Poisson-Networks: A Model for Structured Point Processes

Shyamsundar Rajaram
ECE Department
University of Illinois
Urbana, USA

Thore Graepel
MLP Group
Microsoft Research
Cambridge, UK

Ralf Herbrich
MLP Group
Microsoft Research
Cambridge, UK

Abstract

Modelling structured multivariate point process data has wide ranging applications like understanding neural activity, developing faster file access systems and learning dependencies among servers in large networks. In this paper, we develop the Poisson network model for representing multivariate structured Poisson processes. In our model each node of the network represents a Poisson process. The novelty of our work is that waiting times of a process are modelled by an exponential distribution with a piecewise constant rate function that depends on the event counts of its parents in the network in a generalised linear way. Our choice of model allows to perform exact sampling from arbitrary structures. We adopt a Bayesian approach for learning the network structure. Further, we discuss fixed point and sampling based approximations for performing inference of rate functions in Poisson networks.

1 Introduction

Structured multivariate point processes appear in many different settings ranging from multiple spike train recordings, file access patterns and failure events in server farms to queuing networks. Inference of the structure underlying such multivariate point processes and answering queries based on the learned structure is an important problem. For example, learning the structure of cooperative activity between multiple neurons is an important task in identifying patterns of information transmission and storage in cortical circuits [Brillinger and Villa, 1994, Aertsen et al., 1989, Oram et al., 1999, Harris et al., 2003, Barbieri et al., 2001, Brown et al., 2004]. Similarly, learning the access patterns of files can be exploited for building faster file access systems.

Consider V time series of events $\mathbf{t}_i$. We would like to find a compact representation of the joint probability distribution of the V time series. Assuming each time series is modelled by an inhomogeneous Poisson process, a unique representation is obtained in terms of V rate functions $\lambda_i(t)$ each of which depends on all events of the V times series up to time t. Clearly, we need further assumption on the rate functions to infer the rate functions from a finite amount of data. The proposed approach makes four crucial assumptions:

1. The rate function of each process depends only on the history in a short time window into the past.
2. The rate function of each process depends only on a small number of other processes. Adopting a directed graph notation, these are also referred to as *parent* nodes.
3. The rate function of each process depends only on the empirical rates of its parents.
4. The rate function of each process is parameterised by a generalised linear model.

The standard approach adopted in modelling such time series is by discretising the time axis into intervals of fixed length δ and transforming the time series into a sequence of counts per interval. The dependency structure between nodes of such a network, which is also known as a *dynamic Bayesian network* (DBN) [Dean and Kanazawa, 1989, Murphy, 2001], can be modelled using transition probabilities between states at time t and $t + \delta$ given the states of all the parents of each node at time t. This approach suffers from the problem that the discretisation is somewhat arbitrary: A small value of δ will result in a redundant representation and a huge computational overhead but a large value of δ may smooth away important details in the data.

We overcome the issues of discretisation by considering the limit $\delta \to 0$ and thus modelling the waiting time in each time series directly. Nodelman et al. [2002] and Nodelman et al. [2003] use a similar continuous time approach for modelling homogeneous Markov processes with a finite number of states. In their work the rate functions depend on the current state of the parents only which leads to an efficient and exact learning algorithm. In our setup, we model the waiting times as an exponential distribution with piecewise constant rates that depend on the count history of parent nodes.

The paper is structured as follows: In Sec. 2 we introduce our Poisson network model. In Sec. 3 we describe an efficient technique for performing exact sampling from a Poisson network. We describe parameter estimation and structure learning using approximate Bayesian inference techniques in Sec. 4. In Sec. 5 we discuss approximate marginalisation and inference based on sampling and fixed point methods. Finally, we describe experiments on data generated by our sampling technique for performing parameter learning, structure estimation and inference in Sec. 6.

2 The Poisson Network Model

Consider time series data $\mathbf{t}_i \in (\mathbb{R}^+)^{N_i}$ from V point processes. Each element of $\mathbf{t}_i = [t_{i,1}, \ldots, t_{i,N_i}]$ corresponds to series of times $t_{i,j}$ at which a particular event occurred. We model each time series as an inhomogeneous Poisson process and the modelling problem is to capture the dependency between different processes, both qualitatively (structure) and quantitatively (parameters).

A Poisson process is an instance of a counting process which is characterised by a rate function $\lambda(t)$. If the rate function is constant over time, the Poisson process is called *homogeneous*, otherwise it is called *inhomogeneous* [Papoulis, 1991]. A very useful property of a homogeneous Poisson process is that the waiting time between two consecutive events is exponentially distributed with rate λ, that is,

$$\forall t \in \mathbb{R}^+ : p(t|\lambda) := \lambda \exp(-\lambda t)\,.$$

Note that the mean of the waiting time distribution is given by λ^{-1}.

The waiting time distribution for a non-homogeneous process is a generalized exponential distribution. In the special case of a piecewise constant rate function $\lambda(t)$, the waiting time has a piecewise exponential distribution.

