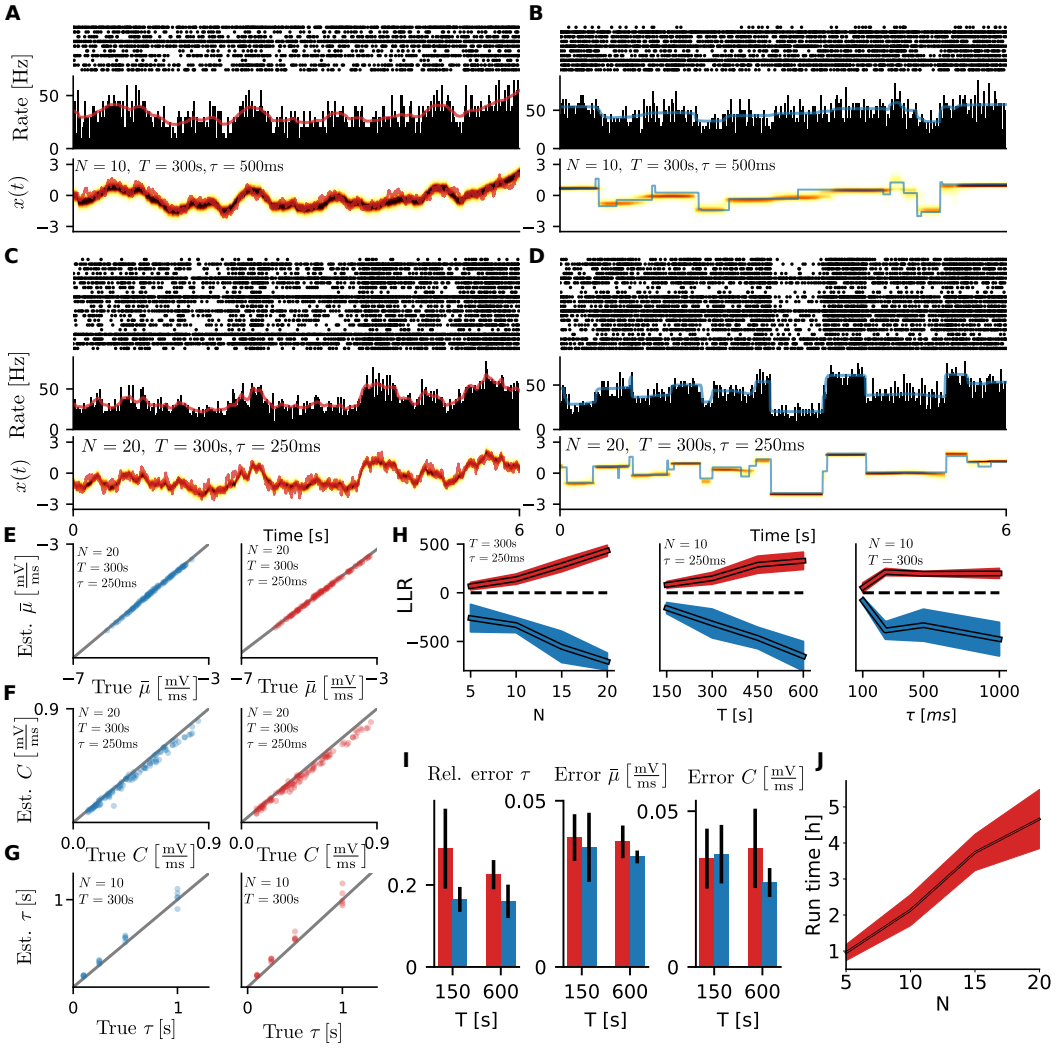
$p(x_k|x_{k-1}, \tau)$ the duration between consecutive spikes (except for the first spike of each neuron) are used. Analogously to the single neuron case, we marginalise with respect to $x_{0:K}$ in Eq. (3) to obtain the likelihood of having observed the data from the model, and proceed as in section 2.2 in order to estimate the model parameters, classify the dynamics of $x(t)$ (OUP versus MJP), reconstruct its time series, and extract a precise estimate of the time-varying spike rate (see sections 4.3–4.5). However, even though the (marginalised) likelihood can be accurately and rapidly evaluated using numerical methods (cf. sections 4.2 and 4.3), maximisation with respect to $> 3N$ parameters becomes a serious challenge for a large population. Therefore, we propose an efficient, approximate technique for optimisation (see section 4.3).

**Evaluation on simulated data from generative model** As for the single neuron case, we validate the population method on data from the generative model (Fig 2). Parameters for generating the data are fixed to $\sigma_i = 4 \frac{\text{mV}}{\sqrt{\text{ms}}}$ $\tau_m = 10$ ms, $\tau = 250$ ms and recording length $T = 300$ s, if not indicated otherwise. The $(\bar{\mu}_i, C_i)$- pairs are drawn uniformly from the parameter space defined for Fig 1**C**. In Fig 2**A**–**B** we see two different exemplary fits for a population of $N = 10$ neurons and process time constant $\tau = 500$ ms, where data are generated with $x(t)$ being an OUP and a MJP, respectively. The inferred process $x(t)$, shown on the bottom, fits the ground truth well. Once the model is inferred we can obtain the expected population firing rate (see Methods for details). The predicted population rate matches well with the empirical rate in the data (Fig 2**A** and **B**, centre). In Fig 2**C**–**D** examples for faster processes and larger population ($\tau = 250$ ms, $N = 20$) are displayed. As before, the fitting results for the process are accurate.

To investigate whether the model parameters can be recovered accurately we compare the estimated offsets $\bar{\mu}_{1:N}$ obtained from 5 different data sets with $N = 20$ (Fig 2**E**). The retrieved estimates are close to the true values. For the couplings $C_{1:N}$ (Fig 2**F**) we observe a small bias, i.e. the estimated values tend to be marginally smaller than the true values. Again, this might be an artefact of the heuristic we use for fitting (see Methods). We obtain good fits for the time constant $\tau$, even if the estimates for slower processes are more variable (Fig 2**G**). This is expected, because the dynamics becomes less prominent for higher values of $\tau$. To examine the quality of parameter estimation with respect to the amount of observed data we plot the mean (relative) error of all 3 parameter sets for fits obtained with $T = 150$ s and $T = 600$ s ($N = 10$, Fig2**I**). The relative error for time constant $\tau$ decreases marginally while the error for $\bar{\mu}_i$ and $\bar{C}_i$ is already marginally small for $T = 150$ s and does not decrease any further. We conclude that already relative short recording times of 150 s = 2.5 min suffice to get accurate fitting results.

In Fig2**H**, we investigate the classification of the correct process, OUP or MJP, in terms of LLR as function of population size $N$, recording time $T$, and process time constant $\tau$. The more neurons are observed, or the longer the recording time $T$, the better we can differentiate between the two process. Note, that the magnitude of the y-axis is one order of magnitude larger than for the single neuron case in Fig 1**D**. For the process time constant $\tau$, classification gets better for increasing values, establishing that slower processes can be classified easier. However, we see saturation of the LLR for $\tau > 250$ ms.

Fig. 2**J** shows that computing time needed to maximise the likelihood of the model increases linearly with the number of neurons.


A
B
C
D
E
F
G
H
I
J
Rate [Hz]
x(t)
N = 10, T = 300s, τ = 500ms
N = 20, T = 300s, τ = 250ms
Time [s]
Est. μ̄ [mV/ms]
True μ̄ [mV/ms]
Est. C [mV/ms]
True C [mV/ms]
Est. τ [s]
True τ [s]
LLR
N
T [s]
τ [ms]
Rel. error τ
Error μ̄ [mV/ms]
Error C [mV/ms]
Run time [h]

In in–vivo experiments oscillatory signals are often observed. To scrutinise how sensible our fitting method is to such oscillations we investigate data not from the generative model any more, but where the shared input $x(t)$ is a sine wave with different frequencies $(2.5, 5, 10$ Hz) in Fig 3. For population size of $N = 20$ the process $x(t)$ can be recovered for all these frequencies, with degrading accuracy for faster frequencies. Both process priors -OUP and MJP- yield a posterior over $x(t)$ that describes the process and the population firing rate well. For a sine wave with frequency of 20 Hz the model fits result in stationary posterior over process $x(t)$ (not shown). For larger $N$ one might be able to capture frequencies larger than 10 Hz. Note that the priors assumed here do not have any knowledge about the sinusoidal nature of $x(t)$. This demonstrates the flexibility of the observed model, even if more accurate inference results could be obtained knowing the correct prior.

### 2.4 Validation based on in-vitro ground truth recordings

Having established that the inference method works well for single neurons and populations we now consider data from experimental recordings. The first dataset is relatively controlled and hypothetically close to the proposed generative model. The data are in–vitro patch–clamp recordings from pyramidal L5 cells in mouse somatosensory brain slices (see Pozzorini

---

Figure 2 *(facing page)*: **Fitting population model.** If not stated explicitly parameter values are $N = 10, \tau = 250$ ms$, T = 300$s and $\sigma_i = 4\frac{\text{mV}}{\sqrt{\text{ms}}}$. Results depicted in red/blue denote fits to data generated with an OUP/MJP prior, respectively. **A**: 6 exemplary seconds of the inferred posterior over $x(t)$ obtained by fitting the population model to data generated by an OUP as mean input process ($N = 10, \tau = 500$ ms). Top: Spiking data. Centre: Empirical (black histogram) and inferred population firing rate (coloured line). Bottom: Inferred distribution over the common input process $x(t)$ with ground truth (red solid line). **B**: Same as in **A** with $x(t)$ being an MJP. **C** and **D**: Same as **A** and **B**, respectively, with $N = 20$, $\tau = 250$ ms. **E**: True vs. estimated parameters for the offsets $\bar{\mu}_{1:N}$. The plot summarises fits to 5 different datasets generated with $N = 20$. **F** Same as **E** for couplings $C_{1:N}$. **G** Fitting result for time constant $\tau$, obtained by 5 different simulations for each time constant value $\tau \in \{100$ ms$, 250$ ms$, 500$ ms$, 1000$ ms$\}$. **H**: From left to right: the LLR as a function of number of neurons $N$, recording length $T$, and time constant $\tau$. Solid line indicates the mean and shaded area the standard deviation obtained from 5 generated datasets for each parameter value. **I**: From left to right: relative error of time constant $\tau$, error of offsets $\bar{\mu}_{1:N}$, and error of couplings $C_{1:N}$ for $T = 150$ s and $T = 600$ s. Bar height indicates mean and errorbars $\pm$ standard deviation of 5 simulation results. **J**: CPU time required for maximising the likelihood as function of number of neurons $N$.

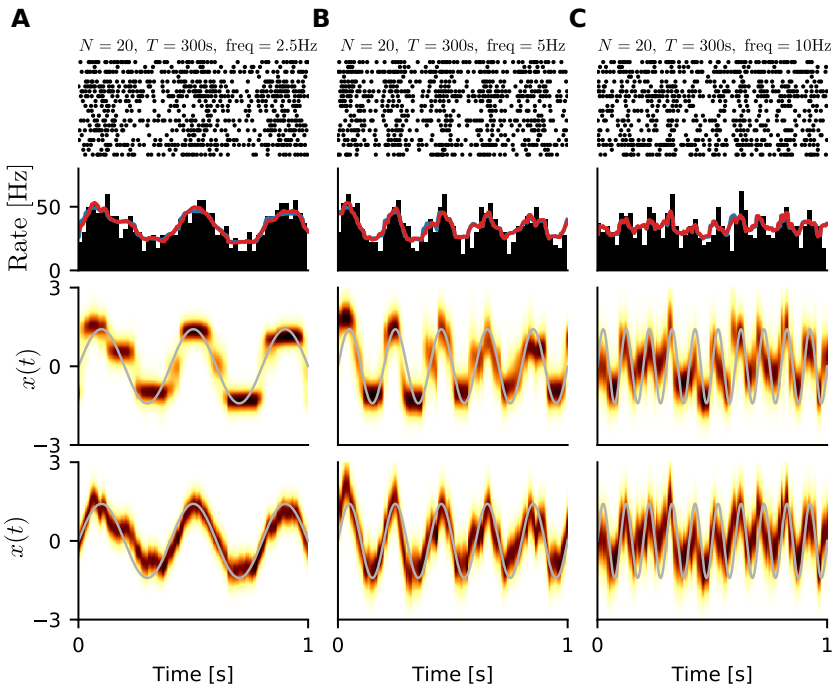
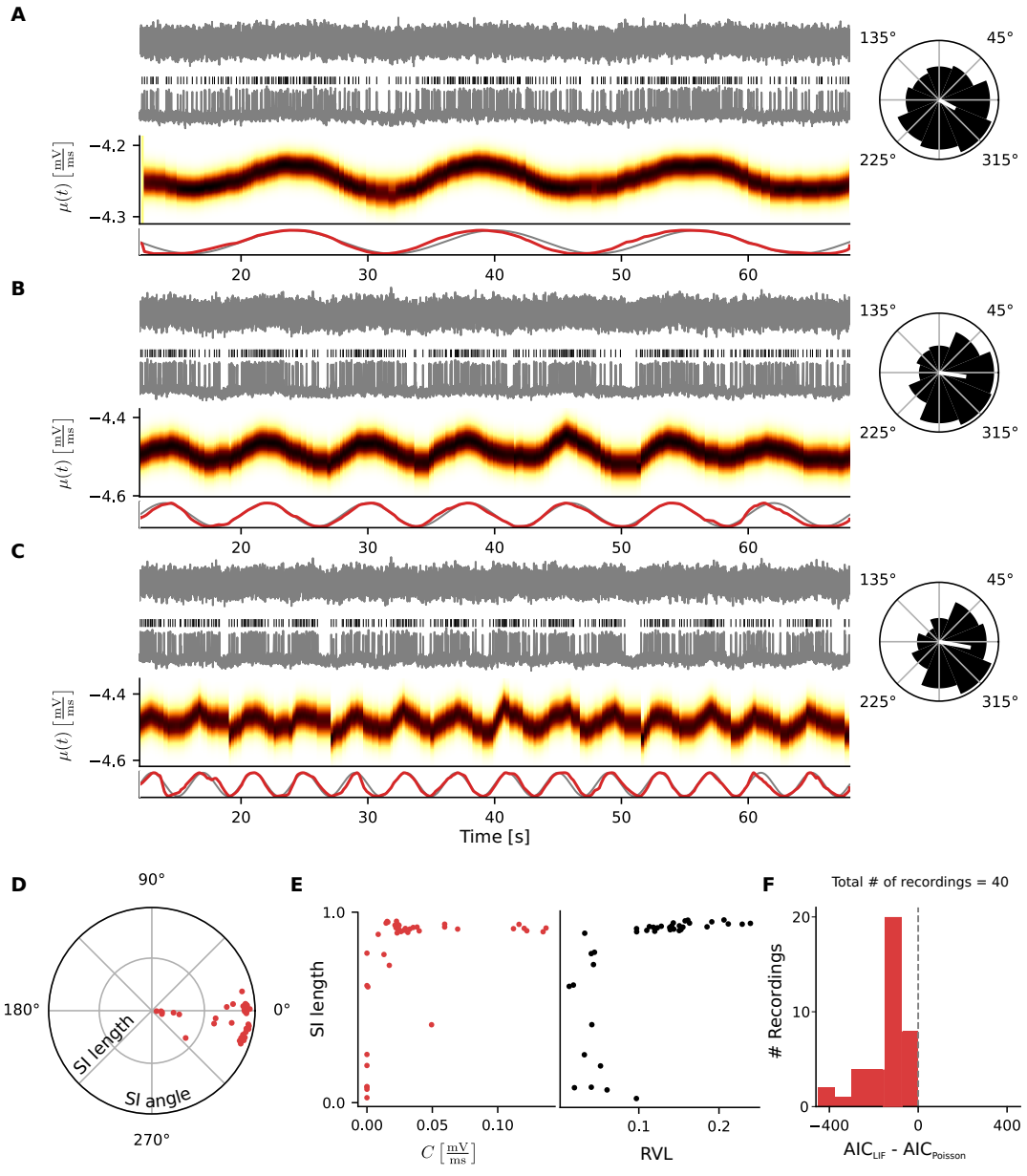



