Problem Set 2: MATLAB exercises. (Worth 7.5 points: 20/2.67).

Administrative comments:

a. Due date: Feb 3 (Mon), midnight. Remember: 5% off for each late hour!
b. This PS is meant to get you into the very trenches of data visualization in MATLAB. Rather than trying to set aside a large chunk of time on a single day to work on this problem set, splitting up work over a couple of days can help. Also, consider grappling with problem #2 last.
c. Upload your answer sheet and code to Gradescope. Note that there will be two assignment pages for each problem set, one for the PDF answer sheet and one for MATLAB code (single .m file, appropriately commented)
    a. Name your MATLAB code file as: ps#_name.m (e.g., ps1_shreesh.m)
    b. Within the code.m file, separate the code for each problem with a "section break", which, in MATLAB, is obtained by inserting a line with %% followed by the problem number. (e.g., %% Problem 2)
    c. Name all your figures as: ps#_name_figX.pdf (ps1_shreesh_fig3.pdf, for problem 3).
d. In general, efficiency of the code will be valued, as will be validity in the choice of variable names (per Tutorial under MATLAB-REVIEW module) and the clarity of the comments. Note: using the % symbol before a line of text comments out that line.
e. Collaboration/discussion are permitted (encouraged, in fact). However, make sure that the solutions you turn in are your own and that you understand everything that you put in your solutions. The goal, here, is to have a solid foundation upon which we can build in the coming weeks.
f. Always abide by JHU honor code (see course syllabus).

General comments: For all the problems below,

a. Label the x,y (& z) axes appropriately. [Hint: "xlabel" (and ylabel and zlabel)]
b. Make sure that the axis labels are of fontsize 15 [Hint: h=xlabel("blood pressure"); set(h,"fontsize",…)]
c. Set the font size of the x and y tick labels to be 10. ["help set"; the syntax that you will ultimately use will look like: set(gca,"fontsize",…)].
d. Provide a title for all plots [hint: "help title"]. Set title's fontsize to be 18.
e. Use the command "print –dpdf figName" to save the current figure as a pdf file. The "current" figure is the one that you most recently clicked on with your mouse.

---

1. 3D plot (5 points). Patients numbered 1, 2 & 3 are administered two drugs 'a' and 'b'. Their responses to the drugs are measured over seven time points in terms of three variables: (i) blood pressure, (ii) reaction time on a cognitive task, and (iii) performance on the cognitive task (all normalized in some way).

Load the associated mat file (**ps2_2_3D.mat**) and you will find six matrices named accordingly. For instance, a1 is the response of patient 1 to drug 'a' and so on. Each row in a1 contains the response at given time point and the columns correspond to the three variables.

A. (4 points) In one figure, plot (in 3D) the responses of all the patients to the two drugs as a function of time and see how the two drugs have different temporal effects on the patients. Make sure to use two different colors for the responses of the patients to the two drugs. Mark all the data points for drug 'a' using circles and for drug 'b' using squares.

This is completed in the problem set script.

B. (1 point) Finally, place a piece of text somewhere on the plot (at an unobtrusive location) that says "Red -> Effect of drug 1, Blue-> Effect of drug 2."

This is completed in the problem set script.

Note: Your plot should resemble the adjoining figure: the three **axes** in your plot need to be the three **variables** (blood pressure, reaction time, and cognitive performance). The two colors you use could be different from those in the plot.

Hint: If you're having trouble starting, try thinking of plotting one curve at a time (one patient, one drug, one variable, all 7 time points). How would you plot this in 3D? Then see if you can generalize. Finally, see if you can make this efficient. (Although, efficiency is not critical for this question.)

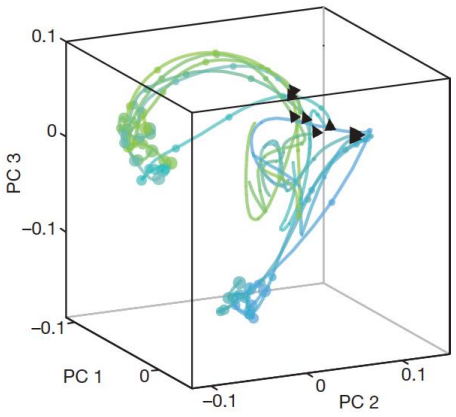
BONUS (2 points): Data corresponding to different time points are to be represented using symbols of increasing size, with early time points represented using small symbols or markers, and later data points represented using large symbols.

