Lab Assignment #6

Due SATURDAY 3/5/2016 at 4pm on bCourses

This E7 assignment focuses on graphical representations. You are encouraged to use the Matlab documentation to get details on some of the plotting functions you will need. This assignment will be checked for proper **functions and figures and *comments* in your code**. For this assignment you will write one function and one script which you will "publish" to create a pdf. More detailed instructions for submitting your assignment are included at the end of this document.

1 X-Y Plots: Bay Area Sports Teams

In the Bay Area, there are two professional football teams (San Francisco 49ers and Oakland Raiders), two professional baseball teams (San Francisco Giants and Oakland Athletics), one professional hockey team (San Jose Sharks), and one professional basketball team (Golden State Warriors). In this problem, you will be asked to write a function, `BayAreaTeams.m`, which creates a plot showing the winning percentages of some combination of the six aforementioned sports teams throughout the years. The winning percentage of a sports team for any given year is calculated as:

$$\text{Winning Percentage} = (\frac{\text{Games Won}}{\text{Games Played}}) \times 100 \qquad (1)$$

The winning percentage is calculated based on games for a single year. Thus, a new winning percentage is calculated for each year. The function header should be:

```
function [ ] = BayAreaTeams(varargin)
```

This function does not return any output variables to the workspace when called. The sole intention of this function is to produce a plot - not to output any variables. Note that there is one input argument in the function declaration line: `varargin`. `varargin` stands for "variable arguments in" and is used when the number of input arguments to a function is not always the same. In other words, `varargin` is used when there are several acceptable inputs (arguments) to a function, but the function does not require every input to be present for any given call to the function itself. (For example, the Matlab function `zeros` which generates an array filled with zeros, can either be used with one argument as `zeros(N)`, or with two arguments as `zeros(M,N)`, or with even more arguments - see `doc zeros`.)

In this case, the function can take up to 6 arguments. Acceptable inputs for the function BayAreaTeams are the following character arrays:
`'Raiders', 'Niners', 'Warriors', 'Giants', 'Athletics','Sharks'.`
You should design your function so that these inputs are case insensitive (see the function `strcmpi` to do this). For instance, let's say you want to plot the winning percentages of only the Oakland Raiders and the San Francisco 49ers on the same figure. To do this, you would call the function as follows: `BayAreaTeams('Raiders', 'Niners')` Note that we did not use the other four acceptable arguments. This is the functionality that `varargin` provides. Type `doc varargin` or `help varargin` into your command window to learn more about how varargin works.

The data file `BayAreaSports.mat` has been provided. Inside this .mat file is a `struct` array called `BayAreaSports`, which contains several fields. Load the `.mat` file into your workspace and explore the `struct` to become familiar with the field names and the data that exists there. You'll notice that for each team, data exists for the years played as well as the number of wins per year. However, while some teams have data for the number of games played (which is necessary for calculating the winning percentage), some teams do not provide the number of games played and instead provide the number of losses per year. Make sure to think critically about how you can obtain the number of games played per year for the teams that do not directly provide this data.

A good regular season record is important, but what ultimately matters is if you win a championship! Thus, the file `BayAreaSports.mat` also includes the years each team won championships in their respective sport.

Your function should perform the following tasks:

1. Load the file `BayAreaSports.mat`

2. Plot the winning percentage as a function of years for as many of the six teams as requested by the function call on a single figure. Hint: Recall structs from Lab 3.

3. For each team requested, circular markers should be used with lines connecting the circular markers. The line thickness should be set to `2`.

4. Grid lines should be shown on the figure (see `help grid` to learn about this command).

5. If no input is provided, the function should not return a figure and instead display the output: `Please enter at least one professional Bay Area sports team.`

6. If an input is provided that is not one of the acceptable six inputs mentioned above, the function should not return a figure and instead display the text:
   `Please enter either 'Raiders', 'Niners', 'Warriors', 'Giants', 'Athletics', or 'Sharks'.`

7. The x-axis of the figure should be labeled `Years` with a font size of `20`.

8. The y-axis of the figure should be labeled `Winning Percentage` with a font size of `20` and should be limited to values from 0 to 100.

9. The figure should be given a title `Comparison of Bay Area Professional Sports Teams` with a font size of `20`.

10. A legend should be provided that contains an entry for each team requested. The legend should be in the upper right-hand corner of the figure and not overlap any of the data.

11. Use team-specific colors for your plots. MATLAB allows you to plot lines with very specific line/marker colors by using RGB values. To make your figure more festive, refer to the table below when assigning line/marker colors for the six teams.

