ECE 5960/6960 BME 5900/6900 – Fall 2022
Mathematical Tools for Neural Data Analysis and Modeling

Homework 2

(courtesy of Profs. Simoncelli and Landy, NYU)

Due: October 7, 2022

See the course web site for submission details. Reminder: rather than using the functions pinv() and norm(), use the linear algebra tools we learned in class. Please: don't wait until the day before the due date... start *now*!

1. **Trichromacy.** Load the file `colMatch.mat` in your MATLAB environment. This file contains matrices and vectors related to the color matching experiment presented in class. In particular, the variable `P` is an $N \times 3$ matrix containing wavelength spectra for three "primary" lights, that could be used in a color-matching experiment. For these problems $N = 31$, corresponding to samples of the visible wavelength spectrum from 400nm to 700nm in increments of 10nm.

   The function `humanColorMatcher.p` simulates a normal human observer in a color matching experiment. The input variable `light` should contain the wavelength spectrum of a test light (a 31-dimensional column vector). The variable `primaries` should contain the wavelength spectra of a set of primary lights (typically, a 31×3 matrix, as for matrix `P` described above). The function returns a 3-vector containing the observer's "knob settings" - the intensities of each of the primaries that, when mixed together, appear identical to the test light. The function can also be called with more than one test light (by passing a matrix whose columns contain 31-dimensional test lights), in which case it returns a matrix whose columns are the knob settings corresponding to each test light.

   (a) Create a test light with an arbitrary wavelength spectrum, by generating a random column vector with 31 positive components (use `rand`). Use `humanColorMatcher` to "run an experiment", asking the "human" to set the intensities of the three primaries in `P` to match the appearance of the test light. Compute the 31-dimensional wavelength spectrum of this combination of primaries, plot it together with the original light spectrum, and explain why the two spectra are so different, even though they appear the same to the human.

   (b) Now characterize the human observer as a linear system that maps 31-dimensional lights to 3-dimensional knob settings. Specifically, run a set of experiments to estimate the contents of a 3×31 color-matching matrix `M` that can predict the human responses. Verify on a few random test lights that this matrix exactly predicts the responses of the function `humanColorMatcher`.

   (c) The variable `Cones` contains (in the rows) approximate spectral sensitivities of the three color photoreceptors (cones) in the human eye: `Cones(1,:)` is for the L (long-wavelength, or *red*) cones, `Cones(2,:)` the M (green) cones, and `Cones(3,:)` the S (blue) cones. Applying the matrix `Cones` to any light $\vec{l}$ yields a 3-vector containing the average number of photons absorbed by that cone (try `plot(Cones')` to visualize

them!). Verify that the cones provide a physiological explanation for the matching experiment, in that the cone absorptions are equal for any pair of lights that are perceptually matched. First, do this informally, by checking that randomly generated lights and their corresponding 3-primary matching lights produce equal cone absorptions. Then, provide a few lines of matlab code that provide a more mathematical demonstration, along with an extended comment explaining your reasoning using concepts from linear algebra. [Hints for two possible approaches: (1) write math/code that computes cone responses for any test light and then computes the weighted combination of primaries that would produce the same cone responses - show that this is numerically the same as the color-matching matrix; (2) convince yourself, and explain why, it is sufficient to show that `M` and `Cones` have the same nullspace. Then use SVD to demonstrate that this is true!]

(d) The function `altHumanColorMatcher(light,primaries)` simulates a color-deficient human observer in a standard color matching experiment. (i) for a random test light, compare the knob settings for this observer with those of the normal human. Do this for several runs of `altHumanColorMatcher(light,primaries)`. How do they differ? (ii) Compute cone absorptions for the test light, and for the mixture of three matching primaries (by applying the `Cones` matrix). Do this for both the normal human observer, and for multiple runs of the abnormal observer. Try this for several different test lights. How do the cone responses of the normal and abnormal observers differ? Can you offer a diagnosis of the underlying cause of color deficiency in the abnormal observer?