Figure 3: **Fitting population activity driven by oscillating common input. A** Fit to $T = 300$ s simulated data, where $N = 20$ neurons are driven by a sinusoidal input $x(t)$ with a frequency of 2.5 Hz. From top to bottom: 1 s of exemplary spike data, population rate histogram with population rate predictions by model fits with OUP and MJP (red and blue line), posterior over common input process $x(t)$ assuming MJP, and below assuming an OUP. Grey line depicts true $x(t)$. **B** and **C** same as in **A**, but with $x(t)$ being a sine wave with frequency of 5 and 10 Hz, respectively.


A
B
C
135°
45°
225°
315°
$\mu(t)\left[\frac{\text{mV}}{\text{ms}}\right]$
−4.2
−4.3
−4.4
−4.6
20
30
40
50
60
Time [s]
D
90°
180°
0°
270°
SI length
SI angle
E
SI length
1.0
0.0
0.00
0.05
0.10
$C\left[\frac{\text{mV}}{\text{ms}}\right]$
0.1
0.2
RVL
F
Total # of recordings = 40
# Recordings
20
10
0
−400
0
400
$\text{AIC}_{\text{LIF}}$ - $\text{AIC}_{\text{Poisson}}$

et al. (2013) and section 4.6 for details). Excitatory synaptic transmission was blocked during the experiment and the only input the recorded neuron received was a noisy current with sinusoidal mean. The mean input, the spike times, and the voltage trace of the recordings can be seen in Fig 4 **A**–**C**. The mean input current is oscillating with period $\mathcal{T} = \{16\text{ s}, 8\text{ s}, 4\text{ s}\}$, respectively. The amplitude of the oscillations was adjusted such that the neurons' firing rates oscillate between 2 and 6 Hz. Spiking activity is modulated by the mean input oscillation as shown in the circular spike histograms (Fig 4 **A**–**C** right, angle denotes the phase of the mean input current). We fit the model to the recorded spike data under the assumption that the slow input component $x(t)$ is an OUP. The inferred posteriors over input current of the model $\mu(t) = Cx(t) + \bar{\mu}$ exhibit oscillations that correspond to the ones of the true input (Fig 4 **A**–**C** centre). The inferred sinusoidal signal (red line) match well the true one (grey line) (for details see section 4.6).

The oscillation can not always be extracted. In some cases zero couplings $C$ are obtained, i.e. the slow component $x(t)$ is just inferred to be stationary. Thus, we investigate, how strongly the different cells lock to the mean input current. For this analysis we focus on recordings, where the input oscillation has a period of 16 s. For comparing true and inferred mean input signal of each recording we compute the synchrony index (SI) between these two signals. The SI, a measure of how strongly two signals are synchronised, is a complex number, which takes on an absolute value of 1 if two signals are perfectly synchronised, and

---

Figure 4 *(facing page)*: **Fits to in–vitro recordings A**: Model fit to a recording where the neuron is stimulated with noisy input, having an oscillatory mean input with a period of 16 s. From top to bottom: Input current, spike times (black bars), recorded voltage trace, inferred distribution over input process $\mu(t)$, and the sine function with true (grey line) and inferred phase (red line) as argument. Right: Circular spike histogram, where angle indicates the phase of the mean input current. White line denotes the circular mean. The length is the RVL. **B** and **C**: Same as **A** with phase of the oscillatory input mean being 8 and 4 s, respectively. **D** SI between true and inferred oscillation. One dot depicts the fit of a recording. Angle corresponds to average phase shifts of the two oscillations, and radius corresponds to the measure of synchrony. **E** Left: Inferred coupling values $C$ compared to the quality of signal reconstruction (|SI|). Right: Strength of spike locking to input signal (RVL of the circular mean of the true phase at spike times) versus quality of signal reconstruction (|SI|) for all recordings. **F** Comparing the goodness of fit (difference in AIC) assuming an LIF neuron and a Poisson neuron for each recording. If the LIF model yields the better fit, the AIC difference is negative. For **D**–**F** only recordings are used, where true signal had a period of 16 s. Results are only reported for inference with an OUP prior for the common input process $x(t)$.

0 if they are desynchronised. The SI–angle corresponds to the average phase shift between the compared signals. We use the absolute value |SI| as a measure of synchrony between the true and the reconstructed signal (the mean of the inferred posterior over $x(t)$). See section 4.6 and (Cohen, 2008) for details. The polar plot in Fig 4 **D** shows the angle and length of the SI for each recording. While there is consistently little phase shift for all inferred signals, the strength of synchrony varies from values 0 to 1, i.e. from bad to almost perfect reconstruction of the input signal. Plotting the inferred couplings $C$ against the SI length in Fig 4 **E** (Left) we see that recordings with $C \approx 0$ fail to recover the true oscillations ($|\mathrm{SI}| \approx 0$). For $|\mathrm{SI}| \gtrsim 0.8$ the algorithm yields non–zero couplings. The quality of signal reconstruction |SI| depends on how strong firing of neurons locks to the true oscillation (Fig 4 **E** right). To measure the latter we utilise the resultant vector length (RVL) of the circular mean of the true phase at observed spike times $t_{0:K}$ (see section 4.6 and circular spike histograms in Fig 4 **A**–**C**, where the circular mean is depicted as white line).

For the recorded data we do not know whether the assumption of observing LIF neurons is any better than the common Poisson assumption. To verify this, we compare the Akaike information criterion (AIC, see section 4.6 and Akaike (1974)), obtained by maximising the likelihood with the proposed model and the AIC of same model, but assuming that the neuron's firing is a Poisson process. The AIC punishes the higher complexity of the LIF model and attains the lowest value for the best model fit. In Fig 4**F** we observe that the LIF provides the better fit for all recordings without exception.

### 2.5 Application to in–vivo multi-electrode recordings

The second dataset is an in–vivo recording from monkey V1 area (Kohn and Smith, 2016). The data consist of 5 minutes of spontaneous spiking activity of $N = 20$ single units, while the animal was anaesthetised. See section 4.6 for details about data preprocessing. After fitting the model with the OUP and MJP prior, it turns out that the MJP is slightly favoured (LLR $= -70.5$). Comparing this to values obtained with simulated data (see Fig 2**H**) the obtained LLR is rather moderate indicating that both process assumptions provide comparable description. In the following we will discuss the fit of the winning MJP, if not explicitly stated otherwise. Benchmarked against a model with Poisson neurons, the AIC comes out much smaller for the model assuming an LIF population ($\mathrm{AIC}_{\mathrm{lif}} - \mathrm{AIC}_{\mathrm{poisson}} = 3143, 4$), i.e. the LIF model provides a substantially better fit. In Fig 5**A** 30 s of the spiking data are shown for the 20 single units, ordered from top to bottom with decreasing value of the inferred coupling $C_i$ (Fig 5**A**, right). While the single units on top exhibit structured firing, activity becomes more disordered with decreasing values of inferred coupling $C_i$. This indicates that the model infers a common drive to the population, which is reflected in the spiking activity. The inferred values of couplings $C_{1:N}$ are a good measure for characterising how strongly single units couple to the inferred common drive. In Fig 5**B**, we show the population spike histogram and the inferred population rate under the MJP assumption (blue). Below the inferred common input process $x(t)$ is depicted. The inferred population rate matches well the empirical one. The inferred process posterior corresponds well to the spiking data in Fig 5**A**, where the process attains low values for sparse firing periods and jumps to higher values, when concerted firing is observed. In periods with high firing rates the model slightly underestimates the population rate. Note in the fitting procedure

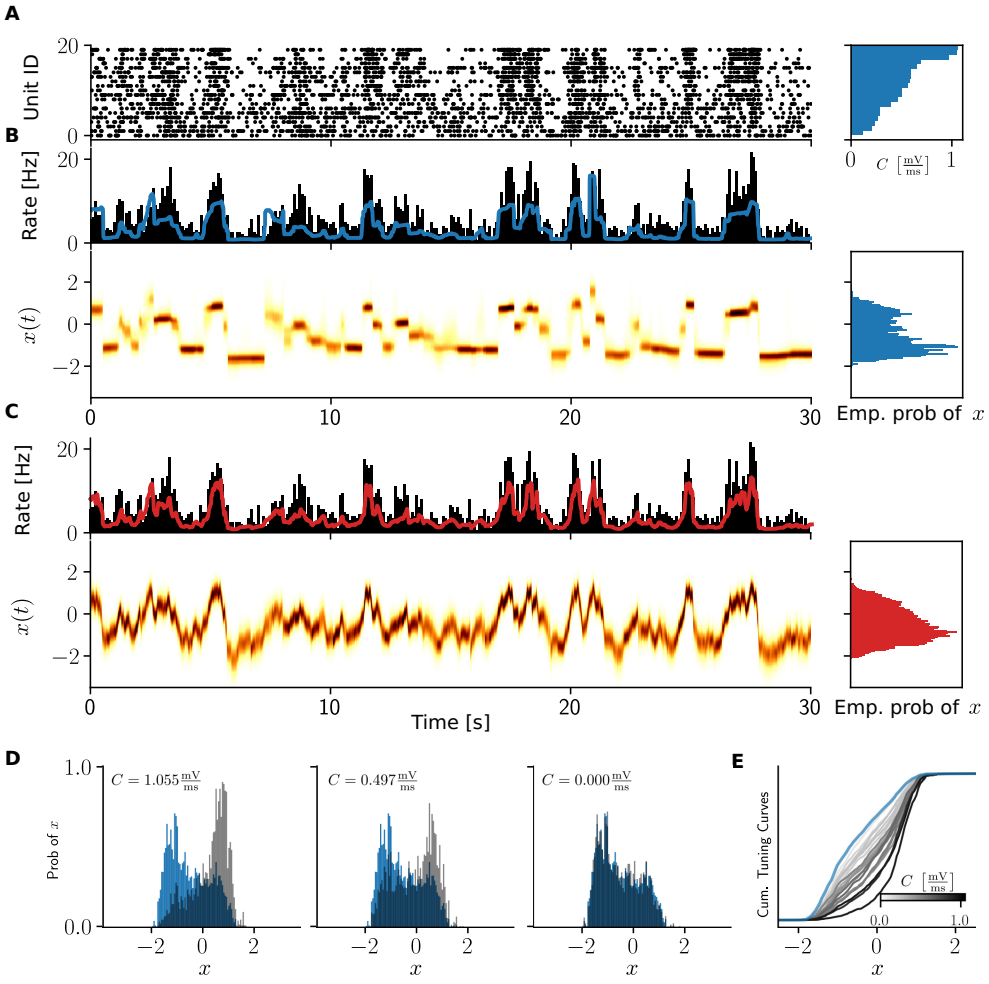



Figure 5: **Fit to in–vivo recorded population of monkey V1 neurons.** **A**: 30 s spike train data of 20 single units, sorted according to the inferred coupling $C_i$ (right). **B**: Empirical (black bars) and inferred population rate (coloured line) with expected rate from MJP obtained from fit to spike data shown in **A**. Below the inferred posterior over the common input process $x(t)$. Right: Time averaged empirical mean process distribution. **C** Same as **B** for fitting the model with the OUP prior. **D** Normalised process tuning curves, i.e. empirical histograms of posterior mean of process $x(t)$ at spike times of neurons (grey) with strong (left), intermediate (centre), and weak coupling $C_i$ (right). The empirical mean process distribution is shown in blue (same as in **B** Right). **E**: The cumulative tuning curves of all 20 single units as shown in **D**. Blue is the cumulative histogram of the whole process as in **C**. The darker the line of the cumulative tuning curve, the stronger the inferred coupling value $C_i$ of the neuron.

we never attempt to reproduce the firing rate of the data explicitly. In Fig 5**B** (right) the empirical distribution of the inferred posterior mean of the process $x(t)$ is shown, averaged over the entire recording time. Subsequently, we call this the 'empirical mean process distribution'. This distribution is quite different from the one that we would expect from the one assumed by the generative model (a standard normal distribution). The process is concentrated at low values, while for higher the distribution is broader. In Fig 5**C**, we show the results obtained with the OUP assumption. The population firing rate is matched better than under the MJP assumption despite the model fit's inferior likelihood.