Proposition 1 (Piecewise Exponential Distribution). *Suppose we are given l rates $\boldsymbol{\lambda} \in (\mathbb{R}^+)^l$ and l sets $\mathcal{T}_k$ of non-overlapping intervals where the average waiting time in each of the intervals T in $\mathcal{T}_k$ is governed by λ_k. The waiting time of the piecewise exponential distribution has the following density:*

$$\begin{aligned} p(t|\lambda, \mathcal{T}) &:= \prod_{k=1}^{l} {\lambda_k}^{a_k(t,\mathcal{T}_k)} \exp(-\lambda_k b_k(t, \mathcal{T}_k)), \\ a_k(t, \mathcal{T}_k) &:= \sum_{[t_0,t_1)\in\mathcal{T}_k} \mathbb{I}_{t\in[t_0,t_1)}, \\ b_k(t, \mathcal{T}_k) &:= \sum_{[t_0,t_1)\in\mathcal{T}_k} (t_1 - t_0)\mathbb{I}_{t>t_1} + (d - t_0)\mathbb{I}_{t\in[t_0,t_1)}. \end{aligned}$$

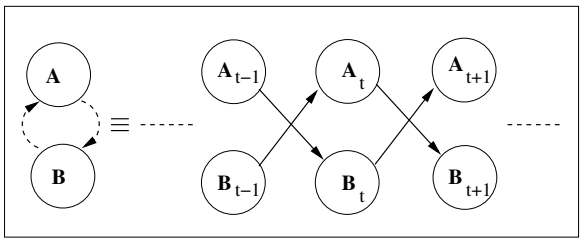

Figure 1: Illustration of a Poisson network with a cycle unwrapped in time. Note that we use dashed lines to indicate parent relationships in Poisson networks. These arrows should always be interpreted as pointing forward in time.

In the Poisson network model we aim at modelling the independence relations between the V processes. Similarly to Bayesian networks, the independence relations are implicitly represented by the connectivity of a directed graph. Hence, the network structure can be fully represented by all parent relationships, $M := \{\pi(i) \subseteq \{1, \ldots, V\}\}$. Interestingly, the semantics of a parent relationship is slightly different from Bayesian networks: Since the rate function of each node only depends on the past history of its parents, cycles w.r.t. the parent set M are permissible (see Figure 1).

We model the dependency of a node i on its parents $\pi(i) = \{p_1, \ldots, p_{m_i}\}$ by constraining the rate function $\lambda_i(t)$ to be a function of the event counts of all parents in time windows of length ϕ, $\mathbf{n}_i(t) := [n_{i,p_1}(t), \ldots, n_{i,p_{m_i}}(t)]$ where n_{i,p_j} represents the number of events of node p_j in the time window $[t-\phi, t)$.[1]

We consider a generalized linear model for the rate function $\lambda(t)$ in terms of the counts $\mathbf{n}_i(t)$. Possible link functions include the probit or sigmoid function which exhibit a natural saturation characteristic. However, in the following we will consider the canonical link

[1] Note that we will treat ϕ as a fixed quantity throughout the paper.

function for the Poisson model resulting in:

$$\lambda_i(t; \mathbf{w}_i, \mathbf{x}_i) = \exp\left(w_{i,0} + \sum_{j \in \pi(i)} w_{i,j} x_{i,j}(t)\right), \quad (1)$$

where $w_{i,0}$ represents a bias term and translates into a multiplicative base rate $\exp(w_{i,0})$. We define

$$\begin{aligned} x_{i,j}(t) &:= \ln\left(1 + \hat{\lambda}_{i,j}(t)\right), \\ \hat{\lambda}_{i,j}(t) &:= \frac{n_{i,j}(t)}{\phi}. \end{aligned}$$

Note that $\hat{\lambda}_{i,j}(t)$ is the *empirical* rate of node j w.r.t. node i. Alternatively, the rate function can be written in a way that is more amenable for inference (see Sec. 5):

$$\lambda_i(t) = \exp(w_{i,0}) \prod_{j \in \pi(i)} \left(1 + \hat{\lambda}_{i,j}(t)\right)^{w_{i,j}}.$$

A positive value for $w_{i,j}$ indicates an excitatory effect and a negative value corresponds to an inhibitory effect on the rate. The rate of each node at any given time instant is a function of the empirical rate of its parents resulting in an inhomogeneous process. Note that in practice there are only finitely many different count vectors due to the finite length time windows. The piecewise constant characteristic of the rate function and the absence of cycles in the Poisson network enables us to do exact sampling (see Sec. 3).

