Discovering Latent Structure in Point Process Data

a dissertation presented
by
Scott Warren Linderman
to
The School of Engineering and Applied Sciences

in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy
in the subject of
Computer Science

Harvard University
Cambridge, Massachusetts
May 2016

Thesis advisors: Professors Leslie G. Valiant and Ryan P. Adams Scott Warren Linderman

Discovering Latent Structure in Point Process Data

Abstract

Point processes are probability distributions over sets of discrete events, like the timestamps of user activity on a social network, the locations of stars in the night sky, or the spatiotemporal pattern of spiking activity in the human brain. With applications across a host of scientific domains, point processes are a fundamental tool for modeling complex phenomena. Our aim in this thesis is to provide a collection of tools for discovering simplifying structure underlying point process data.

We focus on the Poisson process and its generalizations, the linear and nonlinear Hawkes processes. These processes are characterized by a non-negative rate function that specifies the likelihood of events in space or time. Often, our scientific objective is to provide a simple description, or model, of the rate function. The description may take the form of a parametric relationship between the instantaneous rate and a set of measurable covariates, like the time of day or the number of preceding events. Alternatively, we may characterize the rate function in terms of a latent and typically lower dimensional state. These models enable us to predict unseen events, and provide insight into the underlying processes that gave rise to our data.

As both the size of our data and the richness of our models grow, the need for scalable inference algorithms becomes paramount. To address this challenge, we develop a variety of data augmentation schemes to enable efficient Bayesian inference of the rate function and its parameters. Our schemes are amenable to Markov Chain Monte Carlo inference methods as well as variational approaches.

Our emphasis throughout will be on applications to computational neuroscience, where modern recording technologies enable us to measure the simultaneous activity of large populations of neurons, and our goal is to understand the computations and algorithms those populations implement. In the first six chapters, we will develop a suite of Bayesian methods that may shed light on the structure of neural spike train recordings and provide a bottom-up approach to understanding the brain. In closing, we will consider Bayesian inference as a candidate for a top-down theory of neural computation, and discuss potential implementations and testable ramifications of these hypotheses.

Contents

Listing of figures

This is the dedication.

Acknowledgments

...

0

Introduction

I will provide the context for this thesis in a brief introduction. I will touch on the goals of computational neuroscience and the current state of affairs in the field. I will make my pitch for the suite of machine learning methods I have developed for extracting patterns in neural recordings, and I will frame this as a step toward closing the gap between bottom-up neural data analysis and top-down theories of neural computation.

1 Background

In order to make sense of the rest of the thesis, some background is required. I think a section on probabilistic modeling will be helpful, but I would like to keep it short since there are already many good references on the subject. I will then introduce point processes, specifically the Poisson process, the Hawkes process, and the nonlinear Hawkes process, for which I have developed inference methods. Then I will introduce time series models, including the hidden Markov model (HMM), linear dynamical system (LDS), and the vector autoregressive (VAR) models. In subsequent chapters, I will show how these can be combined with Poisson process observations. Finally, I may discuss network models, since I have two chapters that draw on this material. Alternatively, I may defer the network models until the next chapter, where they are first used.

1.1 Probabilistic Modeling

1.1.1 Generative Graphical Models

1.1.2 Bayesian Inference

Markov Chain Monte Carlo

Variational Inference

1.1.3 Model Comparison

1.2 Point Processes

1.2.1 The Poisson Process

1.2.2 The Hawkes Process

1.2.3 The Nonlinear Hawkes Process

1.3 Time Series Models

1.3.1 Hidden Markov Models

1.3.2 Dynamical Systems Models

1.3.3 Autoregressive Models

1.4 Network Models

2

Hawkes Processes with Latent Network Structure

This chapter will draw on two papers, my 2014 ICML paper[1], which shows how to do Gibbs sampling in a continuous time model, and my arXiv preprint[2], which shows how to do Gibbs sampling and stochastic variational inference in a discrete time approximation. Both papers focus on adding prior distributions on the network of interactions, and showing that the latent variables of the network models can be meaningful, and that they can aid in prediction of future activity.

Hawkes processes are a particularly nice starting point because they capture a salient type of structure — excitatory interactions between nodes — while retaining some analytical tractability due to the linear form of the interactions. In addition to deriving inference algorithms, I will also show how we can tailor our prior distributions to maintain stability of the network.