Lastly, we examine the role of the coupling $C_i$ in more detail. If a single unit is strongly coupled to the inferred process $x(t)$, it should fire preferentially at times when the process $x(t)$ yields high values. If the coupling $C_i \approx 0$, we expect the $i^{\text{th}}$ single unit's activity is independent of process $x(t)$. We compute the 'process-tuning curves' of 3 single units in Fig 5**D** (in grey) with strong, intermediate and weak coupling $C_i$, respectively. These tuning curves are obtained similarly to the empirical mean process distribution, but the average is taken only over the observed spike times of the respective single unit. Single units with strong couplings $C_i$ are hypothesised to have tuning curves shifted towards positive values of $x(t)$, while units with weak couplings should yield a tuning curve matching empirical mean process distribution (in blue). For the single unit with strong coupling $C_i$ (Fig 5**D** left) the hypothesised strong shift towards positive values of expected process $x(t)$ can be observed. For intermediate coupling (centre) the tuning curve is closer to the empirical mean process distribution. A basically uncoupled unit yields a tuning curve overlapping with the empirical mean process distribution (right). In Fig 5**E**, we show the cumulative tuning curves of all 20 single units and confirm the previously observed effect: The higher the values of coupling $C_i$, the more the cumulative tuning curve deviates from cumulative empirical mean process distribution (blue line).

## 3. Discussion

We present how to fit a LIF population model subject to doubly-stochastic input (fast neuron-specific white-noise fluctuations and slow common Markovian input) to spike time data. After evaluating accuracy on simulated data we validated practicality on in-vitro, and in-vivo recorded data.

More precisely, we show that for the proposed model the likelihood can be computed efficiently under a small set of assumptions. By maximising this likelihood the model parameters are recovered accurately, and a posterior over the common input process $x(t)$ is inferred, that captures well the dynamics of the true process. Furthermore, we are able classify the type of input dynamics, i.e. continuous OUP vs. piece–wise constant MJP. We demonstrate with in–vitro recordings of single neurons, that the (known) oscillating mean input current can be recovered well from spiking data. For a population of in–vivo recorded single–units the presented methodology characterises the spiking activity well with the parameters obtained from the inferred model. In both experimental datasets the model with LIF neurons is superior to a model with the more commonly used Poisson assumption.

**Related Work** We leverage the fact that the ISI density $p_{\text{ISI}}$ can be computed efficiently by solving the first–passage time problem for a LIF neuron. Inference with such likelihoods has been proposed previously (Mullowney and Iyengar, 2008; Ladenbauer et al., 2018),

but the present work constitutes a non–trivial and substantial extension by considering a population of neurons receiving non–stationary doubly–stochastic inputs. Inferring dynamics from (single–neuron) spike–trains with 'LIF–inspired' likelihoods was considered before by Kim and Shinomoto (2012). Instead of directly working with the ISI density $p_{\mathrm{ISI}}$, they propose a mapping to an analytically tractable Gamma density. Additional approximations allow to avoid discretisation over the input process. The present work shows that the discretisation allows to incorporate easily different process priors over the mean of input. Kim and Shinomoto (2012) only consider a diffusion process, which is implausible as a generative model, since the expected variance of such a process scales linearly with time (Gardiner, 2009); The MJP and OUP priors are more reasonable choices in that sense.

Earlier work addressed whether recorded spike trains are better described by a continuous or a jumping process (Mochizuki and Shinomoto, 2014; Latimer et al., 2015). Those works considered single neuron activity, Poisson likelihoods, and simpler process priors compared to the ones presented here. We showed that in terms of goodness–of–fit considering LIF is superior to Poisson assumptions. This could also affect the result whether a jump or a continuous process explains the data better. How the presented physiological constraint model compares to more flexible model classes with many more parameters, e.g. generalised linear models, will be an interesting point to address in the future.

**Potential extensions & limitations** One of the main technical limitations of the presented approach is the stationary approximation of the effective inputs $\mu_{k,i_k}$ to the neuron. This assumption can be seen as 'event–based' binning scheme, different to other statistical modelling approaches that discretise time according to a regular grid (Pillow et al., 2008; Gaudino et al., 2014). This 'event–based' binning was already utilised by Kim and Shinomoto (2012) and allows to relate the proposed LIF population model to a hidden Markov model. The approximation limits us to the analysis of data with reasonably high firing rates $> 1$ Hz. Also the more common regular binning scheme suffers from this problem.

In this work we focused on modelling the non–stationarity of the observed neuronal population by a latent low–dimensional input process $x(t)$ accounting for the unobserved population activity. This is hypothetically dominating the observed activity compared to the neuronal interactions among the observed neurons. However, one could additionally try to account for the latter in the future. For neuronal couplings the perturbations caused by spikes of the other observed neurons have to be taken into account. For I&F models subject to stationary white noise this problem was addressed by small perturbation theory by (Ladenbauer et al., 2018); This might be also applicable for inference of the proposed non-stationary model. Other methods, that solve the first–passage time problem approximately, but efficiently (Schnoerr et al., 2017) are also potential candidates to address such problems in the future.

In this work, we choose the LIF neuron as spiking model, because it contains a minimal set of parameters. In principle other models from the I&F class, e.g. the exponential I&F neuron (Gerstner and Kistler, 2002), can be incorporated. The only requirement is that the spiking mechanism can be described by a renewal process. These models are usually more flexible, but require the optimisation of more parameters. In addition, including absolute refractory period for the spiking mechanism is possible by a simple shift of the ISI density. In practice, however, spike sorting errors in in-vivo data often violate this refractory period.

While we only assume two qualitatively different Markovian priors for the shared input fluctuations $x(t)$, other process priors could be included. For example, a differentiable process $x(t)$ can be assumed, whose temporal correlation-function is described by specific Matérn–kernels or periodic kernels. These kernels can be incorporated into the state-space framework (Solin, 2016) .

In this work we limited ourselves to one–dimensional dynamics for the slow input dynamics $x(t)$. Higher dimensional processes could be included, but become quickly infeasible when considering that these dimensions are correlated due to the discretisation of the process dimension $x(t)$.

**Experimental significance** We have seen that our model can reproduce the mean input $x(t)$ for in-vitro recorded spiking data. In general, this method is more interesting for data where the process is not known as in the second experimental dataset. We have shown that the inferred couplings $C_{1:N}$ allow to distinguish between single units participating in coordinated population firing, and those that fire independently. This phenomenon was already investigated by Okun et al. (2015), who called these two types of neurons 'choristers' and 'soloists', respectively. They showed that this characterisation is correlated with the underlying synaptic connectivity. For future applications, it will be interesting to study how model parameters and different process priors explain data recorded under different experimental conditions, such as varying stimuli, attention modulation etc.

Overall, the methodological framework presented here, allows for investigating non–stationary spiking data statistically, with model assumptions that are hypothetically closer to the observed neuronal system than those of commonly used alternative methods.

## 4. Materials and methods

### 4.1 Generative model

**Membrane voltage dynamics** We consider a population of $N$ leaky I&F neurons driven by fluctuating inputs. The dynamics of the membrane voltage $V_i$ of neuron $i$ ($i \in \{1, \dots, N\}$) are governed by the stochastic differential equation

$$\frac{dV_i}{dt} = -\frac{V_i}{\tau_{\mathrm{m}}} + \mu_i(t) + \sigma_i \xi_i(t), \tag{4}$$

where $\tau_{\mathrm{m}}$ denotes the membrane time constant. The input variations are described by the time-dependent mean $\mu_i(t)$, Gaussian white noise process $\xi_i(t)$ such that $\langle \xi_i(t)\xi_j(t+\Delta)\rangle = \delta_{i,j}\delta(\Delta)$ for $i, j \in \{1, \dots, N\}$ with expectation $\langle\cdot\rangle$, and noise magnitude $\sigma_i$. The neuron fires a spike when the membrane voltage reaches the spike threshold value $V_{\mathrm{s}}$, subsequently $V_i$ is reset to the value $V_{\mathrm{r}}$.

It is not meaningful to estimate all model parameters. A change of $V_{\mathrm{s}}$ or $V_{\mathrm{r}}$ can be completely compensated in terms of spiking dynamics by appropriate changes of $\mu_i$ and $\sigma_i$, which can be seen using the change of variables $\tilde{V}_i := (V_i - V_{\mathrm{r}})/(V_{\mathrm{s}} - V_{\mathrm{r}})$. Consequently, we may exclude $V_{\mathrm{s}}$ and $V_{\mathrm{r}}$ from the parameters to be inferred and instead set them to reasonable values. Furthermore, we fix $\tau_{\mathrm{m}}$ for most of our results, since a change of that parameter can also be effectively compensated by changes of $\mu_i$ and $\sigma_i$ (see supplementary material). The parameter values were $\tau_{\mathrm{m}} = 10$ ms, $V_{\mathrm{s}} = -40$ mV, $V_{\mathrm{r}} = -65$ mV.

**Shared input dynamics** The dynamics of the mean input $\mu_i(t)$ are determined by a common process $x(t)$ through

$$\mu_i(t) = C_i x(t) + \bar{\mu}_i, \tag{5}$$

where $C_i$ denotes the strength of coupling of neuron $i$ to $x(t)$ and $\bar{\mu}_i$ is the offset for neuron $i$.

We separately consider two processes with qualitatively different dynamics for $x(t)$: an *Ornstein-Uhlenbeck process* (OUP), described by

$$\frac{dx}{dt} = -\frac{1}{\tau}x(t) + \sqrt{\frac{2}{\tau}}\xi(t),$$

with time constant $\tau$ and Gaussian white noise process $\xi(t)$, $\langle \xi(t)\xi(t+\Delta)\rangle = \delta(\Delta)$; alternatively, $x(t)$ is described by a *Markov jump process* (MJP) which is piece-wise constant across intervals with exponentially distributed duration of mean $\tau$ and takes values drawn from a standard normal distribution $\mathcal{N}(0,1)$,

$$x(t) = x_l \quad \text{for } t \in \left[\sum_{l'=0}^{l-1}\gamma_{l'}, \sum_{l'=0}^{l}\gamma_{l'}\right),$$
$$x_l \sim \mathcal{N}(0,1),\ \gamma_l \sim \mathrm{Exp}(\tau^{-1}) \quad \text{for } l \in \mathbb{N}.$$

This process is also known as marked (homogeneous) Poisson point process. For both processes the stationary distribution is standard normal, $\lim_{t\to\infty} x(t) \sim \mathcal{N}(0,1)$, and the autocorrelation function is given by

$$\langle x(t)x(t+\Delta)\rangle = \exp\left(-\frac{|\Delta|}{\tau}\right).$$

Hence, our doubly-stochastic model incorporates fast independent input fluctuations by $\sigma_i\xi_i(t)$ in Eq (4) with Gaussian white noise process $\xi_i(t)$ and slower shared input variations by $\mu_i(t)$ via Eq (5) with OUP or MJP $x(t)$.

### 4.2 Statistical modelling

**Spike-based discretisation** We consider having observed the spike trains of $N$ neurons and collect these data in a sequence of ISIs $s_{1:K} := (s_1, \ldots, s_K)$ that are ordered increasingly according to the last spike time of each ISI. That is, $s_1$ corresponds to the neuron whose second spike time is the earliest, $s_2$ corresponds to the neuron whose $k$-th spike time comes next for $k \geq 2$, etc. Under the assumption that the process $x(t)$ evolves slowly compared to the duration of the ISIs we approximate $x(t)$ for each neuron across each ISI effectively by one value: $\mu_{i_k}(t) = \mu_{k,i_k} = C_{i_k}x_k + \bar{\mu}_{i_k}$ for all times $t$ within the $k$-th ISI $s_k$ in the sequence $s_{1:K}$, where $x_k$ is the value of $x(t)$ at the end time of $s_k$ and $i_k$ indicates the neuron that corresponds to that ISI. Note that in order to simplify notation we utilise that $s_k$, explicitly the duration of the $k$-th ISI, informs also (implicitly) about the start and end times of the ISI.

Hence, for the inference problem $x(t)$ is replaced by the sequence $x_{1:K} = (x_1, \ldots, x_K)$ which greatly facilitates this task.

**Markov property** Due to the fact that both processes, OUP and MJP, are Markovian we can factorise the probability density of the sequence given the time constant parameter as

$$p(x_{0:K}|\tau) = \prod_{k=1}^{K} p(x_k|x_{k-1},\tau)p(x_0), \tag{6}$$

where $p(x_0)$ is the the probability density for the initial process value $x_0 := x(0)$. We assume $p(x_0) \sim \mathcal{N}(0,1)$ which corresponds to the stationary distribution of the process. The transition probability density for the OUP is given by

$$p(x_k|x_{k-1},\tau) = \mathcal{N}\left(x_{k-1}\exp\left[-\frac{\Delta_k}{\tau}\right],\ 1-\exp\left[-\frac{2\Delta_k}{\tau}\right]\right),$$

where $\Delta_k = t_k - t_{k-1}$. For notational convenience we do not explicitly indicate $\Delta_k$ in $p(x_k|x_{k-1},\tau)$ and instead use that $x_k$ also contains information about the corresponding time point. For the MJP, on the other hand, we have

$$p(x_k|x_{k-1},\tau) = (1-P_\mathrm{j})\delta_{x_k,x_{k-1}} + P_\mathrm{j}\mathcal{N}(0,1),$$

where $\delta_{x_k,x_{k-1}}$ is the Kronecker delta and the probability of jumping is $P_\mathrm{j} = 1-\exp(-\Delta_k/\tau)$. Note that $P_\mathrm{j}$ denotes the probability that at least one jump occurs during an interval of duration $\Delta_K$.