Let us derive the probability distribution of the time series corresponding at given nodes. Consider the $l-1^{\text{th}}$ and l^{th} events occurring at times t_{l-1} and t_l in a given node and the instants at which the count vector changes in this interval be represented as $\breve{t}_{l,1}, \ldots, \breve{t}_{l,k_l}$.[2] The probability density of an event at time t_l given the previous event happened at t_{l-1} is denoted by $p(t_l|t_{l-1}, \mathbf{w}, M)$. This density is a product of probabilities of non-occurrence of an event in each of the disjoint subintervals, i.e. $(t_{l-1}, \breve{t}_{l,1}], (\breve{t}_{l,1}, \breve{t}_{l,2}], \ldots, (\breve{t}_{l,k_l-1}, \breve{t}_{l,k_l}]$ and the probability density of occurrence of an event at t_l in the subinterval $(\breve{t}_{l,k_l}, t_l]$:

$$p(t_l|t_{l-1}, \mathbf{w}, M) = \lambda_{l,k_l+1} \left(\prod_{j=1}^{k_l+1} \exp\left(-\lambda_{l,j}\tau_{l,j}\right)\right),$$

where $\lambda_{l,j}$ is the rate as in (1) for a node which is a function of the event counts of its parents in the jth subinterval corresponding to the lth event and the durations $\tau_{l,j}$ are defined by

$$\begin{aligned} \tau_{l,1} &:= \breve{t}_{l,1} - t_{l-1}, \\ \tau_{l,j} &:= \breve{t}_{l,j} - \breve{t}_{l,j-1}, \quad j \in \{2, \ldots, k_l\} \\ \tau_{l,k_l+1} &:= t_l - \breve{t}_{l,k_l}. \end{aligned}$$

The probability density for the time series $\mathbf{t} = [t_1, \ldots, t_N]$ of a given node is obtained as the product of the probability densities for each of the mutually exclusive subintervals $(t_0, t_1], \ldots, (t_{N-1}, t_N]$,

$$p(\mathbf{t}|\mathbf{w}, M) = \prod_{l=1}^{N} p(t_l|t_{l-1}, \mathbf{w}, M), \quad (2)$$

where the initial time t_0 is assumed to be 0.

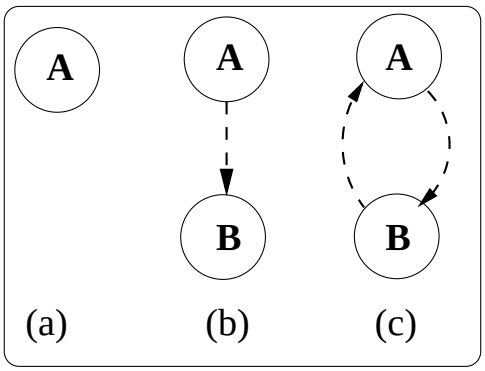

Figure 2: Poisson networks for illustrating sampling.

3 Sampling from a Poisson Network

The piecewise constant behavior of the rate function $\lambda(t)$ and the absence of cycles in the network allows us to perform exact sampling from a Poisson Network. Let us consider the simple case of sampling from a single node network as shown in Fig. 2 (a). Sampling is straightforward in this case, because the node represents a homogeneous Poisson process. The standard way to sample from a homogeneous Poisson process is to make use of the property that the waiting times between two adjacent events are iid and are distributed exponentially with rate parameter λ.

Let us consider a two node network shown in Fig. 2 (b). Sampling for node A can be done in a straightforward way as explained above because at any time t, A_t is independent of B_t. The rate function for node B can be calculated using (1) once the whole time series for node A is known. The rate function for B is piecewise constant because of the finite number of events for node A in the time window $[t-\phi, t]$ for varying t. Sampling from a waiting time distribution with a

[2] We drop the node subscript and suppress the dependence on the time series of all parent nodes for better readability.

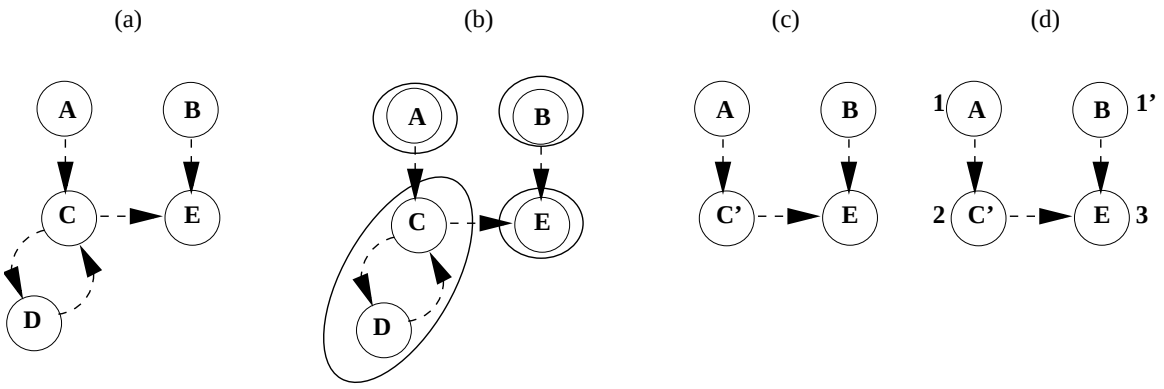

Figure 3: (a) An arbitrary Poisson network. (b) SCC's indicated by ellipses. (c) Directed acyclic graph C' represents the SCC formed by nodes C and D. (d) Topological ordering where the number ($1'$ and 1 can be sampled independently of each other in parallel) written adjacent to the node indicates the order.