Hints:

a. Say "clear all" on the command line before loading the data for this problem. What does this do?

b. Use *plot3* for plotting. For the bonus question, consider *scatter3*.
c. Use "help plot" to figure out how to use different symbols for plotting
d. Rotate the plot to view at a different angle if the temporal difference is not already evident. ("help view")
e. Use "xlabel" (and ylabel and zlabel) commands to name the axes
f. Use "text" to place the text.

Dataset: **ps2_2_3D.mat**



2. Errorband (5 points). Recall the idea of "errorband" discussed in class (visualizing a series of error bars as a shaded area that is translucent). In the associated dataset, you are given the average bold signal from a voxel as a function of time. You are also given the s.e.m of the signal. Plot the average as a function of time, with the sem shown as an errorband.

Hints: Use the fill command. Also use the alpha(value) command, where value is between 0 and 1.

Dataset: **ps2_3_errb.mat**

This is completed in the problem set script.

3. Surface plot (5 points). Create a variable x that varies from 1 to 25 (in steps of 1). Create y so that y varies from 0.02 to 0.09 in steps of 0.01. Create a surface plot of z such that z = exp(-yx).

Hints:

a. "help meshgrid"
b. "help surf"
c. You will need to use the element-by-element-multiplication operation (i.e., A = B.*C).

This is completed in the problem set script.

4. Function (4 points). Write a **function** GetMeanSE.m which

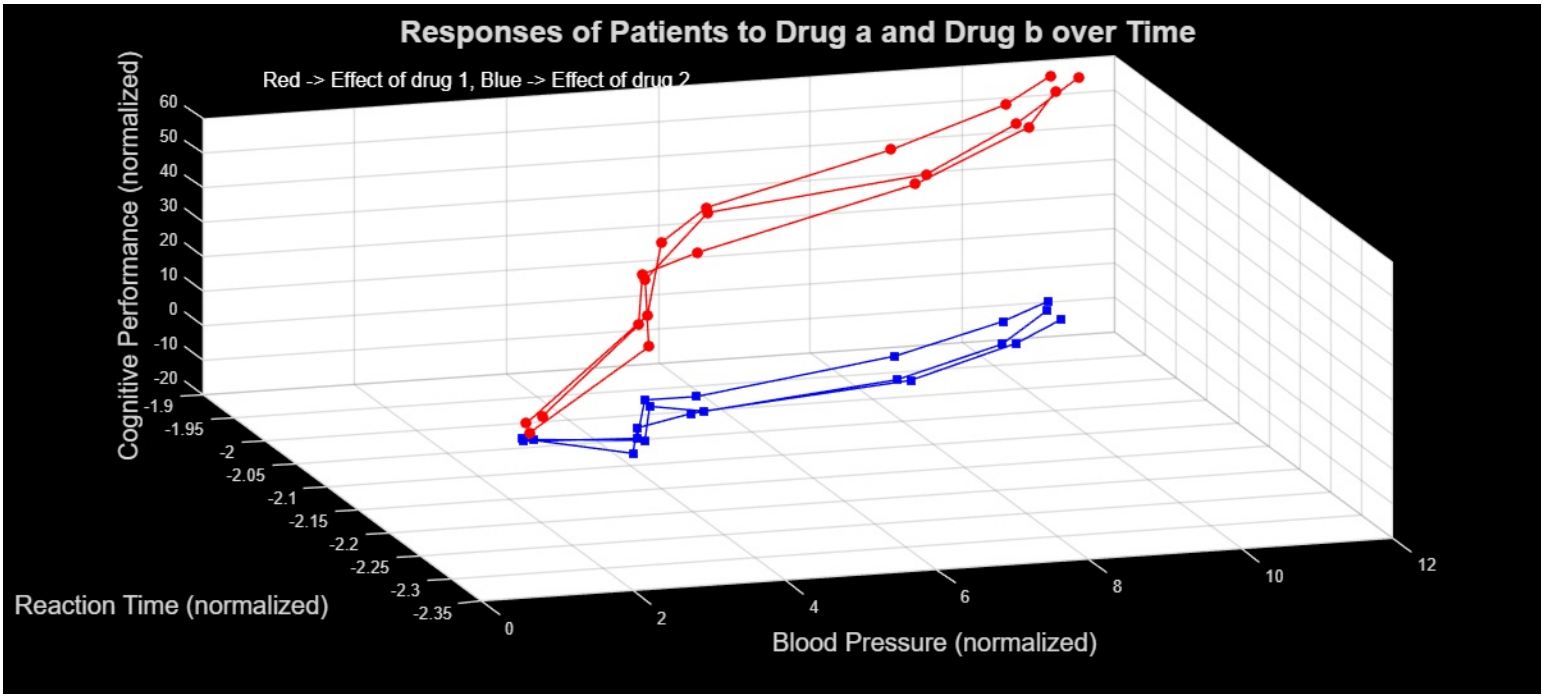
a. takes as input any vector x
b. (1) computes the mean, std, and sem of that vector
c. (1) determines whether or not x is normally distributed,
d. (2) and returns 4 outputs: the mean, std, sem, and the text 'Normal' if normal, and 'Not Normal', if not normal.