**Data likelihood** Since spike emission for each model neuron is a renewal process, the likelihood of the observed data $s_{1:K}$ conditioned on the sequence $x_{1:K}$ (which approximates the process $x(t)$) and neuron-specific parameter values can be factorised as

$$p(s_{1:K}|x_{1:K},\vartheta) = \prod_{k=1}^{K} p(s_k|\mu_{k,i_k},\vartheta_{i_k}) \tag{7}$$

with $\mu_{k,i_k} = C_{i_k}x_k + \bar{\mu}_{i_k}$. $\vartheta = \{\vartheta_1,\ldots,\vartheta_N\}$ contains the parameters for each neuron, where $\vartheta_i = \{C_i,\bar{\mu}_i,\sigma_i,\tau_\mathrm{m}\}$. Each factor on the right hand side is the ISI probability density of an I&F neuron exposed to Gaussian white noise input with constant parameter values evaluated at one point. Fortunately, the ISI density $p(s_k|\mu_{k,i_k},\vartheta_{i_k})$ can be efficiently computed with high precision for a range of values $s \geq 0$ at once. This is achieved by solving a Fokker-Planck partial differential equation that describes the so-called first passage time problem for the stochastic I&F model (Richardson, 2008; Mullowney and Iyengar, 2008; Ladenbauer et al., 2018). Here we apply the finite volume solution method described in (Ladenbauer et al., 2018). In practice, we pre-compute $p(s_k|\mu_{k,i_k},\vartheta_{i_k})$ on a sufficiently fine grid of values for $s_k$ and $\mu_{k,i_k}$, which is then used as a look-up table in the optimisation procedure described below. For values of the mean input $\mu_{k,i_k}$ that are encountered during the inference procedure and are not on the grid we use linearly interpolated value for optimisation.

**Joint and marginal likelihoods** Combining Eqs (6) and (7) we yields the joint likelihood of the observed data and the process sequence as

$$p(s_{1:K},x_{0:K}|\vartheta,\tau) = \prod_{k=1}^{K} p(s_k|\mu_{k,i_k},\vartheta_{i_k})p(x_k|x_{k-1},\tau)p(x_0)$$

with $\mu_{k,i_k} = C_{i_k} x_k + \bar{\mu}_{i_k}$. To obtain the marginal likelihood of having observed the data given the model an integration over $x_{0:K}$ is required:

$$p(s_{1:K}|\vartheta, \tau) = \int p(s_{1:K}, x_{0:K}|\vartheta, \tau) dx_{0:K}. \tag{8}$$

Note the difference between this marginal likelihood and the likelihood in Eq (7) which is conditioned on knowledge of the process sequence.

### 4.3 Parameter estimation

Estimates for a particular set of parameters are obtained by maximising the marginal likelihood with respect to those parameters (maximum likelihood estimation). In the following we describe how we evaluate Eq (8) and how we maximise it with respect to the parameters of interest.

**Evaluating the marginal likelihood** Integration over $x_{0:K}$ in Eq (8) can be achieved sequentially. Assuming knowledge of $p(x_{k-1}|s_{1:k-1}, \vartheta, \tau)$ allows for the prediction step

$$p(x_k|s_{1:k-1}, \vartheta, \tau) = \int p(x_k|x_{k-1}, \tau) p(x_{k-1}|s_{1:k-1}, \vartheta, \tau) dx_{k-1}, \tag{9}$$

which is known as the Chapman-Kolmogorov (forward) equation. The iteration is initialised for $k = 1$ and $p(x_0|s_{1:0}, \vartheta, \tau) := p(x_0)$. In the following step we incorporate the next observation $s_k$ by the filtering

$$p(s_k, x_k|s_{1:k-1}, \vartheta, \tau) = p(s_k|\mu_{k,i_k}, \vartheta_{i_k}) p(x_k|s_{1:k-1}, \vartheta, \tau)$$

with $\mu_{k,i_k} = C_{i_k} x_k + \bar{\mu}_{i_k}$. By marginalising out $x_k$ we obtain

$$p(s_k|s_{1:k-1}, \vartheta, \tau) = \int p(s_k, x_k|s_{1:k-1}, \vartheta, \tau) dx_k$$

and calculate in the last step

$$p(x_k|s_{1:k}, \vartheta_{1:N}, \tau) = \frac{p(s_k, x_k|s_{1:k-1}, \vartheta, \tau)}{p(s_k|s_{1:k-1}, \vartheta, \tau)}, \tag{10}$$

which completes the iteration $k-1 \to k$. After $K$ iterations the marginal likelihood is given by

$$p(s_{1:K}|\vartheta, \tau) = \prod_{k=1}^{K} p(s_k|s_{1:k-1}, \vartheta, \tau). \tag{11}$$

Practically, the steps of the iteration described above are performed using a reasonable discretisation for $x$ (bins over the range $[-3.5, 3.5]$ with width 0.05). Consequently, the step in Eq (9), for example, involves a square matrix for the transition probability density and a sum for the integral.

**Likelihood maximisation for a single neuron** We maximise the logarithm of the likelihood in Eq. (11) with respect to the parameters $C$, $\bar{\mu}$, and $\tau$ using a simplex optimisation method (Nelder and Mead, 1965). Note that derivatives of the (log) likelihood can only be obtained numerically, which renders gradient-based optimisation methods less well suited. The parameter $\sigma$ is estimated via an outer loop, where we iterate over a range of values for $\sigma$ and maximise the likelihood for each of those. The membrane time constant $\tau_{\mathrm{m}}$ is not optimised, because not having the optimal value can be compensated well by the optimisation of $\sigma$ (see Fig A.1).

**Likelihood maximisation for a population** The parameter space for optimisation has $3N + 1$ dimensions (with $\tau_{\mathrm{m}}$ fixed as considered here, otherwise the dimensionality would be $3N + 2$), where $N$ is the number of neurons. To reduce computational efforts and obtain parameter estimates in reasonable time we applied the following approximate optimisation scheme. First, we estimate the noise amplitudes $\sigma_{1:N}$ by fitting for each spike train a neuron model with independent process $x_i(t)$ separately, as described in the previous paragraph. Then, we estimate the offsets $\bar{\mu}_{1:N}$ by fitting $N$ model neurons with $C_i = 0$ ($i \in \{1, \ldots, N\}$) and $\sigma_{1:N}$ obtained in the previous step. Note that these two steps can be performed in parallel across neurons. Next, using the full model we optimise with respect to $C_{1:N}$ given $\tau$ and vice versa in an alternating way keeping $\sigma_{1:N}$ and $\bar{\mu}_{1:N}$ fixed. The optimisation with respect to $C_{1:N}$ is done in parallel across neurons, for optimising $C_i$ the couplings of other neurons are fixed to value found in the previous optimisation step. Finally, after convergence of the previous iteration for $C_{1:N}$ and $\tau$, our estimates for the parameters $C_{1:N}$ and $\bar{\mu}_{1:N}$ are improved by optimising with respect to $C_i$, $\bar{\mu}_i$ using the previously determined values as starting points and keeping all other parameters fixed, for $i \in \{1, \ldots, N\}$. This last step is performed again in parallel. Although this procedure appears rather approximate, it yields (surprisingly) accurate parameter estimates (see Results section 3).

### 4.4 Classification and model comparison

To classify the dynamics of the slow (shared) input variations we compare the two model variants – one using the OUP and one using the MJP – by applying the well-established log-likelihood ratio. This quantity can be expressed as the difference of the logarithm of maximal likelihoods,

$$\mathrm{LLR}(\mathrm{OUP}, \mathrm{MJP}) := \log \max_{\vartheta,\tau} p_{\mathrm{OUP}}(s_{1:K}|\vartheta, \tau) - \log \max_{\vartheta,\tau} p_{\mathrm{MJP}}(s_{1:K}|\vartheta, \tau),$$

where the subscripts OUP and MJP indicate the considered process. Note that this way of selecting the best model does not (need to) take into account the model complexity, because both compared models have the same number of estimated parameters.

To compare models with different complexities (number of parameters to be estimated), specifically the I&F and the Poisson models, we apply the Akaike information criterion (AIC) (Akaike, 1974). This measure is given by $2N_{\vartheta,\tau} - 2 \log \max_{\vartheta,\tau} p(s_{1:K}|\vartheta, \tau)$, where $N_{\vartheta,\tau}$ is the number of model parameters to be determined. The preferred model is indicated by the smallest AIC value.

### 4.5 Reconstruction of time series

In order to reconstruct the process sequence we compute the posterior probability density of the hidden process sequence $x_{0:K}$ (given all observed ISIs) using the parameter estimates for $\vartheta$ and $\tau$. In particular, we calculate $p(x_k|s_{1:K}, \vartheta, \tau)$ for $k \in \{1, \ldots, K\}$ by

$$p(x_k|s_{1:K}, \vartheta, \tau) = \frac{p(s_{k+1:K}|x_k, \vartheta, \tau)p(x_k|s_{1:k}, \vartheta, \tau)}{p(s_{k+1:K}|s_{1:k}, \vartheta, \tau)}, \tag{12}$$

The probability density $p(x_k|s_{1:k}, \vartheta, \tau)$ on the right hand side is already available (cf. Eq (10)). The probability density $p(s_{k+1:K}|x_k, \vartheta, \tau)$ is calculated iteratively by backward smoothing,

$$p(s_{k+1:K}|x_k, \vartheta, \tau) = \int p(s_{k+2:K}|x_{k+1}, \vartheta, \tau)p(x_{k+1}|x_k, \tau)p(s_{k+1}|\mu_{k+1,i_{k+1}}, \vartheta_{i_{k+1}})dx_{k+1}.$$

with $\mu_{k,i_k} = C_{i_k}x_k + \bar{\mu}_{i_k}$. Since $p(x_K|s_{1:K}, \vartheta, \tau)$ is already given by forward filtering, we see from Eq (12) that $p(s_{K+1:K}|x_{K-1}, \vartheta, \tau) = 1$ (for $k = K-2$), which is the initialisation for the backward iteration. The denominator in Eq (12) is obtained by numerical marginalisation. Using the sequence $(p(x_1|s_{1:K}, \vartheta, \tau), \ldots, p(x_K|s_{1:K}, \vartheta, \tau))$ we can reconstruct the process $x(t)$ by the sequence of expected values $(\langle x_0\rangle, \ldots, \langle x_K\rangle)$, where $\langle x_0\rangle$ is calculated using $p(x_0)$.

The sequence of posterior probability densities further allows us to calculate a sequence of (instantaneous) population spike rate estimates $(r_0, \ldots, r_K)$ by

$$r_k := \frac{1}{N}\sum_{i=1}^{N}\left\langle \Big( \int_0^\infty s\, p(s|\mu_{k,i}, \vartheta_i)ds \Big)^{-1} \right\rangle, \tag{13}$$

with $\mu_{k,i} = C_i x_k + \bar{\mu}_i$, where the expectation $\langle\cdot\rangle$ is with respect to $p(x_k|s_{1:K}, \vartheta, \tau)$. Each summand in Eq (13) is the expected inverse mean ISI a neuron at the time that corresponds to $x_k$.

### 4.6 Experimental data

**In–vitro data** We analyse in–vitro recorded data from somatosensory mouse brain slices. Experimental details can be found in Pozzorini et al. (2013). All experiments were performed in accordance with rules of the Swiss Federal Veterinary Office. Activity in L5 pyramidal neurons was recorded while the excitatory synaptic transmission in the slice was blocked. The recorded neuron was stimulated by a current

$$I(t) = I_0 + \Delta I_{\text{mean}} \sin\left(\frac{2\pi}{\mathcal{T}}t\right) + \Delta I_{\text{noise}} N(t), \tag{14}$$

with constant offset $I_0$, sine amplitude $\Delta I_{\text{mean}}$, and noise amplitude $\Delta I_{\text{noise}}$. $N(t)$ is an Ornstein-Uhlenbeck process with zero mean, unitary variance and temporal correlation of 3 ms. This fast process is used as approximation of white noise input to the neuron (not to be confused with the much slower OUP assumed for the common input process $x(t)$). $\mathcal{T}$ is the period of the oscillatory mean current. In this work we focus on stimulus oscillations with

periods $\mathcal{T} = \{4\text{ s}, 8\text{ s}, 16\text{ s}\}$. Parameters $I_0, \Delta I_{\text{mean}}, \Delta I_{\text{noise}}$ were tuned, such that neuron's firing rate oscillates between 2 and 6 Hz, resulting in typical ranges of $100 - 450$ pA for $I_0$, $15 - 30$ pA for $\Delta I_{\text{mean}}$, and $50 - 150$ pA for $\Delta I_{\text{noise}}$. For analysis, only cells with average firing rate $> 4$ Hz are considered. Each recording was 68 s long, where the input stimulus is constant current for the first 4 s followed by the stimulation with Eq (14). We fit our model only to the 60 s, where the stimulation is described by Eq (14). For the analysis of recovered oscillations in Fig 4 the first 8 s of the stimulation are excluded because of a strong onset transient dominating the spiking activity. For optimisation of the noise amplitude $\sigma$ in our model, we fitted the model for a range of $\sigma$'s between 0.1 and $3\frac{\text{mV}}{\sqrt{\text{ms}}}$ with step size $0.1\frac{\text{mV}}{\sqrt{\text{ms}}}$. We report only the fit with the $\sigma$ yielding the maximal likelihood.