piecewise constant rate function is done by rejection sampling. Denote the current value of the rate function by $\lambda(t)$ and assume the rate remains unchanged until time $\hat{t}$. Sample τ from the waiting time distribution with parameter $\lambda(t)$. Accept τ if $t + \tau \leq \hat{t}$ and reject τ otherwise. The sampling time is now updated to $\hat{t}$ or $t + \tau$ depending on whether the sample is rejected or accepted, respectively.

Now, let us consider a two node network as shown in Fig. 2 (c) where there exists a cyclic relationship between nodes A and B. It is worth mentioning again that node B depends on the event counts of node A in the past only and vice versa. Sampling from the nodes cannot be done in an independent way because of the cyclic relationship. Let the values $\lambda_A(t)$ and $\lambda_B(t)$ of the rate functions for nodes A and B, respectively, be calculated by (1) and assume that the rates remain constant until $\hat{t}_A$, $\hat{t}_B$ (excluding the mutual interaction in the future). Sample waiting times τ_A and τ_B for both nodes using the rates $\lambda_A(t)$ and $\lambda_B(t)$, respectively. The sample corresponding to $\max(\tau_A, \tau_B)$ is rejected because an earlier event at the other node might have changed the value of the rate function; the other sample is accepted if it is within the constant time interval of the firing rate. The sampling time is updated to $\min(\hat{t}_A, \hat{t}_B)$ or $t + \min(\tau_A, \tau_B)$ depending on whether the other sample was rejected or accepted, respectively.

This sampling technique can be generalised to an arbitrary number of nodes. We note that the structure of the network can be made use of in order to perform sampling in a very efficient way. The two key observations are that,

1. Certain groups of nodes have to sampled from in a synchronized, dependent way because of mutual dependence of nodes in the group.
2. Sampling has to be done in a certain order.

The first observation is illustrated in Fig. 3 (b) and such groups of nodes are known as strongly connected components (SCC) in graph theory [Cormen et al., 2001]. A strongly connected component is a directed subgraph with the property that for all ordered pairs of vertices (u, v), there exists a path from u to v. A variant of depth first search can be used to find all the SCCs of a graph. A directed acyclic graph graph is obtained by replacing each SCC by a single node as shown in Fig. 3 (c). Now we observe that the nodes of the directed acyclic graph can be sampled from in such a way that, when sampling a given node, its parents have been sampled before (see Fig. 3 (d)). Topological sorting is a method to compute such an order efficiently (see Cormen et al. [2001]).

4 Parameter Estimation and Structure Learning

Given time series data $\mathbf{T} := [\mathbf{t}_1, \ldots, \mathbf{t}_V]$ we are interested in finding the most plausible structure M^* and parameter estimates for $\mathbf{w}$ in the linear model of the rate functions (see (1)). We take a Bayesian approach to the problem of parameter estimation which, at the same time, provides a score over structure M by the marginalised likelihood. We choose a Gaussian prior distribution over the weights

$$p(\mathbf{W}|M) = \prod_{i=1}^{V} \mathcal{N}(\mathbf{w}_i; \mathbf{0}, \sigma^2 \mathbf{I}) \,.$$

Using Bayes rule, we obtain the posterior

$$p(\mathbf{W}|\mathbf{T}, M) = \frac{p(\mathbf{T}|\mathbf{W}, M) p(\mathbf{W}|M)}{p(\mathbf{T}|M)} \,.$$

The structure learning problem can be written as,

$$\begin{aligned} M^* &:= \operatorname*{argmax}_M \; p(M|\mathbf{T}) \\ &= \operatorname*{argmax}_M \; p(\mathbf{T}|M)p(M) \\ p(\mathbf{T}|M) &= \prod_{i=1}^{V} p(\mathbf{t}_i|\{\mathbf{t}_j|j \in \pi(i)\}) \\ p(M) &:= \prod_{i=1}^{V} p(\pi(i)) \,, \end{aligned}$$

where we make use of a structural prior $p(M)$ that factors over the parents of the V nodes. Note that $p(\mathbf{t}_i|\{\mathbf{t}_j|j \in \pi(i)\})$ corresponds to (2) marginalised over $\mathbf{w}$. In contrast to structure learning in Bayesian networks which requires optimisation over the set of directed acyclic graphs, no such constraints are imposed in Poisson network or continuous time Bayesian networks [Nodelman et al., 2003]. Hence, the structure learning problem can be solved by finding the most likely parents of each node independently. In theory the exact structure can be learned without regard to the number of parents. However, in practice the running time of the learning procedure is a function of the number of parents of a node and hence while learning structures we fix the maximum number of parents. Now we present two approximate Bayesian inference techniques for parameter estimation and structure learning.

