NB204 Principal component analysis homework instructions

Your homework this week consists of two parts, a reading assignment and a coding assignment:

1. Read pages 1-3 of *Stephens, Greg J., et al. "Dimensionality and dynamics in the behavior of C. elegans." PLoS Comput Biol 4.4 (2008): e1000028".* You can stop before the "Attractors and Behavioral States" section. This paper provides an example of using PCA to help understand behavioral data. We will discuss this paper in class on Monday and hope it will help you better understand the coding assignment. However, you do not need to provide any written summary/critique of this paper. We advise you to do the reading assignment before starting the coding assignment, because the ideas in the paper should help you understand the motivation for the coding exercise.

2. For the coding part of the assignment, we would like you to perform principal component analysis on some simulated neural data. You are provided with the responses of 58 simulated neurons to a set of stimuli consisting of steps and ramps of stimulus intensity. We provide the responses to these stimuli as peristimulus time histograms (PSTHs). Your goal is to use PCA to reduce the dimensionality of the data from 58 neurons to a smaller number of 'principal component' neurons which capture most of the variance of the original data. You will do this by completing some code that we have started for you. Then, please submit answers the problems below. Your write-up may be longer than two pages because we ask you to include several figures. However, your narrative answer to each numbered point should only be a few sentences long. Do not submit your code – please only submit answers to the problems.

3. Students who already have some background in this area are urged to tackle some of the optional "extension problems" listed at the end of this assignment. We urge you to challenge yourself. If you put more into the assignment, you will get more out of it.

Instructions for coding assignment

You are provided with three files:

- **pca_neurons.m** This is the main script for performing PCA on the neurons and producing figures of your results. There are detailed instructions of how to complete the code provided in the file. As indicated you will need to complete code in places marked "Insert/Modify code here". There are examples of how to load the data, plot and save figures.
- **pca_data.mat** contains the data you will perform pca on. A description of the data format is provided in the pca_neurons.m script.
- **saveFormattedFig.m** is a function we wrote to make it easy for you to save figures in a format that you can insert straight into word document. You shouldn't need to change anything in this function.

Getting started

- Make sure that you have MATLAB 2016a or a more recent version (typing the command 'version' into the command window will tell you the version you have). You can download the latest version of MATLAB by following the instructions on this webpage: http://downloads.fas.harvard.edu/download Doing this on the Harvard Network should be fast. You can make it faster by unchecking the option to download Simulink.
- Make sure you have the Statistics and Machine Learning Toolbox. You can add the toolbox to your current installation of MATLAB by following the instructions here. (If you want to check whether you have the toolbox already, type 'ver' into the command line.)
- Download the above three files from the wiki, they will be zipped together as nb204_pca.zip. Unzip the file and make sure they are in a folder called 'nb204_pca'.
- Open up MATLAB.
- Set the current folder to 'nb204_pca'.
- Open pca_neurons.m
- Read the description at the beginning of the script and then run the 'Load data', 'Plot data...' and 'Save figure' parts of the code and make sure you understand them.
- Take a look at the data that is plotted.

- Complete the rest of the code.
- Write and submit responses to the problems listed below, inserting specific figures as requested in your text document. Number each section of your assignment as indicated. Make sure to include legends and axis labels for your figures so that we know what the figures contain, especially in the optional "extension problems". Please submit your homework as a .doc or .docx file, as usual.

Problems

1. Show a plot of the data (firing rates versus time) for the first six neurons. In a few sentences, describe the properties of these neurons. What part(s) of the stimulus does each neuron respond to?
2. Show a plot of the covariance matrix for the entire data set. Describe the structure of the covariance matrix. What does this structure mean?
3. Show a plot of the percentage of the variance in the data explained by each PC, versus PC number; this is sometimes called a "scree plot". Take an educated guess: how many PCs account for a statistically significant percentage of the variance in the data? We are asking you to make an estimate based on looking at the scree plot. Please explain your reasoning. What do the non-significant PCs represent?
4. What might be a principled way to determine the number of PCs that actually <u>are</u> statistically significant? (Hint: think about how you might be able to create a fake data set that embodies a useful "null hypothesis" that you could compare your data set to. It is not necessary for you to actually implement this suggestion; we just want you to describe what you would do in a clear and detailed manner.)
5. Show a plot of the first few PC scores versus time. Describe each of these PC scores. Keep in mind that each one should be a linear combination of neurons, and so we can think of each as a sort of idealized neuron (to borrow a usage from Stephens et al., we might call each them "eigen-neurons"). What part(s) of the stimulus does each "eigen-neuron" correspond to? Try to give each one a categorical descriptive label.
6. Show a plot of the covariance matrix of the PC scores. Briefly describe this matrix. What can you conclude from the appearance of this matrix?
7. Show a 3D plot of each neuron's loadings onto the first three PCs. Do different neurons seem to cluster in any way? What does this structure mean? (Hint: look back at the PC scores and think about your descriptive labels for each one.)

Extension problems (optional)

8. The first step in performing PCA is almost always to *standardize* the data – i.e., to make different samples more comparable to each other by making them conform to some standard format. The MATLAB 'pca' function centers the data (by subtracting the mean), but it doesn't perform any additional standardization before performing PCA. Sometimes, however, we might want to perform an additional layer of standardization by scaling each vector by its own standard deviation. (In other words, we would be z-scoring the data before performing PCA.) Here, this would amount to dividing each neuron's response by its own standard deviation (computed across all time points). Briefly describe the pros and cons of z-scoring in the particular case of the data set you are working with in this exercise. Can you think of a case in neuroscience where z-scoring is really important – i.e., a case where the results of PCA are pretty meaningless unless you z-score before performing PCA?
9. Implement the proposal you made in problem (4) above to determine the number of PCs that are statistically significant. Describe and show the result. Was the result what you expected?
10. What happens if you only use one of the "ramping" portions of the stimulus response to calculate the PCs? Why might this be?
11. We have included a short paper by Jonathon Shlens that outlines the steps required to perform PCA using concepts from linear algebra.
    a. Use the hints provided by this paper to calculate the covariance matrix without using the 'cov' function in matlab. Paste this section of code in your report.
    b. Next, find the eigenvalues and eigenvectors of the covariance matrix using the 'eig' function in matlab. Using the output of the 'eig' function to generate a plot similar to that in problem 3. (Hint: you may need to use the 'sort' function.)

c. Then, find the PC scores. Show a plot the first few scores (versus time).
d. Finally, find the loadings for each neuron, and generate a plot similar to that in problem 7.