For analysis of the inferred oscillations, we compute the expected input $\langle\mu(t)\rangle = C\langle x(t)\rangle + \bar{\mu}$, where the expectation is over the posterior $p(x(t)|s_{1:K}, \vartheta, \tau)$. Since we have the expectation only at the spike times, we interpolate linearly between them to get estimations of the mean input at all times. The instantaneous phase $\phi_{\text{est}}(t)$ is computed from the Hilbert transformed signal. In Fig 4 **A**–**C** (bottom) we plot $\sin(\phi_{\text{true}}(t))$ (grey) and $\sin(\phi_{\text{est}}(t))$ (red) to compare the inferred oscillations to the true signal.

To quantify how well the true oscillation of the mean input current is recovered, we compute the synchrony index (SI) (Cohen, 2008) as

$$\text{SI} = \frac{1}{n_{\max}} \sum_{n=1}^{n_{\max}} \exp(\text{i}(\phi_{\text{true}}(t_n) - \phi_{\text{est}}(t_n))),$$

where $\{t_n\}_{n=1}^{n_{\max}}$ are the time points of experimental measurements. The absolute value |SI|, the SI length, is always between 0 and 1, where 1 means strong synchrony. The mean phase shift of the signal can be assessed by $\arctan(\text{Im(SI)}/\text{Re(SI)})$.

As a completely data driven measure of how strong neurons lock to the oscillating mean of the input current we use the resultant vector length (RVL) of the circular mean of the true phase at spike times $\{t_k\}_{k=0}^{K}$

$$\text{RVL} = \left| \frac{1}{K+1} \sum_{k=0}^{K} \exp\left(\text{i}\phi_{\text{true}}(t_k)\right) \right|.$$

**In–vivo data** The dataset is an extracellular in–vivo recording of anaesthetised monkeys. For details see (Kohn and Smith, 2016; Smith and Kohn, 2008). All experimental procedures complied with guidelines approved by the Albert–Einstein College of Medicine of Yeshiva University and New York University Animal Welfare Committees. Extracellular potentials were recorded from a 'Utah' array implanted in primary visual cortex (V1). After spike sorting, data consist of spike times of several single and multi-units. We considered the first 5 min of recordings from monkey 5. As done by Smith and Kohn (2008), units with a signal-to-noise ratio $< 2.75$ (average wave form amplitude divided by the standard deviation of the wave form noise) are discarded as multi–units. In addition, we exclude all units that had an average firing rate above 100 Hz. Spikes at the end of an ISI shorter than 3 ms are discarded as spike sorting error. ISIs longer than 1 s will be considered to have the length of 1 s (Note, that this holds only for the ISI $s_k$, but the time between two considered time points $t_k - t_{t-1}$ remains unchanged.). We account for spike sorting errors by modifying the

likelihood as

$$p(s|\mu,\sigma) = U(s)p_{\text{error}} + (1 - p_{\text{error}})p_{\text{ISI}}(s|\mu,\sigma),$$

where $U(s)$ is a uniform distribution. We set the probability of sorting error to $p_{\text{error}} = 0.05$, which corresponds to error rates found experimentally (Rossant et al., 2016).

After data preprocessing we fit all remaining single units (26) separately to the model for different values of $\sigma_i$ in the range between 3.5 and $8\frac{\text{mV}}{\sqrt{\text{ms}}}$ with steps of $0.5\frac{\text{mV}}{\sqrt{\text{ms}}}$. This is done for the OUP and MJP assumption. The $\sigma_i$ values that yield the maximal likelihood are used for the subsequent analysis. For the population analysis we then choose the 20 single units with slowest inferred process in the single unit fits (i.e. those with the highest process time constant $\tau$, average of OUP and MJP fit is considered). Those are used for the analysis shown in Fig 5.

## Appendix A. Supplementary figures

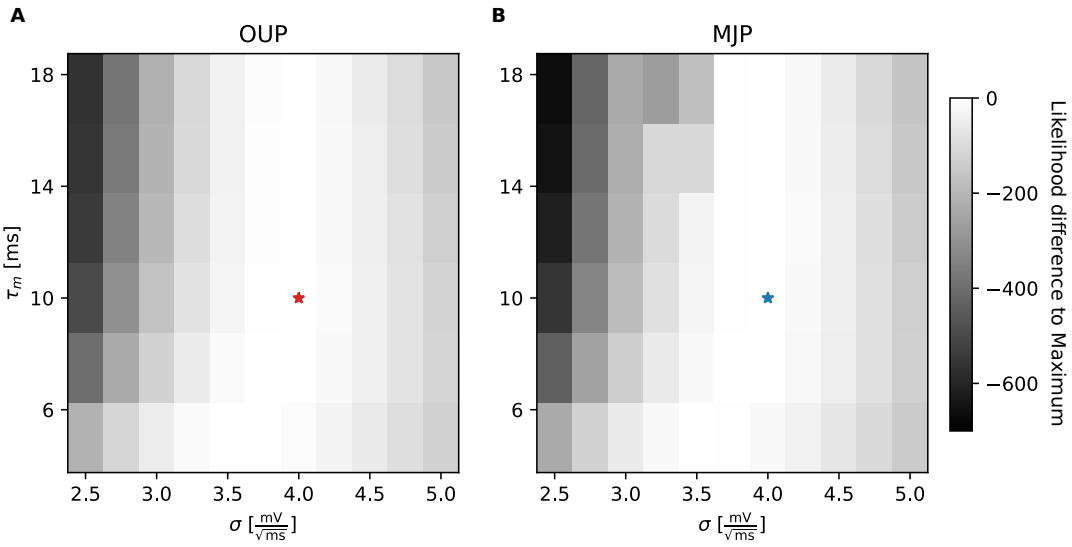



Figure A.1: **Maximised likelihood as function of membrane time constant $\tau_m$ and noise amplitude $\sigma$. A**: Fitting OUP model to $T = 300$ s of artificial data with $x(t)$ being an OUP and $\bar{\mu} = -4.7$ mV/ms, $C = 0.6$ mV/ms, $\tau = 500$ ms, $N = 1$. Red star marks the true values if $\sigma$ and $\tau_m$ data was generated with **B**: Same as **A** for the MJP.

## References

H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 12 1974. ISSN 0018-9286. doi: 10.1109/TAC.1974.1100705. URL `http://ieeexplore.ieee.org/document/1100705/`.

Johnatan Aljadeff, Benjamin J. Lansdell, Adrienne L. Fairhall, and David Kleinfeld. Analysis of Neuronal Spike Trains, Deconstructed. *Neuron*, 91(2):221–259, 7 2016. ISSN 10974199. doi: 10.1016/j.neuron.2016.05.039. URL `https://www.sciencedirect.com/science/article/pii/S0896627316302501`.

Moritz Augustin, Josef Ladenbauer, Fabian Baumann, and Klaus Obermayer. Low-dimensional spike rate models derived from networks of adaptive integrate-and-fire neurons: Comparison and implementation. *PLoS Computational Biology*, 13(6):e1005545, 6 2017. ISSN 15537358. doi: 10.1371/journal.pcbi.1005545. URL `http://dx.plos.org/10.1371/journal.pcbi.1005545`.

Laurent Badel, Sandrine Lefort, Romain Brette, Carl C. H. Petersen, Wulfram Gerstner, and Magnus J. E. Richardson. Dynamic I-V Curves Are Reliable Predictors of Naturalistic

Pyramidal-Neuron Voltage Traces. *Journal of Neurophysiology*, 99(2):656–666, 2 2008. ISSN 0022-3077. doi: 10.1152/jn.01107.2007. URL http://www.physiology.org/doi/10.1152/jn.01107.2007.

D. R. Brillinger. Maximum likelihood analysis of spike trains of interacting nerve cells. *Biological Cybernetics*, 59(3):189–200, 8 1988. ISSN 03401200. doi: 10.1007/BF00318010. URL http://link.springer.com/10.1007/BF00318010.

Nicolas Brunel and Mark C.W. Van Rossum. Lapicque's 1907 paper: From frogs to integrate-and-fire. *Biological Cybernetics*, 97(5-6):337–339, 12 2007. ISSN 03401200. doi: 10.1007/s00422-007-0190-0. URL http://link.springer.com/10.1007/s00422-007-0190-0.

E.J. Chichilnisky. A simple white noise analysis of neuronal light responses. *Network: Computation in Neural Systems*, 12(2):199–213, 1 2001. ISSN 0954-898X. doi: 10.1080/net.12.2.199.213. URL http://www.tandfonline.com/doi/full/10.1080/net.12.2.199.213.

Michael X Cohen. Assessing transient cross-frequency coupling in EEG data. *Journal of Neuroscience Methods*, 168(2):494–499, 3 2008. ISSN 01650270. doi: 10.1016/j.jneumeth.2007.10.012. URL http://www.ncbi.nlm.nih.gov/pubmed/18061683.

John P Cunningham and Byron M Yu. Dimensionality reduction for large-scale neural recordings. *Nature Neuroscience*, 17(11):1500–1509, 11 2014. ISSN 1097-6256. doi: 10.1038/nn.3776. URL http://www.nature.com/articles/nn.3776.

Christian Donner. Python Code: LIF population inference, 2018. URL https://github.com/christiando/doubly_stochastic_lif_inference.git.

Gaute T Einevoll, Felix Franke, Espen Hagen, Christophe Pouzat, and Kenneth D Harris. Towards reliable spike-train recordings from thousands of neurons with multielectrodes, 2 2012. ISSN 09594388. URL https://www.sciencedirect.com/science/article/pii/S0959438811001565.

Crispin W. Gardiner. *Stochastic methods : a handbook for the natural and social sciences*. Springer-Verlag Berlin Heidelberg, 4 edition, 2009. ISBN 9783540707127.

Mario Gaudino, Filippo Crea, Federico Cammertoni, Andrea Mazza, Amelia Toesca, and Massimo Massetti. Technical issues in the use of the radial artery as a coronary artery bypass conduit. *Annals of Thoracic Surgery*, 98(6):2247–2254, 2014. ISSN 15526259. doi: 10.1016/j.athoracsur.2014.07.039. URL http://www.nature.com/nature/journal/v440/n7087/abs/nature04701.html.

Wulfram Gerstner and Werner M. Kistler. *Spiking Neuron Models*. Cambridge University Press, Cambridge, 2002. ISBN 9780511815706. doi: 10.1017/CBO9780511815706. URL http://ebooks.cambridge.org/ref/id/CBO9780511815706.

Wulfram Gerstner and Richard Naud. How good are neuron models? *Science*, 326(5951):379–380, 10 2009. ISSN 00368075. doi: 10.1126/science.1181936. URL http://www.ncbi.nlm.nih.gov/pubmed/19833951.

Paul M. Harrison, Laurent Badel, Mark J. Wall, and Magnus J.E. Richardson. Experimentally Verified Parameter Sets for Modelling Heterogeneous Neocortical Pyramidal-Cell Populations. *PLoS Computational Biology*, 11(8):e1004165, 8 2015. ISSN 15537358. doi: 10.1371/journal.pcbi.1004165. URL `http://dx.plos.org/10.1371/journal.pcbi.1004165`.

A L Hodgkin and A F Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, 117(4):500–44, 8 1952. ISSN 0022-3751. URL `http://www.ncbi.nlm.nih.gov/pubmed/12991237`.

Renaud Jolivet, Ryota Kobayashi, Alexander Rauch, Richard Naud, Shigeru Shinomoto, and Wulfram Gerstner. A benchmark test for a quantitative assessment of simple neuron models. *Journal of Neuroscience Methods*, 169(2):417–424, 4 2008. ISSN 0165-0270. doi: 10.1016/J.JNEUMETH.2007.11.006. URL `https://www.sciencedirect.com/science/article/pii/S0165027007005535`.

Hideaki Kim and Shigeru Shinomoto. Estimating nonstationary input signals from a single neuronal spike train. *Phys Rev E*, 86(5):1–12, 2012. ISSN 15393755. doi: 10.1103/PhysRevE.86.051903.

A. Kohn and M. A. Smith. Utah array extracellular recordings of spontaneous and visually evoked activity from anesthetized macaque primary visual cortex (V1). Technical report, 2016.

Josef Ladenbauer, Sam McKenzie, Daniel Fine English, Olivier Hagens, and Srdjan Ostojic. Inferring and validating mechanistic models of neural microcircuits based on spike-train data. *bioRxiv preprint*, pages 1–31, 2018. doi: 10.1101/261016.

Kenneth W Latimer, Jacob L Yates, Miriam L R Meister, Alexander C Huk, and Jonathan W Pillow. Single-trial spike trains in parietal cortex reveal discrete steps during decision-making. *Science (New York, N.Y.)*, 349(6244):184–7, 7 2015. doi: 10.1126/science.aaa4056. URL `http://www.ncbi.nlm.nih.gov/pubmed/26160947`.

Artur Luczak, Peter Barthó, and Kenneth D Harris. Spontaneous Events Outline the Realm of Possible Sensory Responses in Neocortical Populations. *Neuron*, 62(3):413–425, 2009. ISSN 08966273. doi: 10.1016/j.neuron.2009.03.014. URL `http://www.sciencedirect.com/science/article/pii/S0896627309002372`.

Jan-Matthis Lueckmann, Pedro J. Goncalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnenmacher, and Jakob H. Macke. Flexible statistical inference for mechanistic models of neural dynamics. *Advances in Neural Information Processing Systems*, pages 1289–1299, 2017. ISSN 10495258.

Jakob H Macke, Lars Buesing, and Maneesh Sahani. Estimating state and parameters in state space models of spike trains. In *Advanced State Space Methods for Neural and Clinical Data*, pages 137–159. 2015. ISBN 9781139941433. doi: 10.1017/CBO9781139941433.007. URL `https://pdfs.semanticscholar.org/a71e/bf112cabd47cc67284dc8c12ab7644195d60.pdf`.