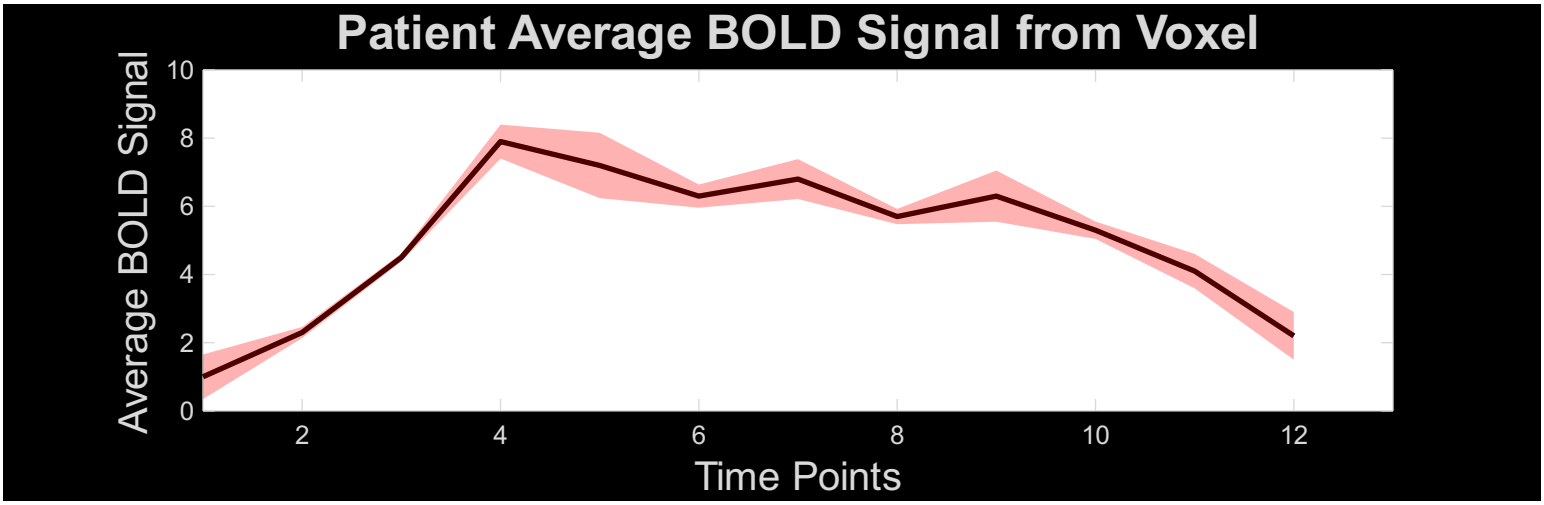
Hint: Check out "help function", and the one slide from class (for syntax). Also, "help if".