2. **Polynomial regression.** Load the file `regress1.mat` into your MATLAB environment. Plot variable Y as a function of X. Find a least-squares fit of the data with polynomials of order 0 (a constant), 1 (a line, parameterized by intercept and and slope), 2, 3, 4, and 5. [Compute this using `svd` and basic linear algebra manipulations that you've learned in class!] On a separate graph, plot the squared error as a function of the order of the polynomial. Which fit do you think is "best"? Explain.

3. **Constrained Least Squares Optimization.** Load the file `constrainedLS.mat` into MATLAB. This contains an $N \times 2$ data matrix, `data`, whose columns correspond to horizontal and vertical coordinates of a set of 2D data points, $\vec{d}_n$. It also contains a 2-vector `w`. Consider a constrained optimization problem:

$$\min_{\vec{v}} \sum_n \left(\vec{v}^T \vec{d}_n\right)^2, \quad \text{s.t.} \quad \vec{v}^T \vec{w} = 1.$$

Thus, the *constraint* on $\vec{v}$ is that it must lie on a line, perpendicular to $\vec{w}$, whose perpendicular distance from the origin is $1/||\vec{w}||$.

(a) Rewrite the optimization problem in matrix form. Then rewrite the problem in terms of a new optimization variable, $\tilde{v}$ (a linear transformation of $\vec{v}$), such that the quantity to be minimized is now $||\tilde{v}||^2$. Note: you must also rewrite the constraint in terms of $\tilde{v}$.

(b) The transformed problem is one that you should be able to solve. In particular, you must find the shortest vector $\tilde{v}$ that lies on the constraint line. Compute the solution for $\tilde{v}$, and plot the solution vector, the constraint line and the transformed data points.

(c) Transform the solution back into the original space (i.e., solve for $\vec{v}$). Plot $\vec{v}$, the original constraint line, and the original data points. Is the optimal vector $\vec{v}$ perpendicular to the constraint line? On the same graph, plot the total least squares solution (i.e., the

vector that minimizes the same objective function, but that is constrained to be a unit vector). Are the two solutions the same?

4. **Dimensionality reduction with PCA.** Professors Hugh Bell and Wee Zell were recording extracellular action potentials (i.e. spikes) from cat primary visual cortex late one evening when their computer malfunctioned. It had already isolated a set of 400 time windows in which voltages had crossed a threshold, indicating the presence of spike. But these traces likely arose from multiple cells, with each cell producing a characteristic waveform, and the computer failed before sorting the voltage traces to determine how many cells were present, and which spikes arose from each cell. The professors come to you (the only math-tools-enabled graduate student still in the building at that hour), asking for help. They provide you with a file `windowedSpikes.mat` containing a 400×150 matrix, `data`, whose rows contain the electrode measurements (voltages recorded for each 150 msec window, at 1msec intervals). Your task is to determine how many neurons produced these 400 spikes.

   (a) Plot the 400 waveforms superimposed and describe what you see. Be sure to label your axes! Using these spike waveform plots, can you devise a way to deduce how many neurons produced these spikes? Feel free to include an additional plot containing just a subset of the waveforms in order to aid in your explanation.

   (b) Perform principal components analysis (PCA) on your data, and plot the eigenvalues in descending order (alternatively, compute the SVD of `data`). It might help to display the eigenvalues on a log-scale. Interpret what you see.

   (c) Project each of the 400 spike waveforms onto the top two principal components of the dataset, and plot the resulting values as points in 2 dimensions. Describe what you see. Can you deduce how many distinct neurons produced the 400 voltage traces?

   (d) Now project each waveform onto the top three principal axes, and plot in 3 dimensions (you may want to spin it around, using `rotate3d` in matlab). Are there any significant changes you see? Using the 3D plot, can you inform Drs. Bell and Zell how many neurons they likely recorded from?

   (e) Extra credit: how would you determine which neuron fired each of the 400 observed spikes? Describe your strategy, including appropriate linear algebraic expressions for the required calculations.