Valerio Mante, David Sussillo, Krishna V Shenoy, and William T Newsome. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, 503(7474): 78–84, 11 2013. ISSN 00280836. doi: 10.1038/nature12742. URL http://www.ncbi.nlm.nih.gov/pubmed/24201281.

Maurizio Mattia and Paolo Del Giudice. Population dynamics of interacting spiking neurons. *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, 66(5):19, 11 2002. ISSN 1063651X. doi: 10.1103/PhysRevE.66.051917. URL https://link.aps.org/doi/10.1103/PhysRevE.66.051917.

C. Daniel Meliza, Mark Kostuk, Hao Huang, Alain Nogaret, Daniel Margoliash, and Henry D. I. Abarbanel. Estimating parameters and predicting membrane voltages with conductance-based neuron models. *Biological Cybernetics*, 108(4):495–516, 8 2014. ISSN 0340-1200. doi: 10.1007/s00422-014-0615-5. URL http://link.springer.com/10.1007/s00422-014-0615-5.

Yasuhiro Mochizuki and Shigeru Shinomoto. Analog and digital codes in the brain. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 89(2):022705, 2 2014. ISSN 15502376. doi: 10.1103/PhysRevE.89.022705. URL https://link.aps.org/doi/10.1103/PhysRevE.89.022705.

Paul Mullowney and Satish Iyengar. Parameter estimation for a leaky integrate-and-fire neuronal model from ISI data. *Journal of Computational Neuroscience*, 24(2):179–194, 4 2008. ISSN 09295313. doi: 10.1007/s10827-007-0047-5. URL http://link.springer.com/10.1007/s10827-007-0047-5.

J. A. Nelder and R. Mead. A Simplex Method for Function Minimization. *The Computer Journal*, 7(4):308–313, 1 1965. ISSN 0010-4620. doi: 10.1093/comjnl/7.4.308. URL https://academic.oup.com/comjnl/article-lookup/doi/10.1093/comjnl/7.4.308.

Michael Okun, Nicholas A. Steinmetz, Lee Cossell, M. Florencia Iacaruso, Ho Ko, Pter Barthó, Tirin Moore, Sonja B. Hofer, Thomas D. Mrsic-Flogel, Matteo Carandini, and Kenneth D. Harris. Diverse coupling of neurons to populations in sensory cortex. *Nature*, 521(7553):511–515, 2015. ISSN 0028-0836. doi: 10.1038/nature14273. URL http://www.nature.com/doifinder/10.1038/nature14273.

Srdjan Ostojic and Nicolas Brunel. From Spiking Neuron Models to Linear-Nonlinear Models. *PLoS Computational Biology*, 7(1):e1001056, 1 2011. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1001056. URL http://dx.plos.org/10.1371/journal.pcbi.1001056.

Chethan Pandarinath, Daniel J. O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D. Stavisky, Jonathan C. Kao, Eric M. Trautmann, Matthew T. Kaufman, Stephen I. Ryu, Leigh R. Hochberg, Jaimie M. Henderson, Krishna V. Shenoy, Larry F. Abbott, and David Sussillo. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nat Methods*, 15(October):805–815, 2018. ISSN 1548-7091. doi: 10.1101/152884. URL https://www.biorxiv.org/content/early/2017/06/20/152884.

Jonathan W Pillow, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M Litke, E J Chichilnisky, and Eero P Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999, 2008.

Christian Pozzorini, Richard Naud, Skander Mensi, and Wulfram Gerstner. Temporal whitening by power-law adaptation in neocortical neurons. *Nat Neurosci*, 16:942–948, 6 2013. ISSN 1097-6256. doi: 10.1038/nn.3431. URL `http://www.nature.com/doifinder/10.1038/nn.3431`.

Christian Pozzorini, Skander Mensi, Olivier Hagens, Richard Naud, Christof Koch, and Wulfram Gerstner. Automated High-Throughput Characterization of Single Neurons by Means of Simplified Spiking Models. *PLOS Computational Biology*, 11(6):e1004275, 6 2015. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1004275. URL `https://dx.doi.org/10.1371/journal.pcbi.1004275`.

L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989. ISSN 00189219. doi: 10.1109/5.18626. URL `http://ieeexplore.ieee.org/document/18626/`.

Alfonso Renart, Nicolas Brunel, and Xiao-Jing Wang. Chapter 15 Mean-Field Theory of Irregularly Spiking Neuronal Populations and Working Memory in Recurrent Cortical Networks. Technical report. URL `https://pdfs.semanticscholar.org/35a4/e6db7f08817bbff169bd25df8bc53336a5f5.pdf`.

Magnus J E Richardson. Spike-train spectra and network response functions for non-linear integrate-and-fire neurons. *Biol. Cybern.*, 99(4-5):381–92, 2008. ISSN 03401200. doi: 10.1007/s00422-008-0244-y. URL `http://www.ncbi.nlm.nih.gov/pubmed/19011926`.

Cyrille Rossant, Shabnam N Kadir, Dan F M Goodman, John Schulman, Maximilian L D Hunter, Aman B Saleem, Andres Grosmark, Mariano Belluscio, George H Denfield, Alexander S Ecker, Andreas S Tolias, Samuel Solomon, Gyrgy Buzsáki, Matteo Carandini, and Kenneth D Harris. Spike sorting for large, dense electrode arrays. *Nature Neuroscience*, 19(4):634–641, 4 2016. ISSN 1097-6256. doi: 10.1038/nn.4268. URL `http://www.nature.com/articles/nn.4268`.

David Schnoerr, Botond Cseke, Ramon Grima, and Guido Sanguinetti. Efficient Low-Order Approximation of First-Passage Time Distributions. *Physical Review Letters*, 119(21), 2017. ISSN 10797114. doi: 10.1103/PhysRevLett.119.210601. URL `https://journals.aps.org/prl/pdf/10.1103/PhysRevLett.119.210601`.

Michael N Shadlen and William T Newsome. The Variable Discharge of Cortical Neurons: Implications for Connectivity, Computation, and Information Coding. *The Journal of Neuroscience*, 18(10):3870–3896, 1998. URL `http://www.jneurosci.org/content/jneuro/18/10/3870.full.pdf`.

Hideaki Shimazaki, Shun ichi Amari, Emery N. Brown, and Sonja Grün. State-space analysis of time-varying higher-order spike correlation for multiple neural spike train data. *PLoS Computational Biology*, 8(3), 2012. ISSN 1553734X. doi: 10.1371/journal.

pcbi.1002385. URL http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1002385.

Matthew A Smith and Adam Kohn. Spatial and Temporal Scales of Neuronal Correlation in Primary Visual Cortex. *The Journal of Neuroscience*, 28(48):12591–12603, 2008. ISSN 1529-2401. doi: 10.1523/jneurosci.2929-08.2008. URL http://www.jneurosci.org/content/jneuro/28/48/12591.full.pdf.

A Solin. Stochastic differential equation methods for spatio-temporal Gaussian process regression. 2016. URL https://aaltodoc.aalto.fi/handle/123456789/19842.

Corinne Teeter, Ramakrishnan Iyer, Vilas Menon, Nathan Gouwens, David Feng, Jim Berg, Aaron Szafer, Nicholas Cain, Hongkui Zeng, Michael Hawrylycz, Christof Koch, and Stefan Mihalas. Generalized leaky integrate-and-fire models classify multiple neuron types. *Nature Communications*, 9(1):709, 12 2018. ISSN 20411723. doi: 10.1038/s41467-017-02717-4. URL http://www.nature.com/articles/s41467-017-02717-4.

Wilson Truccolo. A Point Process Framework for Relating Neural Spiking Activity to Spiking History, Neural Ensemble, and Extrinsic Covariate Effects. *Journal of Neurophysiology*, 93(2):1074–1089, 2 2004. ISSN 0022-3077. doi: 10.1152/jn.00697.2004. URL http://www.physiology.org/doi/10.1152/jn.00697.2004.

# Chapter 10

# Conclusion

As we have emphasised in chapter 1, point process data are encountered frequently in various fields ranging from seismology to neuroscience. While simple methods to describe such data statistically, such as binning or Kernel density estimation, are widely used, more flexible but complex models have been proposed (Adams et al. 2009; Brix and Diggle 2001; Moller et al. 1998; Murray et al. 2009). Due to their complexity inference of these models is challenging, which often prevents their usage in practice. Hence, efficient and scalable inference algorithms are required to obtain tractable posteriors and to make these models attractive to a larger community.

This thesis contributes to the field of Bayesian inference of models for point process data in various ways. Several augmentation schemes were used to derive efficient inference algorithms for models with point process likelihoods and Gaussian priors. This combination constitutes a model class that is widely used to describe discrete, observed events (Adams et al. 2009; Brix and Diggle 2001; Moller et al. 1998; Murray et al. 2009). We showed the practicality with 3 different problems: An inhomogeneous Poisson process model, a density model, and a Markov jump process model. The first two models are potentially the most fundamental examples of point processes, and hence fast and scalable inference is critical for many practical applications. In chapter 3 and 4, we proposed such inference algorithms, that combine the augmentation schemes with the framework of sparse Gaussian processes (Csató and Opper 2002; Titsias 2009). In chapter 5, we showed how inference can be performed for a Markov jump problem, where the sigmoid link function arises naturally. Additionally, we included a non Gaussian (Laplace) prior and showed how this prior can be incorporated into the inference scheme. Finally, in chapter 6 we demonstrated how the augmentation schemes can be used to solve the inference problem for other specific models, namely GP multi–class classification and Hawkes processes. The results of this thesis together, with the related work discussed in chapter 6, contribute to extend the set of models, for which augmentation schemes can be used, and hence to derive more efficient inference algorithms for those models.

While the first part of the thesis discussed models for generic point process data, the second part considered the specific case of non–stationary spiking data. Nowadays, increasingly more spiking data are gathered from organisms to learn more about perceptual and behavioural correlates in the neuronal activity. Hence, understanding of these data is a frequently encountered challenge.

We showed that parameters of the two proposed models can be inferred efficiently from these data. Both models can be used to statistically describe the same dataset, though their approaches are quite different. Because of their dissimilar nature, the models' purposes are also different. The more phenomenological model discussed in chapter 8 is designed for exploratory data analysis to reveal effects that are potentially overseen with trial–average based methods. Furthermore, we expect, that this method scales well to larger populations, especially when couplings are discarded. On the other hand, being physiologically constrained, the LIF model in chapter 9 might be more suited to test the role of different mechanistic factors of the recorded system. Results are potentially easier

to interpret compared to phenomenological models. We expect that this method simplifies forward modelling of experimentally observed phenomena, because models of the I&F type are frequently used and widely accepted in this case (Gerstner and Kistler 2002; Renart et al. 2010). Hence, this work goes one step closer towards unifying the two different perspectives on the challenge associating experimental findings with theoretical modelling results for a deeper understanding of the brain and its computations.

# Part III

# Appendix

# A Augmentation for GP multi–class classification

In the GP community the multi-class classification is of particular interest of research, since the problem is more complex than binary classification tasks and efficient inference algorithms are difficult to obtain. Several approaches based on variational inference (Hensman et al. 2015a; Linderman and Adams 2015) and expectation propagation (Villacampa-Calvo and Hernández-Lobato 2017) have been proposed. In the following we introduce a particular likelihood for the multi–class problem, for that a Gaussian representation can be derived with the previously discussed augmentations.

## Multi-class likelihood

Consider a multi–class problem with $K$ classes. One data point consists of a pair of vectors $(\boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{x} \in \mathcal{X}$ is an observation in the observed feature-space $\mathcal{X}$ and $\boldsymbol{y} \in \{0,1\}^K$ is a binary vector, with $y_k = 1$ if $\boldsymbol{x}$ belongs to class $k$ and the remaining entries are $\boldsymbol{y}_{\setminus k} = \mathbf{0}$. Crucial for multi-class GP classification is to define the conditional observations of class–labels, given the observation $\boldsymbol{x}$, i.e. $p(\boldsymbol{y}|\boldsymbol{x})$. A common choice the softmax function

$$p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{g}) = \prod_{k=1}^{K} [s_k(\boldsymbol{x})]^{y_k} = \prod_{k=1}^{K} \left[ \frac{e^{h_k(\boldsymbol{x})}}{\sum_{l=1}^{K} e^{h_l(\boldsymbol{x})}} \right]^{y_k}, \tag{9}$$

where $\boldsymbol{g} = (g_1(\boldsymbol{x}), \ldots, g_K(\boldsymbol{x}))$ is a vector with values of $K$ latent functions at point $\boldsymbol{x}$, where we assume for $\boldsymbol{g}$ GP prior factorising in classes. A classical choice for functions $h_k$ is the identity $h_k(\boldsymbol{x}) = g_k(\boldsymbol{x})$, which then will have a GP prior. However, we will make a quite different choice and choose the function to be non–linear $h_k(\boldsymbol{x}) = \log \sigma(g_k(\boldsymbol{x}))$, where $\sigma(\cdot)$ is the sigmoid function. Hence, Eq (9) becomes

$$p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{g}) = \prod_{k=1}^{K} [s_k(\boldsymbol{x})]^{y_k} = \prod_{k=1}^{K} \left[ \frac{\sigma(g_k(\boldsymbol{x}))}{\sum_{l=1}^{K} \sigma(g_l(\boldsymbol{x}))} \right]^{y_k}. \tag{10}$$

Interestingly, for the binary classification setting one recovers the logistic likelihood, if one defines $g_2(\boldsymbol{x}) = -g_1(\boldsymbol{x})$. On first sight, Eq (10) appears more complicated than the classical choice. However, we will show, that we can render this function into a Gaussian form.

## Augmentation of softmax

We will require 3 steps to render the softmax into a Gaussian form.

