Learning the Structure of Local Neural Circuits in Mouse Ectorhinal Cortex

Scott Brookes and Derek Racine

May 19, 2014

Introduction

The brain is structured such that local networks of neurons can perform distinct, modular, tasks. For example, the ectorhinal cortex is known to receive the majority of its input from visual sensory areas and is involved in visual memory and object recognition. However, the network structure is not well understood. Training an animal to recognize certain images, such as through visual paired-associate learning, might also affect the network dynamics. In particular, one might hypothesize that this training could cause neurons to become tuned to images that the mouse has seen before, but not others. That is, the activity of neurons would be well correlated with stimuli used during training. Yet, this relationship hasn't been explored.

Simplified neurons can be described as being in one of two states: resting or firing. When a neuron fires, calcium ions flow into it from the extracellular fluid. As such, fluorescence observed from calcium sensors can serve as a proxy for neural activity.

The recent discovery of an ultrasensitive family of fluorescent calcium sensors, GCaMP6, has caused an explosion in the availability of time-series data based on the use of this technique.[1] Whereas previous recording methods were limited to only a few cells, fluorescent imaging techniques can measure the activity of an entire region of neurons simultaneously, as shown in Figure 1. This data offers an opportunity to learn the structure of local neural circuits through statistical methods. Indeed, previous work has addressed this problem by modeling fluorescent imaging data as a collection of coupled Hidden Markov chains, one for each neuron.[2]

Dataset

We have obtained data sets from the Max Planck Institute for Neurobiology corresponding to four recording sessions from the same mouse. Each recording session consists of multiple time series of fluorescent activity corresponding to the neurons within the imaged area. GCaMP6s, a slower decaying member of the GCaMP6 family, was used as the calcium sensor in these experiments. The first data set consists of activity of neurons in the primary visual cortex (V1) that are known to reliably respond to oriented gratings at a certain angle, which are presented to the mouse during the trial. An illustration of this setup is shown in Figure 2. This data will be a useful control for testing our structure learning method because we expect that there will be a strong correlation between neural activity and the display of specific stimuli.

The remaining data sets track the activity of neurons in the ectorhinal cortex during periods of spontaneous activity, the display of novel stimuli, and the display of these same stimuli after learning to do a visual

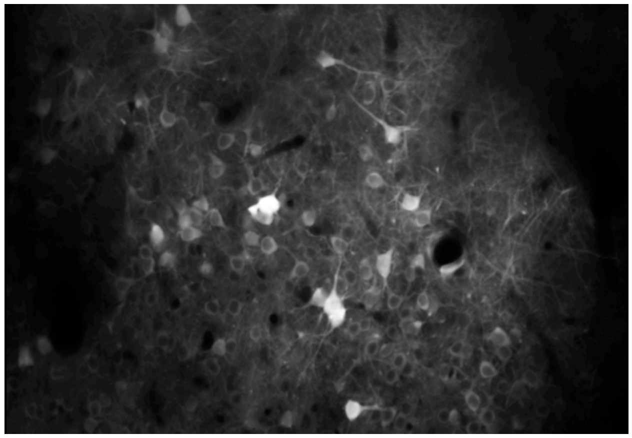


Figure 1: A snapshot of neurons labeled with GCaMP6s within a region of the cortex.

paired-associate task with a subset of them. Inferring both network structure and edge weights for these time series will allow us to determine whether neurons become tuned to stimuli that the mouse has learned to dscriminate. We also plan on investigating the difference between the correlation neurons exhibit during spontaneous activity versus when a stimulus is present.

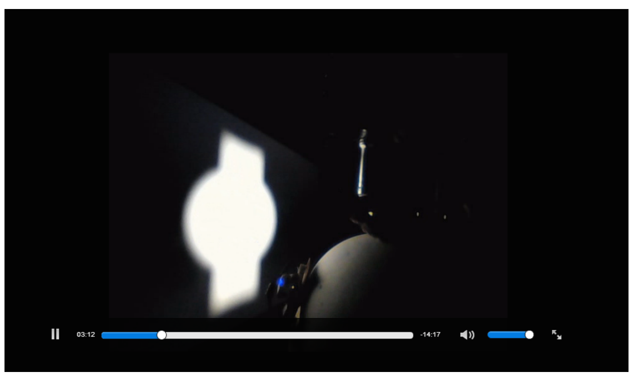


Figure 2: A photo of the experimental setup in which the mouse is simultaneously viewing a screen where stimuli are presented and being recorded from.

Method

We propose to model the local neural circuits as a Bayesian network and infer the edges from the correlations of the activity of individual neurons, which we will treat as binary random variables firing or not firing. In order to learn the structure, we plan to discretize our continuous series into buckets that each represent a single instance of the overall network.

