Homework 2

Wednesday, April 19, 2017 1:56 AM

Logistics

- This is a long homework wo we'll break it into parts.
- Submit your solutions on Canvas, this applies to each and all sections:
  - Code in one or more Matlab (.m) files
  - Report and figures as a single MS Word or pdf file
  - Put everything into a single ZIP file
- You must use Octave or Matlab code/instructions to obtain the results
  - Submit any scripts you used
- Besides just finding the results, you must discuss your findings, explain the why's and hows, and give reasonable interpretations.
- List any assumptions when needed to disambiguate

READ ch 2.3-2.6

Bayes Practice [100p]

In this exercise we will use Matlab/Octave to write a linear discriminant based on the Bayes theory we have covered in lecture.

A look at the data

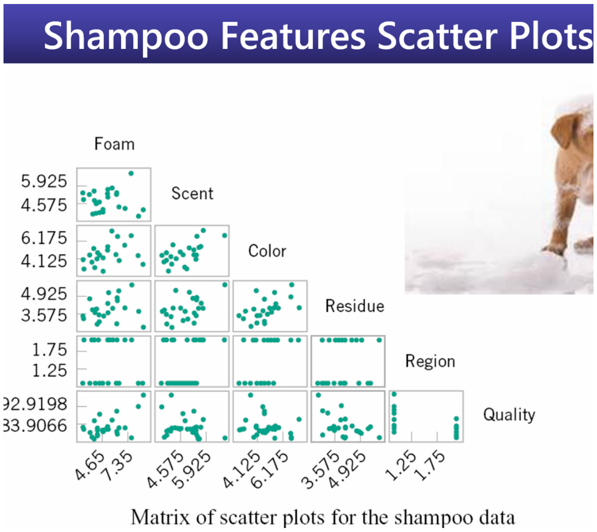
The iris file below contains a popular dataset used in pattern recognition. This dataset contains information about three varieties of flowers. The features (variables) used to identify flowers are petal and sepal width, and petal and sepal length.	

We will be using this da taset for various exercises in class and home work.

Linear Discriminan t

The idea is that based on the measure of a sepal and a petal from an iris flower, you try to classify what species it belongs to. The classes are going to by the three different types of flower.

Part 0 [15 pt]

Dataset	Instructions
iris	• Load the iris dataset in Matlab/Octave • This is composed of three matrices, one for each type of flower: • Versicolor • Virginica • Setosa • Each matrix has 50 observations with the following features • Petal Width, Petal Length, Sepal Width, Sepal Length • Create a scatter plots of the data, coloring each flower type with a different color • Include these plots in your report • Use a format like the one discussed in the Statistics review for the Shampoo example (refer to the slides for background and detail) •  Shampoo Features Scatter Plots Matrix of scatter plots for the shampoo data • Include a legend or clearly explain what each color represents, e.g., •

Rubric

Criteria	Points	Notes
Code	5pt	Code must be well structure and commented In this part of the assignment, if the code is present, you'll get points, but in the future, lousy code will be penalized
Plots	5pt	-1p for missing units or descriptions of axis, e.g., "Petal Length" -1p for no color-coding (legend not required at this point) -0.5p for missing title -0.5p if plots are hard to read -2p if required summary plot format is missing (see Shampoo example above) MUST SHOW 6 PLOTS
Analysis	Outstanding: 5pt Sufficient: 5pt (in the future, it won't be the same, eg., 4pt) Poor: 3pt Missing: 0pt	This part of the assignment didn't need much detailed analysis, but in the future, a reasonable level of depth and discussion will be required. Remember, your interpretation of the results is what matters the most! In the future, points will also be deducted for lousy redaction, spelling/grammar mistakes, typos -- professionalism is expected.

Note: Matlab already provides some built-in functions used for classification, and for example, using a linear discriminant analysis:

```
fitcdiscr
resubPredict
```

This is ok, but where is the fun of learning if we just use premade functions without understanding what's going on. And of course, keep practicing and familiarizing yourself with Matlab/Octave!

Part I [15 p]

Basic statistics

For every feature:

- Compute means and variances

Decide if any of the gaussian discriminant cases applies

Density Plots

We'll start with petal information only. Then we will extend to the other variables

- Using the ***petal width*** data, plot normal distributions for each of the classes
  - Plot all the normal distributions on the same figure, and title it "Petal Width Normal Distributions"
  - Code color each of the distributions differently and add a Legend to the plot
- Do the same using the ***petal length*** with a suitable title
- You can use **normfit** function for starters

Snippet of code to use instead of normfit (if you are using Octave)

```
mu = mean(X);
st = std(X);
x = linspace(mu-3*st,mu+3*st,100);
plot(x, normpdf(x,mu,st));
```

Generates a plot similar to this:

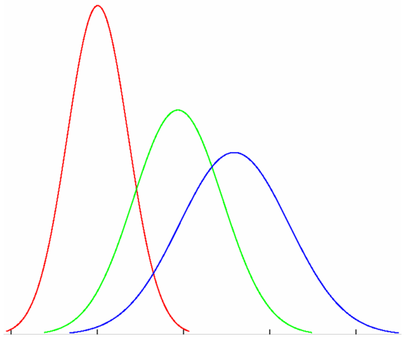


Figure 1

Rubric

Criteria	Points	Notes
Code	2pt	Code must be organized and commented: should include code for the **plots**

		and code for the **statistics** -1 no coments
Statistics	2pt	-0.25 for a missing calculation (4 features: mean and variance for each)
Plots	6pt	-1p for missing units or descriptions of axis, e.g., "Petal Length" -1p for no color-coding -1p for no legend -0.5p for missing title -0.5p if plots are hard to read -2p if format is incorrect (See figure 1 above)
Analysis	Outstanding: 5pt Sufficient: 4pt Poor: 3pt Missing: 0pt	Statistics and plots should be discussed in this section. Remember, your interpretation of the results is what matters the most! In the future, points will also be deducted for lousy redaction, spelling/grammar mistakes, typos -- professionalism is expected.