12. Plot the championship-year winning percentages on top of the data already plotted on your figure for each team requested. This championship data should be plotted as gold pentagrams (5-point stars) with a marker size of 20. The RGB array for gold is `[1 .843 0]`. These "gold stars" should not be connected with lines - plot only the markers on top of the pre-existing data.

Team	Line Color	Marker Color
Raiders	[0, 0, 0]	[0.737, 0.769, 0.788]
Niners	[0.784, 0, 0.126]	[0.753, 0.6, 0.302]
Giants	[0, 0, 0]	[1, 0.302, 0]
Athletics	[0.016, 0.290, 0.235]	[0.988, 0.729, 0.204]
Warriors	[0.004, 0.275, 0.678]	[0.973, 0.773, 0.212]
Sharks	[0.075, 0, 0.028]	[0.04, 0.467, 0.541]

Test case: if you want to compare the Oakland Raiders and the San Francisco 49ers (using the function declaration above), your figure should look like Figure 1 below. After completing the function, include the following code in your script file:

```
type BayAreaTeams % display function text
BayAreaTeams('Raiders', 'Niners') % show Raiders/49ers comparison
BayAreaTeams('Athletics', 'Giants') % show A's/Giants comparison
BayAreaTeams('Warriors', 'Sharks') % show Warriors/Sharks comparison
```

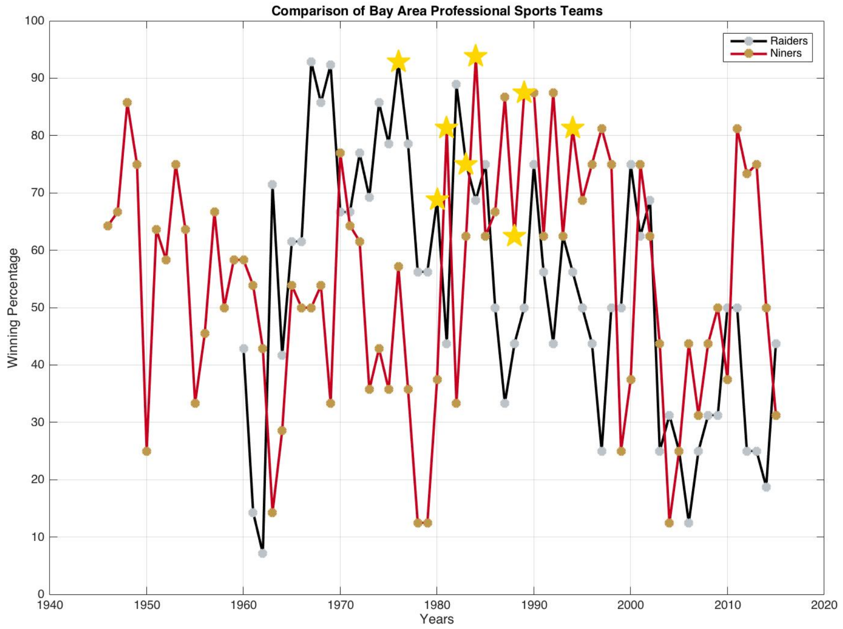

Figure 1: Plot comparing the win percentages over the years of the Raiders and Niners

2 3D Surface Plots and Subplots

The purpose of this problem is to introduce you to three dimensional plots and subplots. Imagine we have a function as follows:

$$z(x,y) = \frac{-y\cos(x)}{\exp(\cos(x)^2 + \sin(y)^2)} \tag{2}$$

First, you will plot the mathematical function above in three dimensions across the following domain for the independent variables x and y:

$$x, y \in [-5, 5] \tag{3}$$

You will need to create a discrete set of values for x and y at which MATLAB will evaluate the function $z(x,y)$. For this problem, you should assume a step size of 0.1 between these discrete values. Look into the built-in MATLAB functions `meshgrid` to create the underlying x, y grid. Compute z, then use `surf` to make your three dimensional figure. (Type `doc meshgrid` and `doc surf` into your command window).
You do not need to write a function for this problem. Instead, type all of your code for this question in the same script file as the code for the other questions (except for the function–see the end of the document for details). For your figure, define the axes and title as follows:

1. Set the x-axis to the text `X Axis` with a font size of 16
2. Set the y-axis to the text `Y Axis` with a font size of 16
3. Set the z-axis to the text `Z Axis` with a font size of 16
4. Set the title to the text `E7 - Lab 6 - Question 2` with a font size of 20
5. Add a colorbar to your figure.

When you are finished, your figure should look like Figure 2. You can experiment with the `shading` command and change your `colormap` if you like.