**$\lambda$–Augmentation** We will make use of the fact that a denominator can be written as $z^{-1} \propto \int_0^\infty \exp(-\lambda z) dz$ and hence we can rewrite

$$p(y_k = 1|\boldsymbol{x}, \boldsymbol{g}) = s_k(\boldsymbol{x}) = \frac{\sigma(g_k)}{\sum_{l=1}^{K} \sigma(g_l)} = \sigma(g_k) \int_0^\infty \exp\left( -\lambda \sum_{l=1}^{K} \sigma(g_l) \right) d\lambda, \tag{11}$$

where the argument of the functions $g_l$, where dropped for notational convenience. Writing down Eq (11) conditioned on augmented variable $\lambda$ we get

$$p(y_k = 1|\boldsymbol{x}, \lambda, \boldsymbol{g}) = \sigma(g_k) \exp\left( -\lambda \sum_{l=1}^{K} \sigma(g_l) \right), \tag{12}$$

and we define the improper $\lambda$–prior being $p(\lambda) = 1_{[0,\infty[}(\lambda)$.

**Poisson augmentation** By noting that the moment generating function of a Poisson distribution $p_{\text{po}}(\cdot|\lambda)$ with mean parameter $\lambda$ is

$$\exp(\lambda(z-1)) = \sum_{n=1}^{\infty} z^n p_{\text{po}}(z|\lambda).$$

We can rewrite the exponential term in Eq (12) as

$$\exp\left(-\lambda \sum_{l=1}^{K} \sigma\left(g_l\right)\right) = \prod_{l=1}^{K} \exp\left(\sigma\left(-g_l\right) - 1\right) = \prod_{l=1}^{K} \sum_{\rho_l=0}^{\infty} \left[\sigma\left(-g_l\right)\right]^{\rho_l} p_{\text{po}}(\rho_l|\lambda).$$

Conditioning on the Poisson variables $\boldsymbol{\rho} \doteq \{\rho_l\}_{l=1}^{K}$, we rewrite Eq (12) as

$$p(y_k = 1|\boldsymbol{x}, \lambda, \boldsymbol{\rho}, \boldsymbol{g}) = \sigma\left(g_k\right) \prod_{l=1}^{K} \left[\sigma\left(-g_l\right)\right]^{\rho_l}, \tag{13}$$

and the prior for $\rho_l$ is $p(\boldsymbol{\rho}|\lambda) = \prod_l p_{\text{po}}(\rho_l|\lambda)$.

**Pólya–Gamma augmentation** Eq(13) is a product of sigmoid functions. The Pólya–Gamma augmentation allows us to rewrite (integer) powers of sigmoid functions as infinite scale mixture models

$$[\sigma(z)]^{\rho} = \frac{1}{2} \int_0^{\infty} e^{\rho \frac{z}{2} - \frac{z^2}{2}\omega} p_{\text{PG}}(\omega|\rho, 0) d\omega.$$

This allows us to rewrite Eq (13) as

$$\begin{aligned} p(y_k = 1|\boldsymbol{x}, \lambda, \boldsymbol{\rho}, \boldsymbol{g}) =& \frac{1}{2} \int_0^{\infty} e^{\frac{g_k}{2} - \frac{g_k^2}{2}\omega_k} p_{\text{PG}}(\omega_k|1, 0) d\omega_k \\ & \times \prod_{l=1}^{K} \int_0^{\infty} 2^{-\rho_l} e^{-\rho_l \frac{g_l}{2} - \frac{g_l^2}{2}\tilde{\omega}_l} p_{\text{PG}}(\tilde{\omega}_l|\rho_l, 0) d\tilde{\omega}_l, \end{aligned}$$

which has the desired Gaussian representation of Eq (10). Again we write down the conditional function

$$p(y_k = 1|\boldsymbol{x}, \lambda, \boldsymbol{\rho}, \omega_k, \tilde{\boldsymbol{\omega}}, \boldsymbol{g}) = \frac{1}{2} e^{\frac{g_k}{2} - \frac{g_k^2}{2}\omega_k} \prod_{l=1}^{K} 2^{-\rho_l} e^{-\rho_l \frac{g_l}{2} - \frac{g_l^2}{2}\tilde{\omega}_l} p_{\text{PG}}(\tilde{\omega}_l|\rho_l, 0),$$

where $\tilde{\boldsymbol{\omega}} = (\tilde{\omega}_1, \ldots, \tilde{\omega}_K)$ and the priors for $\omega_k$ and $\tilde{\boldsymbol{\omega}}$ are $p(\omega_k) = p_{\text{PG}}(\omega_k|1, 0)$ and $p(\tilde{\boldsymbol{\omega}}|\boldsymbol{\rho}) = \prod_l p_{\text{PG}}(\tilde{\omega}_l|\rho_l, 0)$, respectively. Writing the product of classes we get

$$p(\boldsymbol{y}|\boldsymbol{x}, \lambda, \boldsymbol{\rho}, \boldsymbol{\omega}, \tilde{\boldsymbol{\omega}}, \boldsymbol{g}) = \prod_{k=1}^{K} \left[p(y_k = 1|\boldsymbol{x}, \lambda, \boldsymbol{\rho}, \omega_k, \tilde{\boldsymbol{\omega}})\right]^{y_k},$$

with $\boldsymbol{\omega} = (\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_K)$.

## Sketching inference

The previous obtained likelihood is completely conjugate to the priors for all variables. Hence, it is straightforward to derive the conditional distributions for a Gibbs sampling scheme. Along the lines of chapter 3, it is also straightforward to implement a variational mean–field algorithm, where one only has to assume that the GPs $\boldsymbol{g}$ and the $\lambda$ parameters are independent of the Pólya–Gamma variables $\boldsymbol{\omega}, \tilde{\boldsymbol{\omega}}$ and Poisson variables $\rho$. For scalability sparse GP methods (Csató and Opper 2002;

Titsias 2009) and stochastic variational inference can be incorporated, additionally.

## B Alternative derivations of variational lower bound

Here, we show an alternative derivation of the variational lower bound, previously derived with the marked Poisson process augmentation. First, we note that the variational lower bound derived for the sigmoid function is equivalent to the bound of (Jaakkola and Jordan 2000)

$$\begin{aligned}\sigma(z) \geq& \sigma(z_0) \exp\left\{\frac{z-z_0}{2} - \frac{1}{4|z_0|}\tanh\left(\frac{z_0}{2}\right)\left(z^2 - z_0^2\right)\right\} \\ =& Z(z_0)\exp\left(-\frac{1}{2s(z_0)}\left(z-\mu(z_0)\right)^2\right)\end{aligned} \tag{14}$$

where

$$\begin{aligned}Z(z_0) =& \sigma(z_0)\exp\left(\frac{1}{2s(z_0)}(z_0-\mu(z_0))^2\right) \\ \mu(z_0) =& \frac{s(z_0)}{2} \\ (s(z_0))^{-1} =& \frac{1}{2|z_0|}\tanh\left(\frac{z_0}{2}\right)\end{aligned}$$

as shown in (Wenzel et al. 2018). Now, we want to bound $\exp\left(-\lambda\sigma(z)\right)$ from below with $\lambda \geq 0$. With Eq (14), we can write this as

$$\begin{aligned}\exp\left(-\lambda\sigma(z)\right) =& \exp\left(\lambda(\sigma(-z)-1)\right) \\ \geq& e^{-\lambda}\exp\left\{\lambda Z(-z_0)\exp\left(-\frac{1}{2s(z_0)}(-z-\mu(z_0))^2\right)\right\} \\ \geq& e^{-\lambda}\exp\left\{\lambda\sigma(-z_0)\left(1-\frac{1}{2s(z_0)}\left((-z-\mu(z_0))^2 - (-z_0-\mu(z_0))^2\right)\right)\right\} \\ =& e^{\lambda(\sigma(-z_0)-1)}\exp\left\{-\frac{\lambda\sigma(-z_0)}{2s(z_0)}\left((z+\mu(z_0))^2 - (z_0+\mu(z_0))^2\right)\right\}\end{aligned} \tag{15}$$

where for the second inequality we made use of the fact that $e^{-af(x)} \geq e^{-af(x_0)}\left(1 - a(f(x) - f(x_0))\right)$ for $f(x) \geq 0$.

Next we derive the lower bound for the augmentation scheme. We have

$$\begin{aligned}\exp\left(-\lambda\sigma(z)\right) =& \mathbb{E}_{P_\lambda}\left[e^{-\rho\frac{z}{2} - \frac{z^2}{2}\omega}\right] \\ \geq& \exp\left(\mathbb{E}_Q\left[(-\rho\frac{z}{2} - \frac{z^2}{2}\omega)\right] + \mathrm{D}_{\mathrm{KL}}(Q(\rho,\omega)\|P(\rho,\omega))\right)\end{aligned} \tag{16}$$

where $P_\lambda(\rho,\omega) = p_{\mathrm{PG}}(\omega|1,0)p_{\mathrm{po}}(\rho|\lambda\sigma(z))$. For $z = z_0$ we find the optimal posterior $Q(\rho,\omega) = p_{\mathrm{PG}}(\omega|\rho,|z_0|)p_{\mathrm{po}}(\rho|\lambda\sigma(-z_0))$. Then we can derive

$$\begin{aligned}\mathbb{E}_Q\left[\rho(-\frac{z}{2} - \frac{z^2}{2}\omega)\right] =& \lambda\sigma(-z_0)\left(-\frac{z}{2} - \frac{z^2}{2}\frac{1}{2|z_0|}\tanh\left(\frac{|z_0|}{2}\right)\right) \\ \mathrm{D}_{\mathrm{KL}}(Q(\rho,\omega)\|P(\rho,\omega)) =& \lambda(\sigma(-z_0)-1) + \lambda\sigma(-z_0)\left(\frac{z_0}{2} + \frac{z_0^2}{2}\frac{1}{2|z_0|}\tanh\left(\frac{|z_0|}{2}\right)\right)\end{aligned}$$

Substituting this into Eq (16) we get

$$\begin{aligned}\exp\left(-\lambda\sigma(z)\right) \geq& e^{\lambda(\sigma(-z_0)-1)}\exp\left(-\lambda\sigma(-z_0)\left(\frac{z-z_0}{2} + \frac{1}{2z_0}\tanh\left(\frac{z_0}{2}\right)\left(\frac{z^2-z_0^2}{2}\right)\right)\right) \\ =& e^{\lambda(\sigma(-z_0)-1)}\exp\left\{-\frac{\lambda\sigma(-z_0)}{2s(z_0)}\left((z+\mu(z_0))^2 - (z_0+\mu(z_0))^2\right)\right\}\end{aligned}$$

This result is the same as in Eq (15) and the proof is complete.

# Bibliography

Abbott, L. F. and Kepler, T. B. (1990). Model neurons: From Hodgkin-Huxley to hopfield. In *Statistical Mechanics of Neural Networks*, pages 5–18. Springer Berlin Heidelberg, Berlin, Heidelberg.

Adams, R. P., Murray, I., and MacKay, D. J. C. (2009). Tractable nonparametric Bayesian inference in Poisson processes with Gaussian process intensities. In *Proceedings of the 26th Annual International Conference on Machine Learning - ICML '09*, pages 1–8, New York, New York, USA. ACM Press.

Ahmed, M. A. and Alkhamis, T. M. (2009). Simulation optimization for an emergency department healthcare unit in Kuwait. *European Journal of Operational Research*, 198(3):936–942.

Ahrens, M. B., Orger, M. B., Robson, D. N., Li, J. M., and Keller, P. J. (2013). Whole-brain functional imaging at cellular resolution using light-sheet microscopy. *Nature Methods*, 10(5):413–420.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*, volume 4. Springer.

Bishop, C. M. and Tipping, M. E. (2000). Variational Relevance Vector Machines. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, pages 46—-53. Morgan Kaufmann Publishers.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, 112(518):859–877.

Brix, A. and Diggle, P. J. (2001). Spatiotemporal prediction for log-Gaussian Cox processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(4):823–841.

Brody, C. D. (1999). Correlations Without Synchrony. *Neural Computation*, 11(7):1537–1551.

Chornoboy, E. S., Schramm, L. P., and Karr, A. F. (1988). Maximum likelihood identification of neural point process systems. *Biological Cybernetics*, 59(4-5):265–275.

Csató, L. and Opper, M. (2002). Sparse On-Line Gaussian Processes. *Neural Computation*, 14(3):641–668.

Cunningham, J. P. and Yu, B. M. (2014). Dimensionality reduction for large-scale neural recordings. *Nature Neuroscience*, 17(11):1500–1509.

Daley, D. J. and Vere-Jones, D. (2008). *An Introduction to the Theory of Point Processes*.

Dayan, P. and Abbott, L. F. (2001). *Theoretical neuroscience : computational and mathematical modeling of neural systems*. Massachusetts Institute of Technology Press.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.

Donner, C. and Opper, M. (2017). Inverse Ising problem in continuous time: A latent variable approach. *Physical Review E*, 96(6):062104.

Donner, C. and Opper, M. (2018a). Efficient Bayesian Inference for a Gaussian Process Density Model. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 1–10, Monterey.

Donner, C. and Opper, M. (2018b). Efficient Bayesian Inference of Sigmoidal Gaussian Cox Processes. *Journal of Machine Learning Research*, 19(67):1–34.

Dunn, B., Mørreaunet, M., and Roudi, Y. (2015). Correlations and Functional Connections in a Population of Grid Cells. *PLoS Computational Biology*, 11(2):e1004052.

Einevoll, G. T., Franke, F., Hagen, E., Pouzat, C., and Harris, K. D. (2012). Towards reliable spike-train recordings from thousands of neurons with multielectrodes. *Current opinion in neurobiology*, 22(1):11–7.

Embrechts, P., Liniger, T., and Lin, L. (2011). Multivariate hawkes processes: An application to financial data. *Journal of Applied Probability*, 48A(A):367–378.

Feynman, R. P., Leighton, R. B., and Sands, M. (1964). *The Feynman Lectures on Physics: Volume II*.

Flaxman, S., Teh, Y. W., and Sejdinovic, D. (2017). Poisson intensity estimation with reproducing kernels. *Electronic Journal of Statistics*, 11(2):5081–5104.