4.1 Laplace Approximation

In the Laplace approximation [Kass and Raftery, 1995], the posterior is approximated by a multivariate Gaussian density,

$$p(\mathbf{w}_i|\mathbf{T}, M) \approx \mathcal{N}(\mathbf{w}_i; \mathbf{w}_{\text{MAP}}, \boldsymbol{\Sigma}) \,,$$

where $\mathbf{w}_{\text{MAP}}$ is the mode of the posterior,

$$\mathbf{w}_{\text{MAP}} := \operatorname*{argmax}_{\mathbf{w}_i} p(\mathbf{T}, \mathbf{w}_i|M) \,,$$

and the covariance matrix $\boldsymbol{\Sigma}$ is given by

$$\boldsymbol{\Sigma} := \left(-\nabla\nabla^T_{\mathbf{w}_i} \; \ln(p(\mathbf{T}, \mathbf{w}_i|M))|_{\mathbf{w}=\mathbf{w}_{\text{MAP}}}\right)^{-1} .$$

The marginalised likelihood can be obtained by,

$$\begin{aligned} p(\mathbf{T}|M) &= \int p(\mathbf{T}, \mathbf{w}_i|M) \; d\mathbf{w}_i \\ &\approx \int \sqrt{(2\pi)^{m_i}|\boldsymbol{\Sigma}|}\mathcal{N}(\mathbf{w}_i; \mathbf{w}_{\text{MAP}}, \boldsymbol{\Sigma}) \; d\mathbf{w}_i \\ &= \sqrt{(2\pi)^{m_i}|\boldsymbol{\Sigma}|} \,. \end{aligned}$$

The mode $\mathbf{w}_{\text{MAP}}$ is found by the conjugate gradients algorithm using the gradient of $\ln(p(\mathbf{T}, \mathbf{w}_i|M))$ w.r.t. $\mathbf{w}$ (see also (2)).

4.2 Variational Approach

The variational approach to solving problem of parameter estimation and structure learning is to introduce a family of probability distribution $q_{\boldsymbol{\theta}}(\mathbf{w}_i)$ parameterised over $\boldsymbol{\theta}$ which gives rise to a lower bound on the marginalised likelihood[3]:

$$\ln(p(\mathbf{T}|M)) \geq \mathrm{L}_M(\boldsymbol{\theta}) \,,$$

$$\mathrm{L}_M(\boldsymbol{\theta}) := \left\langle \ln\left(\frac{p(\mathbf{w}_i|M)}{q_{\boldsymbol{\theta}}(\mathbf{w}_i)}\right)\right\rangle_{q_{\boldsymbol{\theta}}} + \langle \ln(p(\mathbf{T}|\mathbf{w}_i, M))\rangle_{q_{\boldsymbol{\theta}}} \,,$$

which follows from an application of Jensen's inequality to the concave logarithm function [Jordan et al., 1999]. This lower bound on the log-marginal probability is a function of the parameters $\boldsymbol{\theta}$ of the distribution q. This bound is tight if we choose $q_{\boldsymbol{\theta}}(\mathbf{w}_i)$ to be the true posterior $p(\mathbf{w}_i|\mathbf{T}, M)$. We also observe that the lower bound is the sum of two terms which correspond to the negative of the Kullback-Leibler divergence between the prior term and the distribution $q_{\boldsymbol{\theta}}(\mathbf{w}_i)$ and the second term is the log-likelihood of the data averaged over $q_{\boldsymbol{\theta}}(\mathbf{w}_i)$.

The idea of the variational approximation Bayesian algorithm is to maximize the lower bound L_M with respect to the parameters of the q distribution. The structure learning problem can be posed as maximization of the lower bound $\mathrm{L}_M(\boldsymbol{\theta})$ which can be written as,

$$M^* := \operatorname*{argmax}_M \left[\max_{\boldsymbol{\theta}} \mathrm{L}_M(\boldsymbol{\theta})\right] .$$

Similar to the Laplace approximation, we choose $q_{\boldsymbol{\theta}}(\mathbf{w}_i)$ to be a multivariate Gaussian $\mathcal{N}(\mathbf{w}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma})$. For a given network structure M, we can use conjugate gradients to find the maximum of $\mathrm{L}_M(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Note that the gradients are given by