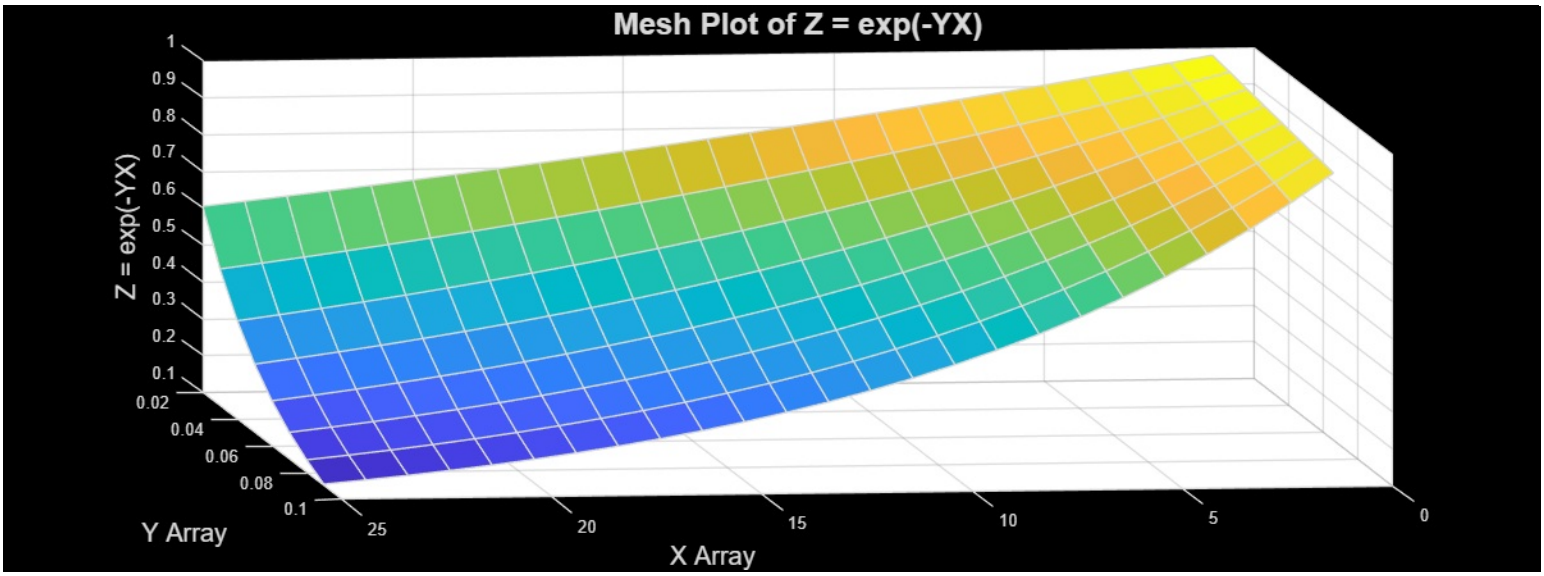
This is completed in the problem set script.

5 (1 point). x is a 3D matrix. Say, x=rand(3,5,2); Try typing sum(x). What does it do? Can you describe the principle MATLAB is using to produce the answer?

The help function tells us that the sum function returns the sum of elements of a matrix A along the first array dimension whose size does not equal one. Our first dimension of rows is 3! This means that MATLAB will sum down the rows! So we end up with a result of a 1x5x2 matrix! Its a sum of the rows for each column and there are two slices!


Responses of Patients to Drug a and Drug b over Time
Red -> Effect of drug 1, Blue -> Effect of drug 2
Cognitive Performance (normalized)
60
50
40
30
20
10
0
-10
-20
Reaction Time (normalized)
-1.9
-1.95
-2
-2.05
-2.1
-2.15
-2.2
-2.25
-2.3
-2.35
Blood Pressure (normalized)
0
2
4
6
8
10
12


Patient Average BOLD Signal from Voxel
Average BOLD Signal
0
2
4
6
8
10
2
4
6
8
10
12
Time Points


Mesh Plot of Z = exp(-YX)
Z = exp(-YX)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.02
0.04
0.06
0.08
0.1
Y Array
25
20
15
10
5
0
X Array