3

Nonlinear Interaction Networks

Hawkes process inference relied on an augmentation strategy made possible by the linear form and excitatory nature of the interactions. In neural settings, these assumptions of the Hawkes process are not as realistic. Instead, we turn to the nonlinear Hawkes process, which allows for both excitatory and inhibitory interactions by introducing a nonlinearity into the model. This corresponds to the popular generalized linear model that is widely used in computational neuroscience. We develop a novel approach that leverages the Pólya-gamma augmentation to enable efficient, fully-conjugate inference. Again, we combine the nonlinear Hawkes process with prior distributions on the network of interactions, and we focus on a discrete time approximation. This is part of ongoing work with Ryan and Jonathan Pillow at Princeton, and it has grown out of a Cosyne abstract[3]. I also have an unpublished draft in which I extend the abstract and apply it to a variety of neural datasets, and show how the network models recover interesting latent features like neural types and locations. I will submit a journal version of this paper as soon as possible.

4

Dynamic Network Models

This chapter will extend the previous two chapters to model data with underlying networks that change over time according a learning rule. This work was originally presented in a 2014 NIPS paper with Chris Stock and Ryan Adams [4]. Our original paper focused specifically on modeling synaptic plasticity, but in this thesis I would like to frame it more broadly as a model for networks with dynamic weights. I also can introduce inference algorithms that are more efficient given the Pólya-gamma augmentation schemes I have since developed, and which were introduced in Chapter 3.

This is the abstract of our 2014 NIPS paper: Learning and memory in the brain are implemented by complex, time-varying changes in neural circuitry. The computational rules according to which synaptic weights change over time are the subject of much research, and are not precisely understood. Until recently, limitations in experimental methods have made it challenging to test hypotheses about synaptic plasticity on a large scale. However, as such data become available and these barriers are lifted, it becomes necessary to develop analysis techniques to validate plasticity models. Here, we present a highly extensible framework for modeling arbitrary synaptic plasticity rules on spike train data in populations of interconnected neurons. We treat synaptic weights as a (potentially nonlinear) dynamical system embedded in a fully-Bayesian generalized linear model (GLM). In addition, we provide an algorithm for inferring synaptic weight trajectories alongside the parameters of the GLM and of the learning rules. Using this method, we perform model comparison of two proposed variants of the well-known spike-timing-dependent plasticity (STDP) rule, where nonlinear effects play a substantial role. On synthetic data generated from the biophysical simulator NEURON, we show that we can recover the weight trajectories, the pattern of connectivity, and the underlying learning rules.

5

Nonparametric Hidden Markov Models for Neural Data

In joint work with Zhe Chen, Matt Johnson, and Matthew Wilson, I have been developing nonparametric hierarchical Dirichlet process hidden Markov models (HDP-HMMs) for neural spike train recordings and applying them to hippocampal data recorded in the Wilson lab. The advantage of these nonparametric methods is that the number of states is inferred in a data-driven way, allowing the complexity of the model to grow with that of the data. However, inference becomes more challenging. In this work, we have conducted a thorough empirical study that compares parametric and nonparametric methods, as well as a variety of inference algorithms and approaches to setting the hyperparameters (which can have a large effect on the subsequent inferences). A preliminary version of this work is on the arXiv[5], and we are submitting final revisions to the Journal of Neuroscience Methods this week.

6

Dynamical Systems with Discrete Observations

In the final chapter on modeling neural data, I will expand upon the Cosyne abstract that Aaron Tucker, Matt Johnson, and I submitted this year[6]. We show how the Pólya-gamma augmentation strategy can be leveraged to render the observations conjugate with linear Gaussian latent structure. This allows us to use off-the-shelf inference tools for Gaussian LDS models. It also allows us to extend to switching variants of the LDS by including another layer of discrete, Markovian states. While these models apply directly to Bernoulli and negative-binomial observations, we can approximate Poisson observations as a limit of the negative binomial. Alternatively, we can use a Poisson thinning model with multinomial conditionals to exactly simulate the Poisson model. If time and space allow, we will introduce this novel approach in this chapter. Finally, I will compare these models on the basis of their ability to generalize to held out spike counts. Since this is the final chapter, we will also provide a comparison against the models introduced in preceding chapters.

