STA 4103/5107: Midterm Project

Neural Decoding in Motor Cortex

(Wednesday, 02/18)
Due: noon, Friday, 03/06

In this project, we will conduct neural decoding in motor cortex using two important inference methods in dynamic systems. Method 1: classical Kalman filter, Method 2: inhomogeneous Poisson Process. Both methods are widely used in computational statistics.

Your report should be in the standard form of a scientific report, which includes the following sections:
(1) Abstract
(2) Introduction (the problem statement)
(3) Data Description
(4) Computational Methods
(5) Results
(6) Discussion and Summary (comparison of the two methods in decoding accuracy)
(7) References (optional)

Points are mainly allocated towards presentation in your report!

1. **Goal:** Using observed neural activity from brain cortex in research animals, we will perform statistical analysis on this data with several models to understand the brain mechanism and make inferences about the external behaviors.

2. **Experiment:** Download the datasets (midterm_train.mat and midterm_test.mat) from the class website. Each set has two variables: *kin* and *rate*. *kin* is the kinematic state of the hand which includes *x-position*, *y-position*, *x-velocity*, and *y-velocity*. *rate* is the spiking rates of 42 neurons, where the rate at each time is the number of spikes within 70*ms*. Use the *kin* and *rate* in the training data to identify the model. In the test data, use the identified model and *rate* to infer *kin*, and compare the estimate with the true *kin*. In this project, the comparison only needs to focus on the hand positions, which are the first two components in *kin*.

Method 1: Kalman Filter Model

1. **Kalman Filter Model:** Let x_t in $\mathbf{R}^4$ denote [x-position, y-position, x-velocity, y-velocity] of a 2-d hand movement at time t, and y_t in $\mathbf{R}^c$ denote the spiking rate of c neurons in the primary motor cortex at the same time. A classical Kalman filter is used to model the hand kinematic state and neural activity as follows:

$$x_{t+1} = Ax_t + w_t$$
$$y_t = Hx_t + q_t$$

where $w_t \sim N(0,W), q_t \sim N(0,Q)$.

2. **Model Identification:** In the training set, both hand state and neural activity are known. Use the close-form formula to estimate the model parameters A, H, W, Q.

3. **Neural Decoding:** Once the parameters are identified, we can perform neural decoding on the test data. That is, we will use neural activity to infer the movement behaviors of the hand. Two inference methods need to be used here:

    a. **Kalman Filter Algorithm:** Use a Kalman filter to estimate the hand movement. In particular, plot the true and estimated hand positions (the first two components in the state). Compute the estimation accuracy of the positions using R^2 Error.

    b. **Sequential Monte Carlo Method:** Based on the same Kalman filter model, estimate the hand positions using a sequential Monte-Carlo method. Let the number of samples *n* be 20, 50, 100, and 500. For each *n*, plot the true and estimated states. Compute the estimation accuracy using R^2 Error. Plot the accuracy as a function of *n* and compare it with the accuracy in the Kalman filter algorithm.

Method 2: Inhomogeneous Poisson Model

1. **Inhomogeneous Poisson Process Model:** Let $x_k = [p_{x,k}, p_{y,k}, v_{x,k}, v_{y,k}]^T$ in $\mathbf{R}^4$ denote (x-position, y-position, x-velocity, y-velocity) of a 2-d hand movement at time t_k, and $y_k = \{y_{k,c}\}$ in $\mathbf{R}^C$ denote the spiking rate of C neurons in the primary motor cortex at the same time.

    a. For $y_{k,c}$, $c = 1, \ldots, C$, we assume a generalized linear model (GLM) with an inhomogeneous Poisson process condition on x_k. That is,

    $$y_{k,c} \sim Poisson(\lambda_{k,c})$$

    where $\lambda_{k,c} = \exp(\mu_c + \alpha_{1,c} p_{x,k} + \alpha_{2,c} p_{y,k} + \alpha_{3,c} v_{x,k} + \alpha_{4,c} v_{y,k})$.

    b. For x_k, we assume a simple linear Gaussian model over time. That is,

    $$x_k = A x_{k-1} + w_k,$$

    where $w_k \sim N(0, W)$.

2. **Model Identification:** In the training set, both hand state and neural activity are known. Use the close-form formula to estimate the model parameters A, W (as in a Kalman filter model). Note that the kinematic data needs to be centralized before the model is fit. Use the MLE method to estimate parameters $\{\mu_c, \alpha_{1,c}, \alpha_{2,c}, \alpha_{3,c}, \alpha_{4,c}\}$, $c = 1, \ldots, C$.

3. **Neural Decoding:** Once the parameters are identified, we can perform neural decoding on the testing data. That is, we will use neural activity to infer the movement behaviors of the hand. Two inference methods need to be used here:

a. **Point Process Filter:** Use a point process filter to estimate the hand movement. Plot the true and estimated hand position and velocity (one subplot for each component in x_k). Compute the estimation accuracy for each component using R^2 Error. Compare the result with the Kalman filter model.

b. **Sequential Monte Carlo Method:** Based on the same model, estimate the hand positions using a sequential Monte-Carlo method. Let the number of samples *n* be 20, 50, 100, and 500. For each *n*, plot the true and estimated states. Compute the estimation accuracy in each component using R^2 Error. Plot the accuracy as a function of *n* and compare it with the accuracy in the point process filter.