We are using the GlobalMIT toolbox for learning the globaly optimal Dynamic Bayes Net structure.[3] The toolbox uses the recently introduced *mutual information test (MIT)*. It is implemented in both Matlab® and C++.

The Mutual Information Test

We will present just a brief overview of this method. For a full discussion of this algorithm, we direct an interested reader to.[4]

A potential network is scored by the total mutual information shared between each node and its parents, less a term that quatifies how statistically significant the shared information is. The full discussion shows that the first term, considering the total mutual information shared between each node and its parents, is equivalent to maximizing the log-likelihood criterion. As we know, however, learning a Bayes Net using maximum log-likelihood alone risks overfitting the data. In order to avoid this unnecessary complexity, the MIT method considers whether the information gained by adding a parent to a node is statistically significant. This effectively penalizes a complex model.

Using the Toolkit

Using the toolkit is rather simple. In order to illustrate the power of the toolkit, we give two toy examples that illustrate the output of the toolkit given some input. For the output shown in Figure 5a, the input is a data matrix shown in Figure 3 where that table has different timeslices as columns and different variables as rows. The same format is used to display the data table shown in Figure 4 used for the toy 2 example shown in Figure 5b.

1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1

Figure 3: Toy1 Dataset

1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1

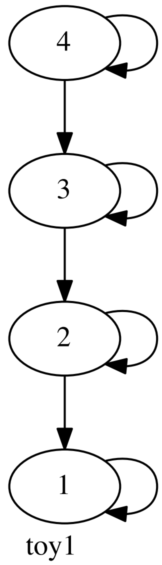
Figure 4: Toy2 Dataset

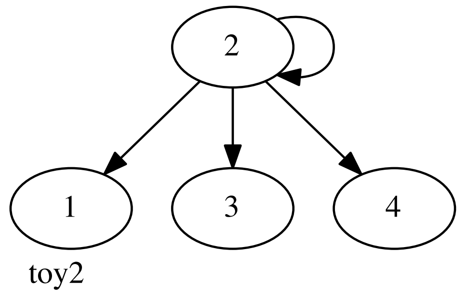
Implementation Considerations

The toolkit takes a parameter $\alpha : 0 < \alpha \leq 1$ that tunes the rigor to which the algorithm will try to fit the model. Lower α's allow for a lower degree of uncertainty, so a lower α generates a more accurate result but also requires a great deal more computation time.

In order to model the entire neural network that we are interested in, we will need to make provisions for minimizing the complexity. The paper[3] demonstrates a problem in which there are 20 variables and 2,000 observations. In the Matlab® version of the toolkit this model takes more than a full day to be learned. Our dataset, naively, has as many as 100 variables with 24,000 observations.

In order to achieve practical computation times, we will have to preform data preprocessing discussed in other sections of this paper including bucketing to get the number of observations down. In addition, the paper discusses that the C++ implementation of the toolkit takes only an hour to process this same





(b) Toy2 Example Output

(a) Toy1 Example Output

Figure 5: Toy Example Outputs

example, leading us to believe that we may need to move towards the C++ implementation. This should not require too much effort once we familiarize ourselves with the tools on the Matlab® side.

Measuring Success

We expect that the structure we learn from the first data set will reflect the high correlation between neural activity and the presence of particular stimuli in V1. After we have verified that this is indeed the case, we will apply the same method to the other data sets in order to learn the network structure of the neural circuit in the ectorhinal cortex.

Challenges

Data wrangling has been an enormous time sink. The time series activity is extremely noisy, varies independently for each neuron, and includes observations when the shutter is closed. As such, the data requires significant preprocessing in addition to hand-tweaking, for example, the window size considered for smoothing and the threshold for determining when a neuron should be considered firing.

Structure learning was also complicated by the fact that the data is sparse; neurons fire very infrequently throughout the training session.

References

[1] Tsai-Wen Chen and et al. Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature*, 499.7458:295–300, 2013.

[2] Yuriy Mishchenko, Joshua T. Vogelstein, and Liam Paninski. A bayesian approach for inferring neuronal connectivity from calcium fluorescent imaging data. *The Annals of Applied Statistics*, 5.2B:1229–1261, 2011.

[3] N. X. Vinh, Coppel R. Chetty, M., and P. P. Wangikar. Globalmit: Learning globally optimal dynamic bayesian network with the mutual information test (mit) criterion doi:10.1093/bioinformatics/btr457, 2011.

[4] N. X. Vinh, Coppel R. Chetty, M., and P. P. Wangikar. Polynomial time algorithm for learning globally optimal dynamic bayesian network. *Neural Information Processing*, pages 719–729, 2011.