7

Bayesian Computation in Neural Circuits

The final chapter will take a different view on Bayesian computation. Rather than using probabilistic models and Bayesian inference to discover structure in neural recordings, we will consider the hypothesis that the neural circuits under study are actually implementing Bayesian computations. This "Bayesian brain" hypothesis is by no means novel — it has been the subject of much recent research and debate in the computational neuroscience community. Under this hypothesis, the brain uses a probabilistic model of the world, and implements Bayesian inference algorithms to infer latent or missing variables under noisy observations. The probabilistic model is learned through experience, supervised training, or evolution, though the mechanisms of learning are less thoroughly explored.

I think the starting point is a background of cognitive and neural evidence for and against these hypotheses, which will lead naturally into open questions. While the work for this chapter is not yet complete, I think there are two avenues that I could pursue.

First, I have been working with Ishita Dasgupta and Sam Gershman on an idea for data-driven "proposal" mechanisms that use the observed inputs to instantiate hypotheses about the latent variables. These hypotheses are then simulated in order to evaluate their consistency with the observed data, and weighted accordingly. This is an example of importance sampling, but with an intelligent proposal. The proposal is, in turn, learned through experience using stochastic gradient descent. We have implemented a number of simulations to show that this works in theory, at least on a variety of toy problems. For this thesis, it would be interesting to tie the results of these simulations to real-world cognitive science experiments. Predictions about neuronal implementations will be necessarily speculative, but we can make some high level arguments about layered implementations (e.g. that the proposal may be carried by long range feed-forward connections, whereas the local recurrent connections likely implement the simulation and weighting code). This could lead to some qualitative predictions about the effects of lesions. Finally, it may be possible to quantify the effect of a proposal network in terms of the variance of the importance sampling estimator.

A second potential direction is to consider the effect of limited connectivity on the convergence of a Gibbs sampling algorithm. Many Bayesian theories of neural computation postulate that the brain is running a Markov chain to sample from the posterior distribution over latent variable states. Hypothesized implementations assume that neurons represent single variables, and that the neurons are fully connected so that they may update their states given the states of all others. In practice, neural connectivity is sparse, and neural representations are likely distributed. Perhaps we can quantify the effects of sparsity and distributed representation size in a simple, jointly Gaussian model. In this setting, we can derive analytical forms for the updates and compare the stationary distribution of the approximate model to the true posterior. When the representation is not distributed (one neuron per variable) and the neurons are fully connected, the model reduces to Gibbs sampling with exact updates. As the degree of connectivity goes to zero, presumably the representations need to become more distributed in order to approximate the true posterior.

8

Conclusion

Here I will suggest some future directions. The sequence of models we built up in this thesis are a progression toward models of neural "programs." Already, our switching linear dynamical systems allow us to express complex, nonlinear dynamics. By conditioning upon outside stimuli, these models looks more and more like closed loop programs. In future work, I would like to make this connection concrete by showing how simple classes of probabilistic programs can be instantiated as a prior distribution on patterns of neural activity, and how they may be inferred from actual neural data. Example classes of programs may be those that implement simple forms of Bayesian inference, or those that attempt to maximize a reward function. Certainly our first examples will be simple, but this is clearly the direction in which our methods are pointing.

A
Pólya-gamma Augmentation

Since the Pólya-gamma augmentation scheme is relatively new, I will provide a detailed analysis of this distribution and its relevant properties, and show why it works well for inference in Gaussian models with discrete observations.

References

[1] Scott Linderman and Ryan Adams. Discovering latent network structure in point process data. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1413–1421, 2014.

[2] Scott W Linderman and Ryan P Adams. Scalable bayesian inference for excitatory point process networks. *arXiv preprint arXiv:1507.03228*, 2015.

[3] Scott W Linderman, Ryan P Adams, and Jonathan W Pillow. Inferring structured connectivity from spike trains under negative-binomial generalized linear models. *Computational and Systems Neuroscience (Cosyne) Abstracts*, 2015.

[4] Scott Linderman, Christopher H Stock, and Ryan P Adams. A framework for studying synaptic plasticity with neural spike train data. In *Advances in Neural Information Processing Systems*, pages 2330–2338, 2014.

[5] Scott W Linderman, Matthew J Johnson, Matthew A Wilson, and Zhe Chen. A nonparametric bayesian approach to uncovering rat hippocampal population codes during spatial navigation. *arXiv preprint arXiv:1411.7706*, 2014.

[6] Scott W Linderman, Aaron Tucker, and Matthew J Johnson. Bayesian latent state space models of neural activity. *Submitted to Computational and Systems Neuroscience (Cosyne) Abstracts*, 2016.