

MA 568 – Statistical Analysis of Point Process Data

Problem Set #3: Due November, 6, 2018

In class we discussed constructing generalized linear models for point process data as a function of three distinct classes of variables: 1) time varying signals that influence the intensity, 2) the history of the point process being modeled, and 3) the history of other point processes that influence the intensity of one being modeled. In this problem we will analyze the firing properties of a place cell in rat hippocampus as a function of these classes of variables. Recall that the firing activity of these cells is known to relate to the animal's current location during a free foraging task.

1. Please download the **hipp_data.mat** (or **hipp_data.csv**) file from the course website. The data consists of the spiking activity of two neurons recorded simultaneously from the CA1 region of a rat's hippocampus as it performed a free-foraging task. Loading this dataset into MATLAB (or R) creates the following variables:

T - Time index in seconds.
xN, yN - Normalized (x,y) coordinates of the animal at each time.
spikes - Spike times for the neuron to be modeled in 1 ms bins.
spikes2 - Spike times for a second neuron in 1 ms bins.
spikes_hist - History matrix for neuron to be modeled
spikes2_hist - History matrix for second neuron

Plot the animal's movement trajectory with its position at the spike times for each neuron overlaid. What can you say about the spatial firing properties of these two neurons?

The MATLAB function **glmfit** can be used to rapidly compute maximum likelihood estimates, confidence bounds, and p-values for simple GLM models. For point process data, binned at a fine time scale, the call to **glmfit** is as follows:

```
>>[b,dev,stats] = glmfit(X,spikes,'poisson');
```

where **X** represents a matrix of your GLM covariates at each time, and **spikes** is the spike train time series which is being modeled. For example, to construct an intensity model,

$$\lambda(t) = \exp\{\beta_0 + \beta_1 \mathbf{xN}(t) + \beta_2 \mathbf{yN}(t)\}, \quad (1)$$

whose logarithm is linear in x and y position, we can make the following call:

```
>>[b,dev,stats] = glmfit([xN yN],spikes,'poisson');
```

The returned vector **b** is the maximum likelihood estimate of the model parameters, **dev** is the model deviance, defined as $\text{dev} = -2\log L(\theta)$, and **stats** is a structure containing many useful quantities including **stats.se**, the standard error of each parameter and **stats.p** their p-values.

In R, the comparable command is **glm**, which could be called as follows:

```
> modelfit <- glm(spikes ~ X, 'poisson')
```

The model components can then be assessed using the **summary** command.

2. Use the above GLM command to fit the exponential linear model in equation (1) to the **spikes** data. Are all of the resulting parameter estimates significant? Plot the maximum likelihood model intensity as a function of (x,y) position. How well does this capture the spatial firing properties of the neuron?

Hint: The script **hipp_glm.m**, provided on the class website, performs the fitting and visualization for this exponential linear position model. Feel free to use and modify this script as you please.

3. One way to improve on this exponential linear model is to add quadratic functions of **xN** and **yN**. Write down and fit a GLM model containing all terms of quadratic and smaller order. Are all of the maximum likelihood estimates for this model significant? Plot the maximum likelihood model intensity as a function of (x,y) position. Does this improve the description of the spatial firing properties of the neuron? Compute the AIC ($\text{dev} + 2 \times (\text{number of parameters})$) values for the exponentiated linear and quadratic models. Which model describes the data better?

4. The **spikes_hist** matrix contains the spiking history at each point in time in 1 ms bins going back 20 ms. **spikes_hist(:,1:p)** would contain the history going back p ms. For p = 0 to 20, fit the following GLM model:

$$\lambda(t | H) = \lambda^S(\mathbf{xN}(t), \mathbf{yN}(t)) \exp \left\{ \sum_{i=1}^p \beta_i \mathbf{spikes_hist}(:, i) \right\},$$

where $\lambda^S(\mathbf{xN}(t), \mathbf{yN}(t))$ is the quadratic spatial model from question 3. Plot the AIC as a function of p. What is the optimal model order?

5. Augment your GLM model from question 4 to include the past spiking history of the other recorded neuron. Are any of the parameters associated with these spiking interactions significant? Which of your models (of all you have fit) is most parsimonious?

6. Assume we didn't know that these neurons were tuned to position. Fit a GLM model with only history and network interaction components. Are the parameters related to the network interactions now significant? Explain why the interaction terms might be significant only when we fail to model the spatial component of the firing activity.

7. For the most parsimonious model you have constructed, construct a KS plot based on time rescaling of the interspike intervals. How well does this model capture the observed spiking activity.