**Homework 5**

Due 5/30/2020

**Problem 1:**

Load ‘hw5\_pca.mat’. You should find a variable called ‘D’ which is a data matrix. This is a simulation of an intracellular recording. You are trying to determine (1) how many different presynaptic cells are providing inputs to the cell you are recording, and (2) what are the release probabilities of each of the input synapses. Your data consists of 10^4 trials (rows of D) in which you provided a stimulus pulse to the presynaptic nerves at time 0. The columns of D are 1000 time samples at a sampling rate of 10kHz, some of which contain EPSPs. The units in D are mV in your recorded neuron. Throughout the problem, **please remember to label all axes in graphs with meaningful units.**

**a**) Plot a heat map of D. Find one trial in which there appears to be no EPSP and two trials in which you do see a response. Plot each of those 3 trials individually using ‘subplot’ and label them.

**b**) Make the covariance matrix for D. What is its size? State concisely the meaning of the element at position i,j in the covariance matrix. Plot the covariance matrix (remember to include a color bar and label for it). What is the difference between positive and negative values in the covariance matrix? What insights can you glean from the pattern of the positive vs. negative covariance values?

**c**) Run principal components analysis on D. Plot the percent variance explained by each PC. How many PCs should you use to represent the data? Why?

**d**) Plot each of the PCs you chose. Do these look like reasonable waveforms for EPSPs?

**e**) The ‘score’ output of matlab’s PCA function provides a shortcut to perform the projection step (projecting each data vector along each of the PCs). Make histograms of all the projections values along the first N PCs you chose in part 3. Make a histogram of the projections of the data along PC N+1. Do these histograms confirm that you made the correct choice for the number of PCs to use? Why?

**f**) Make scatter plots (2D or 3D) of the projection values of the data along all possible combinations of the N PCs you chose in part 3. How many clusters do you see in these scatter plots?

**g**) Use ‘kmeans’ to cluster the data using ‘k’ from part f. Plot the mean trace for the elements of D in each cluster with error bars for the standard error of the mean.

**h**) How many different presynaptic inputs are there to your neuron? Which clusters correspond to the individual inputs? What does each of the other clusters represent? Do you now have an explanation for the shapes of the PCs you found in part 4?

**i**) What is the event rate of each of the synapses?

**Problem 2:**

Load ‘HW5\_machineLearning.mat’. In this file you will find real transcriptomics data from my lab. The goal of this problem is to find whether a set of genes can predict whether a retinal ganglion cell (RGC) is from the left or right eye. ‘transcriptomics\_data’ is a matrix of gene expression. Each row is a RGC and each column is the logarithmically normalized expression value of a gene. There are only 1133 genes because I have removed the ones with low or inconsistent expression in this population of cells. ‘geneNames’ is a cell array of 1133 gene names, and ‘whichEye’ is a cell array with a ‘R’ or ‘L’ corresponding to whether each RGC came from a right or left eye.

**a**) Separate the data into two separate matrices for cells from the L and R eyes. How many cells are there from each eye?

**b**) Compute the mean expression value of each gene in cells from the R eye and do the same in the L eye. Now compute the expression ratio for each gene as mean(R) ./ mean(L). Store these variables in the workspace, but no need to print them out since they are length 1133!

**c**) Plot a histogram of the expression ratio values you obtained in b. What does a value of ‘1’ mean? What does a very low value mean? A very high value?

**d**) Fit the expression ratio vector as a Normal distribution. What is the mean and standard deviation?

**e**) From this fit to the distribution, estimate the probability of a value being > 2. Now estimate the probability of a value being < 0.5.

**f**) Now find the genes with expression ratios > 2, and separately, find those with values < 0.5. List the gene names in each category. Make a bar graph plotting the expression ratio for each of these genes, with the gene name on the X axis. Label axes correctly. Based on your answer to part e, what statement can you make about these genes?

**g**) Make a Matlab table containing the full data set and the ‘whichEye’ vector.

Use the Classification Learner App to load in this table. How many variables are in your learner? How many classes are in your predictor?

**h**) Train a classifier on the data set to test if a cell is from the L or R eye. Use 10-fold cross validation. Which type of classifier performed the best? What is the error rate? What is chance on this task? Given the L:R imbalance in the data, what would the performance be if a trivial classifier just chose L every time? Does your classifier achieve good performance above this baseline?

**i**) Now let’s restrict the data set by choosing differentially expressed genes to start with. Define a relative expression threshold of 1.3. Find all the genes with an expression ratio > 1.3 or with a ratio < 1/1.3 (which is .7692). How many genes is this? Remake your data table using only these genes and train the classifier again. Which is the best classifier this time? How well did it perform? Can you explain why this classifier did better than the one in part h even though it had fewer variables?

**j**) For the classifier in part i, what is the chance your classifier will incorrectly label a cell from the L eye as being from the R eye? What is the chance it will incorrectly label a cell from the R eye as being from the L eye?

**Bonus:**

**k**) Try different values of the threshold from part i to determine which has the best performance.