Geman, S. and Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741.

Gerstner, W. and Kistler, W. M. (2002). *Spiking Neuron Models*. Cambridge University Press, Cambridge.

Glauber, R. J. (1963). Time-dependent statistics of the Ising model. *Journal of Mathematical Physics*, 4(2):294–307.

Gonçalves, F. B. and Gamerman, D. (2018). Exact Bayesian inference in spatiotemporal Cox processes driven by multivariate Gaussian processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(1):157–175.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109.

Hawkes, A. G. (1971). Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90.

Hensman, J., Matthews, A., and Ghahramani, Z. (2015a). Scalable Variational Gaussian Process Classification. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, pages 351–360.

Hensman, J., Matthews, A. G., Filippone, M., and Ghahramani, Z. (2015b). MCMC for Variationally Sparse Gaussian Processes. In *Advances in Neural Information Processing Systems 28*, pages 1648–1656.

Hodgkin, A. L. and Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, 117(4):500–544.

Ishiyama, S. and Brecht, M. (2016). Neural correlates of ticklishness in the rat somatosensory cortex. *Science (New York, N.Y.)*, 354(6313):757–760.

Jaakkola, T. S. and Jordan, M. I. (2000). Bayesian parameter estimation via variational methods. *Statistics and Computing*, 10(1):25–37.

Khan, M. E., Aravkin, A., Friedlander, M., and Seeger, M. (2013). Fast Dual Variational Inference for Non-Conjugate Latent Gaussian Models. In *Proceedings of the 30th International Conference on Machine Learning*, pages 951–959.

Khan, M. E. and Lin, W. (2017). Conjugate-Computation Variational Inference : Converting Variational Inference in Non-Conjugate Models to Inferences in Conjugate Models. *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pages 878–887.

Kilsby, C. G., Jones, P. D., Burton, A., Ford, A. C., Fowler, H. J., Harpham, C., James, P., Smith, A., and Wilby, R. L. (2007). A daily weather generator for use in climate change studies. *Environmental Modelling and Software*, 22(12):1705–1719.

Kim, H. and Shinomoto, S. (2012). Estimating nonstationary input signals from a single neuronal spike train. *Phys Rev E*, 86(5):1–12.

Kingman, J. F. C. (1993). *Poisson processes*. Clarendon Press.

Ladenbauer, J., McKenzie, S., English, D. F., Hagens, O., and Ostojic, S. (2018). Inferring and validating mechanistic models of neural microcircuits based on spike-train data. *bioRxiv preprint*, pages 1–31.

Lázaro-Gredilla, M. and Titsias, M. K. (2011). Variational heteroscedastic Gaussian process regression. In *In 28th International Conference on Machine Learning (ICML-11*, pages 841–848. [International Machine Learning Society].

Linderman, S., Johnson, M., and Adams, R. P. (2015). Dependent Multinomial Models Made Easy: Stick-Breaking with the Polya-gamma Augmentation. In *Advances in Neural Information Processing Systems 28*, pages 3456–3464.

Linderman, S. W. and Adams, R. P. (2015). Scalable Bayesian Inference for Excitatory Point Process Networks.

Linderman, S. W., Adams, R. P., and Pillow, J. W. (2016). Bayesian latent structure discovery from multi-neuron recordings. In *Advances in neural information processing systems*, pages 2002–2010.

Lindon, M. S. (2018). Continuous-Time Models of Arrival Times and Optimization Methods for Variable Selection.

Lloyd, C., Gunter, T., Osborne, M. A., and Roberts, S. J. (2014). Variational Inference for Gaussian Process Modulated Poisson Processes. *Proceedings of the 32nd International Conference on Machine Learning*, pages 1814–1822.

Mackay, D. and Gibbs, M. (2000). Variational Gaussian process classifiers. *IEEE Transactions on Neural Networks*, 11(6):1458–1464.

Macke, J. H., Opper, M., and Bethge, M. (2011). Common Input Explains Higher-Order Correlations and Entropy in a Simple Model of Neural Population Activity. *Physical Review Letters*, 106(20):208102.

Matthews, A. G. d. G., Hensman, J., Turner, R., and Ghahramani, Z. (2016). On Sparse Variational Methods and the Kullback-Leibler Divergence between Stochastic Processes. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51, pages 231–239.

Meng, X.-l. and Van Dyk, D. A. (1999). Seeking efficient data augmentation schemes via conditional and marginal augmentation. Technical Report 2.

Mercer, J. (1909). Functions of Positive and Negative Type, and their Connection with the Theory of Integral Equations. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 209(441-458):415–446.

Minka, T. P. (2001). Expectation propagation for approximate Bayesian inference. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 362–369. Morgan Kaufmann Publishers.

Moller, J., Syversveen, A. R., and Waagepetersen, R. P. (1998). Log Gaussian Cox Processes. *Scandinavian Journal of Statistics*, 25(3):451–482.

Mullowney, P. and Iyengar, S. (2008). Parameter estimation for a leaky integrate-and-fire neuronal model from ISI data. *Journal of Computational Neuroscience*, 24(2):179–194.

Murray, I., Ghahramani, Z., and MacKay, D. (2012). MCMC for doubly-intractable distributions. In *Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence*, page 569. AUAI Press.

Murray, I., MacKay, D., and Adams, R. P. (2009). The Gaussian Process Density Sampler. In *Advances in Neural Information Processing Systems*, pages 9–16.

Nguyen, H. C., Zecchina, R., and Berg, J. (2017). Inverse statistical problems: from the inverse Ising problem to data science. *Advances in Physics*, 66(3):197–261.

Ogata, Y. (1998). Space-Time Point-Process Models for Earthquake Occurrences. *Annals of the Institute of Statistical Mathematics*, 50(2):379–402.

Ohiorhenuan, I. E., Mechler, F., Purpura, K. P., Schmid, A. M., Hu, Q., and Victor, J. D. (2010). Sparse coding and high-order correlations in fine-scale cortical networks. *Nature*, 466(7306):617–621.

Opper, M., Fraccaro, M., Paquet, U., Susemihl, A., and Winther, O. (2015). Perturbation Theory for Variational Inference. In *NIPS Workshop on Approximate Inference*, pages 1–9.

Opper, M. and Winther, O. (2000). Gaussian Processes for Classification: Mean-Field Algorithms. *Neural Computation*, 12(11):2655–2684.

Palmer, J. A., Wipf, D. P., Kreutz-Delgado, K., and Rao, B. D. (2006). Variational EM algorithms for Non-Gaussian Latent Variable Models. In *Advances in Neural Information Processing Systems 18*, pages 1059–1066. MIT Press.

Pandarinath, C., O'Shea, D. J., Collins, J., Jozefowicz, R., Stavisky, S. D., Kao, J. C., Trautmann, E. M., Kaufman, M. T., Ryu, S. I., Hochberg, L. R., Henderson, J. M., Shenoy, K. V., Abbott, L. F., and Sussillo, D. (2018). Inferring single-trial neural population dynamics using sequential auto-encoders. *Nat Methods*, 15(October):805–815.

Penttinen, A. and Stoyan, D. (2000). Recent applications of point process methods in forestry statistics. *Statistical Science*, 15(1):61–78.

Pillow, J. W., Shlens, J., Paninski, L., Sher, A., Litke, A. M., Chichilnisky, E. J., and Simoncelli, E. P. (2008). Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999.

Polson, N. G., Scott, J. G., and Windle, J. (2013). Bayesian inference for logistic models using Pólya-Gamma latent variables. *Journal of the American Statistical Association*, 108(504):1339–1349.

Pontil, M., Mukherjee, S., and Girosi, F. (2000). On the Noise Model of Support Vector Machines Regression. pages 316–324. Springer, Berlin, Heidelberg.

Renart, A., de la Rocha, J., Bartho, P., Hollender, L., Parga, N., Reyes, A., and Harris, K. D. (2010). Asynchronous state in cortical circuits. *Science*, 24(5.9):16–37.

Ressel, P. (1976). A short proof of Schoenberg's theorem. *Proceedings of the American Mathematical Society*, 57(1):66–66.

Rieke, F., Bialek, W., van Stevenick, R. d. R., and Warland, D. (1997). *Spikes : exploring the neural code*. MIT Press.

Riihimäki, J. and Vehtari, A. (2014). Laplace Approximation for Logistic Gaussian Process Density Estimation and Regression. *Bayesian Analysis*, 9(2):425–448.

Sasaki, T., Matsuki, N., and Ikegaya, Y. (2007). Metastability of Active CA3 Networks. *Journal of Neuroscience*, 27(3):517–528.

Schneidman, E., Berry, M., Segev, R., and Bialek, W. (2006). Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature*, 440(7087):1007.

Scott, J. and Pillow, J. W. (2012). Fully Bayesian inference for neural models with negative-binomial spiking. *Advances in Neural Information Processing Systems*, 25:1898–1906.

Shimazaki, H., Sadeghi, K., Ishikawa, T., Ikegaya, Y., and Toyoizumi, T. (2015). Simultaneous silence organizes structured higher-order interactions in neural populations. *Scientific Reports*, 5(1):9821.

Shomali, S. R., Ahmadabadi, M. N., Rasuli, S. N., and Shimazaki, H. (2018). Uncovering Network Architecture Using an Exact Statistical Input-Output Relation of a Neuron Model. *bioRxiv*, page 479956.

Silverman, B. W. (1986). *Density estimation for statistics and data analysis*. Chapman and Hall.

Stensola, T., Stensola, H., Moser, M.-B., and Moser, E. I. (2015). Shearing-induced asymmetry in entorhinal grid cells. *Nature*, 518(7538):207–212.

Stuart, A. and Ord, J. K. (2010). *Kendall's Advanced Theory of Statistics, Vol I, Distribution Theory*. Arnold, 6 edition.

Titsias, M. (2009). Variational Learning of Inducing Variables in Sparse Gaussian Processes. In *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5, pages 567–574.

Tsodyks, M., Kenet, T., Grinvald, A., and Arieli, A. (1999). Linking spontaneous activity of single cortical neurons and the underlying functional architecture. *Science*, 286(5446):1943–1946.

Villacampa-Calvo, C. and Hernández-Lobato, D. (2017). Scalable Multi-Class Gaussian Process Classification using Expectation Propagation. In *Proceedings of the 34th International Conference on Machine Learning*, pages 3550–3559.

Wenzel, F., Galy-Fajou, T., Donner, C., Kloft, M., and Opper, M. (2018). Efficient Gaussian Process Classification Using Polya-Gamma Data Augmentation.

Wilkinson, D. J. (2006). *Stochastic Modelling for Systems Biology*. Chapman & Hall/CRC Mathematical & Computational Biology. Taylor & Francis.

Yedidia, J. (2013). Understanding belief propagation and its generalizations. *Journal of Chemical Information and Modeling*, 53(9):1689–1699.

Zammit-Mangion, A., Dewar, M., Kadirkamanathan, V., and Sanguinetti, G. (2012). Point process modelling of the Afghan War Diary. *Proceedings of the National Academy of Sciences of the United States of America*, 109(31):12414–9.

Zeng, H.-L., Alava, M., Aurell, E., Hertz, J., and Roudi, Y. (2013). Maximum likelihood reconstruction for Ising models with asynchronous updates. *Physical Review Letters*, 110(21):210601.

# Glossary

| *Abbreviation* | *Full name* |
|---|---|
| AUC | area under curve |
| EM | expectation maximisation |
| GLM | generalised linear model |
| GMM | Gaussian mixture model |
| GP | Gaussian process |
| GPFA | Gaussian process factor analysis |
| I&F | integrate–and–fire |
| ISI | inter spike interval |
| KDE | kernel density estimator |
| LIF | leaky integrate–and–fire |
| MAP | maximum a posteriori |
| MCMC | Markov chain Monte Carlo |
| MJP | Markov jump process |
| MLE | maximum likelihood estimate |
| MSE | mean squared error |
| OUP | Ornstein–Uhlenbeck process |
| RKHS | reproducing kernel Hilbert space |
| ROC | receiver operating characteristic |
| RVL | resultant vector length |
| SI | synchrony index |
| VB | variational Bayes |

# Contributions

This section delineates my contribution to each of the preceding chapters.

**Chapter 1** I wrote the introduction chapter.

**Chapter 2** I wrote the introduction chapter for the first part.

**Chapter 3** I (CD) and Prof. Manfred Opper (MO) conceived and designed the work. CD derived the inference algorithms and developed the Python code. CD performed the numerical experiments and produced all figures. CD wrote the manuscript with substantial contribution of MO.

**Chapter 4** I (CD) and Prof. Manfred Opper (MO) conceived and designed the work. CD derived the inference algorithms and developed the Python code. CD performed the numerical experiments and produced all figures. CD wrote the manuscript with substantial contribution of MO.

**Chapter 5** I (CD) and Prof. Manfred Opper (MO) conceived and designed the work. CD derived the inference algorithms and developed the Python code. CD performed the numerical experiments and produced all figures. CD wrote the manuscript with substantial contribution of MO.

**Chapter 6** I wrote the conclusion chapter for the first part.

**Chapter 7** I wrote the introduction chapter for the second part.

**Chapter 8** I (CD) and Prof. Manfred Opper (MO) conceived and designed the work. CD derived the inference algorithms and developed the Python code. CD performed the numerical experiments and produced all figures. CD wrote the manuscript with contribution of MO.

**Chapter 8** I (CD) and Dr. Josef Ladenbauer (JL) conceived and designed the work with help of Prof. Manfred Opper (MO). CD derived the inference algorithms and developed the Python code. CD performed the numerical experiments. CD wrote the manuscript with substantial contribution from JL.

**Chapter 10** I wrote the conclusion chapter.

**Appendix B** I did all derivations and wrote this appendix.

**Appendix A** I did all derivations and wrote this appendix.

# Copyright