$$\begin{aligned} \nabla_{\boldsymbol{\mu}}(\mathrm{L}_M(\boldsymbol{\mu}, \boldsymbol{\Sigma})) &= -\boldsymbol{\mu} + \sum_{i=1}^{N} \mathbf{x}_{i,k_i} \ldots \\ &\quad - \sum_{i=1}^{N}\sum_{j=1}^{k_i} \tau_{i,j}\mathbf{x}_{i,j} \exp\left(\frac{\mathbf{x}^T_{i,j}\boldsymbol{\Sigma}\mathbf{x}_{i,j} + 2\boldsymbol{\mu}^T\mathbf{x}_{i,j}}{2}\right) , \\ \nabla_{\boldsymbol{\Sigma}}(\mathrm{L}_M(\boldsymbol{\mu}, \boldsymbol{\Sigma})) &= -\frac{1}{2\sigma^2}\mathbf{I} + \frac{\mathbf{1}}{\mathbf{2}}(\boldsymbol{\Sigma})^{-\mathbf{T}} \ldots \\ &\quad - \sum_{i=1}^{N}\sum_{j=1}^{k_i} \frac{1}{2}\tau_{i,j}\mathbf{x}_{i,j}\mathbf{x}^T_{i,j} \exp\left(\frac{\mathbf{x}^T_{i,j}\boldsymbol{\Sigma}\mathbf{x}_{i,j} + 2\boldsymbol{\mu}^T\mathbf{x}_{i,j}}{2}\right) . \end{aligned}$$

5 Inference in a Poisson network

Once the network structure and parameters are learned, several queries regarding the distribution represented by the network can be answered. A common

[3] We use the shorthand notation $\langle f(\cdot)\rangle_q$ to denote an expectation of f w.r.t. the distribution q.

inference problem that is encountered in a Bayesian network is one where data is available for some nodes (denoted by M_D), there are a few query nodes (denoted by M_Q) whose behavior has to be estimated from the data and the remaining nodes are hidden nodes (denoted by M_H for which no data is available. We pose an analogous problem for Poisson networks. The rate of any arbitrary node can be computed if the time series data for all its parents are known for the period of interest. However, certain configurations of the problem involving hidden nodes are not easy to solve because of the problem of *entanglement* as mentioned in Nodelman et al. [2002]. The standard procedure in a Bayesian network is to marginalise over all the hidden nodes and obtain an estimation for the query nodes using the data in observed nodes. Marginalising over a hidden node amounts to integrating over all possible time series for a particular node which, in general, is impossible. Hence, approximations are necessary.

5.1 Inference by Sampling

A straightforward way to solve the inference problem is to perform marignalisation of the hidden nodes by using a few instantiations of them which can be obtained by sampling from the network. The samples can then be used to obtain averaged rate estimates at each of the query nodes. The sampling procedure is the same as our procedure in Sec. 3 with a minor modification that the sampling is to be performed conditioned on the data that has been observed in the data nodes M_D. We observe that if the data for a node i is completely known, then the parent nodes of i do not have any influence on i in the subsequent sampling process. Hence, we can safely remove all the parental relationships for all the fully observed nodes and obtain a new network $M' = M - \{i \to j, i \in M_D, j \in \pi(i)\}$. Sampling is done from the new network M' for all the unobserved nodes and then average firing rates can be obtained for all the query nodes.

5.2 Estimation of Steady State Rate

An alternative way to solve the inference problem is to approximate the empirical rate $\hat{\lambda}$ with a steady state rate. According to our model, the rate of a node can be written as,

$$\lambda_i(t) = \exp\left(w_{i,0} + \sum_{j=1}^{V}\left(w_{i,j}\ln\left(1+\hat{\lambda}_{i,j}(t)\right)\right)\right) . \quad (3)$$

We notice that this is (1) if we consider $w_{i,j} = 0$ for all the pairs of nodes which are not dependent. The empirical rate $\hat{\lambda}_{i,j}(t)$ cannot be obtained unless the whole time series is observed. Hence, we approximate the empirical rate by the true rate,

$$\hat{\lambda}_{i,j}(t) = \frac{n_{i,j}(t)}{\phi} \approx \frac{1}{\phi}\int_{t-\phi}^{t}\lambda_j(\tilde{t})d\tilde{t} \approx \lambda_j(t)\,,$$

where we assume that the length of time window, ϕ, is very small. Substituting back in (3) we obtain

$$\ln(\lambda_i(t)) \approx w_{i,0} + \sum_{j=1}^{V} w_{i,j}\ln\left(1+\lambda_j(t)\right). \quad (4)$$

We make the assumption that the rate is constant in the time interval $[t-\phi, t]$ and hence the Poisson process of each of the parents $j, j \in \{1,\ldots,V\}$ is assumed to be a homogeneous Poisson process with rate $\lambda_j(t)$. Now, (4) can be constructed for all nodes that are not observed and the set of equations have the form $\boldsymbol{\lambda}(t) = F(\boldsymbol{\lambda}(t))$, $F : \mathbb{R}^V \to \mathbb{R}^V$, whose solution are the fixed points of the system. We perform fixed point iterations starting from randomly initialized values for $\lambda_i(t), i \in \{1 \ldots V\}$. In experiments we observe that at convergence, the estimated rate corresponds to the mean rate in the time interval $[t-\phi, t]$ calculated from actual data.