Next, you will look at two dimensional "slices" of the function. The slices we are interested in analyzing are the following:

1. The slice on the $x = 0$ plane
2. The slice on the $y = 0.5$ plane

Keep in mind that these slices are two dimensional, not three dimensional. For example, a slice at $x = 0$ means that the equation above is evaluated at $z(0, y)$. Similar observations can be made about the slice $y = 0.5$. Note that you do not need to do additional calculations. Since you are just going to be plotting a slice across the existing data that you have already computed in creating `z`, you can create this "slice" by properly indexing the array `z`.

You should plot both two dimensional slices on the same figure. To do this, make use of subplots – a feature in MATLAB that allows you to break a figure down into smaller,

individual axes that can contain different plots. Type `doc subplot` into your command window to learn more about how subplots work. You will be making a 1×2 subplot.
The left-hand-side subplot should contain the two dimensional plot of the x = 0 slice and contain the following axis/title labels:

1. Set the x-axis to the text `Y Axis` with a font size of 16

2. Set the y-axis to the text `Z Axis` with a font size of 16

3. Set the title to the text `Slice at X = 0` with a font size of 20

The right-hand-side subplot should contain the two dimensional plot of the $y = 0.5$ slice and contain the following axis/title labels:

1. Set the x-axis to the text `X Axis` with a font size of 16

2. Set the y-axis to the text `Z Axis` with a font size of 16

3. Set the title to the text `Slice at Y = 0.5` with a font size of 20

Turn on the background grid for both subplots so your figure looks like Figure 3. Note that both of plots shown are on a single Matlab figure window. This is the functionality that `subplot` provides. Visually compare the subplots to the 3D plot you plotted earlier and try to find where these slices exist within the 3D figure.