Part II [10p]

Only this part of the assingment will be In-Class work. You may work in teams of 2-3 students, one submission per team.

Write your own Matlab/Octave function for linear discriminant g_i($\mathbf{x}$) based on Bayes Theory covered in class using Gaussian model.

You only implement g once, and you call it three times for each x, so the "prototype" for the function implementation would have to take the class (i) and x: `p = g(x, i)`,

(https://en.wikipedia.org/wiki/Mahalanobis_distance)

Matlab function Example

This is a function that generates a plot (unrelated to g), just for example purposes.
Create a new .m file, the first line of the function has the following format

```
function foo = generate_3D2C_plot(x, y, z, class1, class2)
```

Where foo is the return value

Rubric

Criteria	**Points**	**Notes**
Code	5pt	Code must be organized and commented: should include code for the function g; -1 no coments
Analysis	Outstanding: 5pt Sufficient: 5pt (in the future, it won't be the same, eg., 4pt) Poor: 3pt Missing: 0pt	Discuss your approach to the implementation of the discriminant function, explain what formulas you used and why. This is essentially the documentation for your code Remember, your interpretation of the results is what matters the most! In the future, points will also be deducted for lousy redaction, spelling/grammar mistakes, typos -- professionalism is expected.

Part III [70p]

This part of the assignment is homework, henceforth, you have to work individually.

- Test your discriminant function, i.e., g_i(**x**)
  - Out of the 150 observations, randomly select a sample of 60 data points (**x**) from each flower class.
  - Use your discriminant functions on each of those **x** and see what the classification results are
  - Repeat this process with **three different random samples**

- Find the average classification performance of your classifier (what percentage of values are correctly classified):
  - You need to test each feature vector $\mathbf{x}_j$ (where j=1...60) from the random sample with the discriminant function for each class, e.g.,
    r_setosa=g($\mathbf{x}_j$, setosa)
    r_virginica=g($\mathbf{x}_j$, virginica)
    r_versicolor=g($\mathbf{x}_j$, versicolor)
    and select the optimal return value among those three (r_setosa, r_virginica, r_versicolor). ***The optimal value is that closest to 0.***
  - How to compute the performance:
    - The performance of a classifier is measured as the percentage of correctly classified observations, e.g., if your classifier only correctly identifies 30 observations out of the 60, then the performance is 50%
    - Initialize a counter to count the number of "correct identifications"
    - For every vector $\mathbf{x}_j$, check if the optimal value corresponds to the class that vector actually belongs to. For instance, if $\mathbf{x}_j$ belongs to *setosa* and the optimal value from testing the three g's is coming from r_setosa=g($\mathbf{x}_j$, setosa), then, increase the counter by 1.
    - After checking those 60 vectors, compute the performance and save that value.
    - Repeat this process with three different sets of 60 random feature vectors, then average the resulting performance measurements.
    - Include these values in your report and discuss the results: focus on explaining the WHYs not just state the facts. For instance, you can discuss why your discriminant function classifier works well or not. Can you explain this in terms of the previous parts of the homework? Think about the plots and statistics in Parts 0 and Part I.

Example approach for selecting random rows from an array

You may use any suitable approach to select the 60 random feature vectors. The strategy suggested below is just to provide guidance for those unfamiliar with Matlab:

- Load your data into a single matrix, i.e., 2D array (150 rows, 5 columns), we'll call that matrix **c1**

```
Type    PW      PL      SW      SL
0       2       14      33      50
```

1	24	56	31	67
1	23	51	31	69
0	2	10	36	46
1	20	52	30	65

The feature vector for the first row is [0 2 14 33 50]. Note that the first column is just a class tag indicating the type of flower (e.g., 0 for setosa, 1 for virginica, 2 for versicolor), but has no statistical significance

- You can get the random sample by randomizing the ordering of your data, for example:

```
%get the dimensions of your array
%(in our case, we know what these are, but let's do it anyway)
[s1, f1] = size(c1);

% add another column with a random number
for i=1:s1
    c1(i,f1+1)=rand;
end

% Randomize by sorting the matrix based on the new random column
cls1=sortrows(c1,f1+1);

% Copy only the columns and rows of interest into a new matrix
% first 60 rows, and columns 1 through 5
% (remove random number)
% I suggested that you keep the class tag so that you can compare
% at the end whether your classifier selected the right class
ctotal=[cls1(1:60,1:f1)];
```

The desired random sample is now in ctotal. You can now pass each row (feature vector or observation) to your discriminant function!

Deliverables:

- Code: .m file(s)
- Report: Problem descrip tion, implemen tation discussion and Analy sis

Rubric

Criteria	Points	Notes
Code	20p -3 for no comments or badly structured	Code should be well commented and structured
Testing	20p Outstanding: 20pt Sufficient: 18pt Poor: 10pt Missing: 0pt	Discuss your testing process, explain how you did it and why. Show your results. (This should be a section of your report)
Analysis	Outstanding: 30pt Sufficient: 26pt Poor: 15pt Missing: 0pt	Include and discuss statistics and plots from previous parts of the homework and use them towards your discussions. Discuss your performance results, putting them in context of the results (plots and statistics) obtained in previous parts of the homework, but also in terms of the theory and formulas for Bayes Linear Discriminant functions. Focus on the WHYs not just the What's.

		Remember, your interpretation of the results is what matters the most! Points will be deducted for lousy redaction, spelling/grammar mistakes, typos -- professionalism is expected.