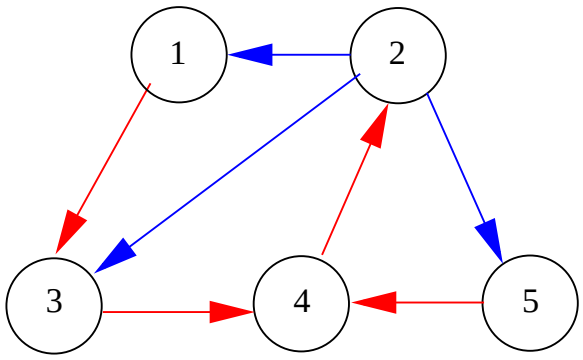

Figure 4: The random Poisson network graph that generated the samples in Fig. 5. Red arrow indicates an inhibitory influence and the blue arrow indicates an excitatory influence.

6 Experimental Results

In this section, we test the presented methods on data sampled using the algorithms from Sec. 3. We show experiments of approximate inference of rate using a sampling based approximation and a fixed point approximation.

Sampling Firstly, we generate a random graph (see Fig. 4) with V nodes. As mentioned before, because of computational issues we fix the maximum number of

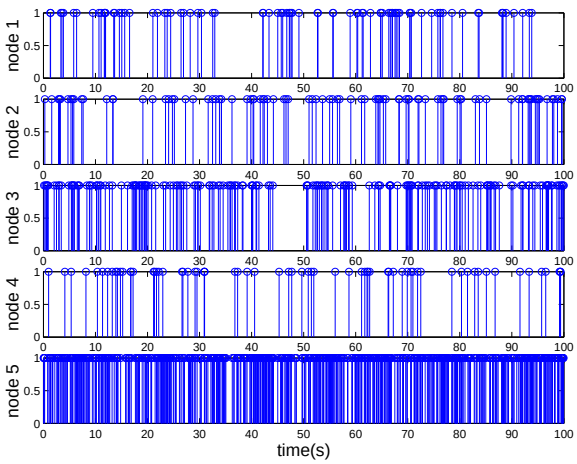

Figure 5: Samples generated from a random network with $V = 5$ and maximum number of parents is 2

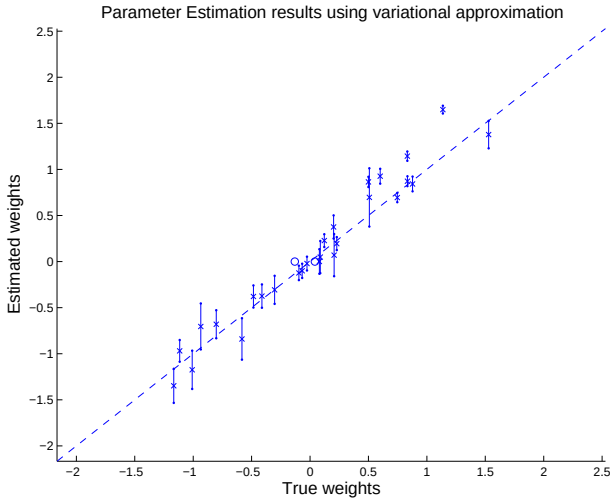

Figure 6: Results for parameter estimation and structure learning using the variational approximation for a network with 15 nodes in which maximum number of parents is 2. Each node i with a parent j has a true expected weight $w_{i,j}$ (x-axis) and a posterior estimate (y-axis) shown as a 5%-95% posterior quantile interval with a mark indicating the mean. The empty circles indicate the edges that were not identified by the structure learning algorithm.

parents $\pi_{\max}$ for every node. Now, our efficient sampling technique given in Sec. 3 is used to generate samples from the network. Samples generated from a randomly generated network with $V = 5$ and $\pi_{\max} = 2$ is as shown in Fig. 5. The time window duration ϕ was fixed to 1 second.

Parameter estimation and structure learning The Laplace and variational approximation developed in Sec. 4.1 and Sec. 4.2 are tested using samples generated from random graph structures. We observed that the posterior distribution $p(\mathbf{w}_i|\mathbf{T}, M)$ can be approximated very well by a multivariate Gaussian. Thus, both approximations methods perform very accurately. Fig. 6 shows the parameter estimation and structure learning results obtained using variational approximations for a 15 node network with maximum number of parents restricted to 2. The results indicate that the few edges that were missed by the structure learning algorithm (circles) correspond to weak dependencies i.e., edges with weights close to zero. The results of the Laplace approximation is similar to the variational approximation except that the variational approximation had higher confidence in its estimate.

Approximate Inference of rates The approximate inference techniques developed in Sec. 5 was tested on a random graph having 10 nodes. Samples were generated from the network using our sampling technique. Nodes were chosen at random and marked as observed, hidden and as query nodes. The inference task was to estimate rates for all the nodes marked as query nodes. The samples generated for the observed nodes were made use of to perform inference of rates using the sampling based approximation and the fixed point approximation. The results shown in Fig. 7 shows that the fixed point approximation technique (which is significantly faster than the sampling based approximation) closely tracks the true rate.