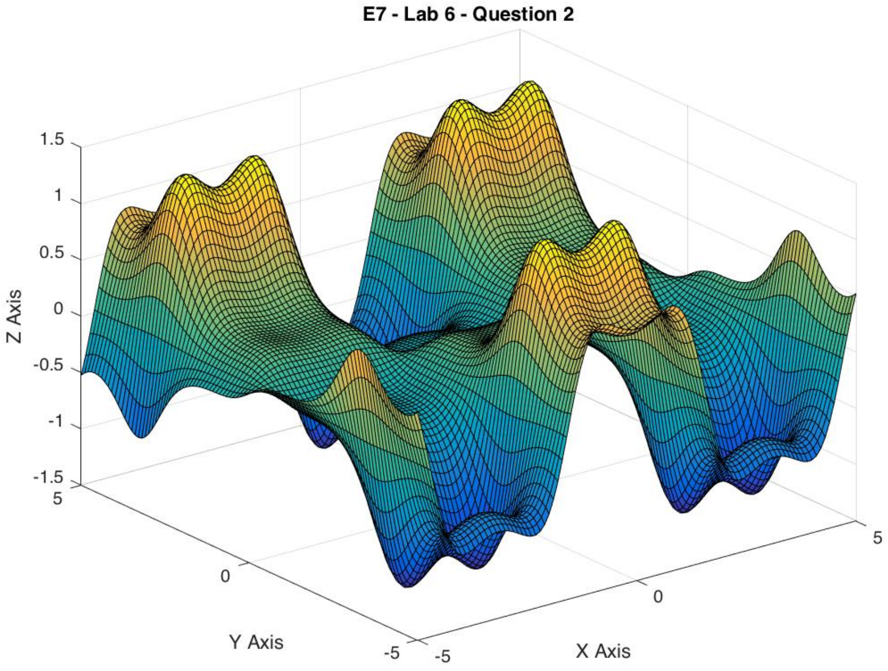

Figure 2: 3-D Surface plot

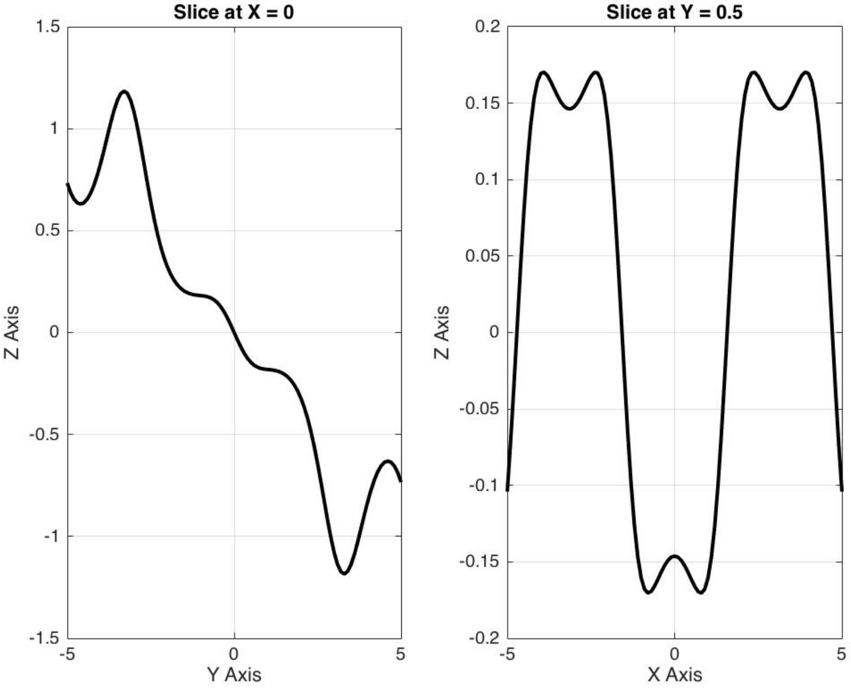

Figure 3: 2-D line plots of (1) Z axis versus the Y Axis, Slice at X = 0; (2) Z Axis versus the X Axis, Slice at Y = 0.5

3 Contour & Vector Plotting: Aquifer Simulation

Groundwater is an important resource which provides water for municipal, agricultural, and industrial needs. Contamination from a variety of sources can diminish the quality of groundwater supplies, causing health problems for people who drink it or other problems, depending on the use. Modeling the motion of contaminants can allow us to use groundwater resources while preventing exposure to toxic contaminants. Modeling contaminant migration in groundwater requires computing groundwater velocity, which can be controlled by various soil properties, such as permeability. Permeability is a measure of how conducive a soil is to flow, and can vary by orders of magnitude over relatively short distances. (You can imagine that the permeability of sand will be much greater than that of clay, allowing water to flow through sand with significantly greater velocity.) In this question you will visually demonstrate the relationship between soil permeability and groundwater velocity.

The file `K.csv` contains values of permeability for an aquifer, which was simulated using geostatistical methods. The files `vx.csv` and `vy.csv` contain x components and y components of velocity, respectively, throughout the same aquifer, obtained using numerical methods. All of the files contain data on a 50×50 grid, with each grid cell representing a $10m \times 10m$ area. These datasets were generated by your head GSI, Brad Harken, who works on problems of this nature for his PhD research!

In this question you will add to your script `.m` file from problem 2 where you:

1. Load the three data files named above using `csvread` (be careful with the headers).

Hint: Adjust `csvread` to start at appropriate point in data file.

2. Show the spatial distribution of permeability using a filled contour plot (use the command `contourf`).

3. Display the spatial variability of groundwater velocity by plotting velocity vectors on the same plot as the filled contour plot. Use `quiver` to help you with the velocity plot.

4. Often when analyzing data, transformations of the data can be a helpful tool for seeing the data in a new perspective and visualizing different trends. For instance, taking the logarithm of a dataset can take emphasis away from extreme values and more simply show overall variability. On a new figure, show a filled contour plot of the natural logarithm of permeability.

5. Ensure that your plots are presentable (adjust the colorbar, shape and colors of the arrows, legend, title, axis as needed). Create a title and axis labels that make the content of your plots easy to understand on their own (without needed accompanying text). Note the axis values and ensure they match the appropriate field measurements. Hint: You need to convert your axis to adequately match the field scale measurements.

Submission Instructions

For this assignment, your answers will be contained in a script file called `Lab6.m`. This script file should be separated into three code subsections, one for each problem (use the double percentage sign %%). Your script should contain the following:

1. A section which displays your `BayAreaTeams` function (use the command `type BayAreaTeams`) and calls this function three times in order to show:

   A figure comparing the Oakland Raiders to the San Francisco 49ers

   A figure comparing the Oakland Athletics to the San Francisco Giants

   A figure comparing the Golden State Warriors to the San Jose Sharks

2. Two sections with code for Problems 2 and 3 respectively (in the same .m file).

3. Comments (where appropriate) in all the Matlab code you write.

After you are done with your `Lab6.m` script, publish it and make a pdf that includes the figures. This can be done by going to the "Publish" tab in Matlab and clicking the "Publish" button. A window will pop up with the file in the form of a `.html` document. You will then need to convert this `.html` file into a `.pdf` (you can either "print to" a pdf or "save as" a pdf depending on your html viewer).

Finally, submit a SINGLE .zip file named `LAB6.zip` containing:

1. Lab6.m

2. Lab6.pdf

3. BayAreaTeams.m