7 Conclusions and Discussion

Poisson networks are models of structured multivariate point processes and have the potential to be applied in many fields. They are designed such that sampling and approximate learning and inference are tractable using the approaches described above. In particular, structure learning is carried out in a principled yet efficient Bayesian framework.

Future applications of the Poisson network model include biological problems such as analysing multiple neural spike train data Brown et al. [2004] as well as application in computer science such as prediction of file access patterns, network failure analysis and queuing networks.

A promising direction for future research is to combine Poisson networks with continuous time Bayesian networks in order to be able to model both event counts and state (transitions). For example, consider file access events and the running states of CPU processes. Clearly, they exhibit an interesting relationship the discovery of which would it possible to predict and

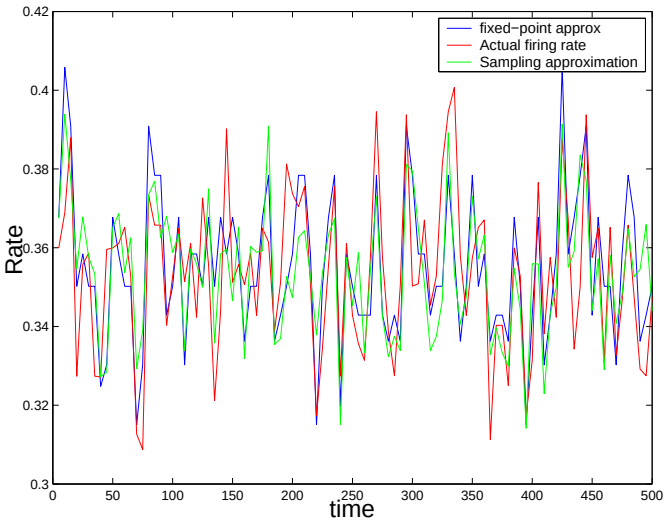

Figure 7: Inference of rate functions

hence optimise process-file interactions. Similarly, in biological application external stimuli can be modelled as states influencing physiological events such as neural spike activity.

Finally, it would be of great interest to study the information processing potential of Poisson networks in the sense of artificial neural networks (see Barber [2002]).

Acknowledgments Shyamsundar would like to thank Microsoft Research for providing funding for this project during an internship in Cambridge and Thoams Huang for generous support. We are very indebted to Ken Harris for interesting discussions about biological applications of Poisson networks, and we would like to thank Tom Minka for interesting discussions about approximate inference.

References

A. M. Aertsen, G. L. Gerstein, M. K. Habib, and G. Palm. Dynamics of neuronal firing correlation: modulation of "effective connectivity". *Journal of Neurophysiology*, 61(5):900–917, 1989.

D. Barber. Learning in spiking neural assemblies. In *Advances in Neural Information Processing Systems*, pages 165–172, 2002.

R. Barbieri, M. C. Quirk, L. M. Frankb, M. A. Wilson, and E. N. Browna. Construction and analysis of non-poisson stimulus-response models of neural spiking activity. *Journal of Neuroscience Methods*, 105(1):25–37, 2001.

D. R. Brillinger and A. E. P. Villa. Examples of the investigation of neural information processing by point process analysis. *Advanced Methods of Physiological System Modelling*, 3:111–127, 1994.

E. N. Brown, R. E. Kass, and P. P. Mitra. Multiple neural spike train data analysis: state-of-the-art and future challenges. *Nature Neuroscience*, 7:456–461, April 2004.

T. Cormen, C. Leiserson, R. Rivest, and C. Stein. *Introduction to Algorithms, Second Edition*. MIT Press, 2001.

T. Dean and K. Kanazawa. A model for reasoning about persistence and causation. *Computational Intelligence*, 5:142–150, 1989.

K. D. Harris, J. Csicsvari, H. Hirase, G. Dragoi, and G. Buzsaki. Organization of cell assemblies in the hippocampus. *Nature*, 424:552–556, July 2003.

M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.

R. E. Kass and A. E. Raftery. Bayes factors. *Journal of the American Statistical Association*, 90:773–795, 1995.

K. P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, UC Berkeley, Computer Science Division, 2001.

U. Nodelman, C. Shelton, and D. Koller. Continuous time Bayesian networks. In *Proceedings of the 18th Annual Conference on Uncertainty in Artificial Intelligence (UAI-02)*, pages 378–387, San Francisco, CA, 2002. Morgan Kaufmann Publishers.

U. Nodelman, C. Shelton, and D. Koller. Learning continuous time Bayesian networks. In *Proceedings of the 19th Annual Conference on Uncertainty in Artificial Intelligence (UAI-03)*, pages 451–458, San Francisco, CA, 2003. Morgan Kaufmann Publishers.

M. W. Oram, M. C. Wiener, R. Lestienne, and B. J. Richmond. Stochastic nature of precisely timed spike patterns in visual system neuronal responses. *Journal of Neurophysiology*, 81(6):3021–3033, 1999.

A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. Mcgraw-Hill, New York, 3 edition, 1991.