CS171 - Introduction to Machine Learning and Data Mining
Spring 2018 - Assignment 1

Instructor: Vagelis Papalexakis, University of California Riverside

In this assignment you will work on understanding the attributes of a given dataset, visualizing the dataset in various ways, and doing a preliminary analysis on the data.

Question 0: Getting real data *[5%]*

In this assignment you are going to use data from the UCI Machine Learning repository (https://archive.ics.uci.edu/ml/index.php). In particular, you are going to use the following two datasets:

1. https://archive.ics.uci.edu/ml/datasets/Iris
2. https://archive.ics.uci.edu/ml/datasets/Wine

Download the datasets and write a script that loads them to your workspace. That script can be the preamble for the rest of the questions.

Question 1: Feature distribution *[35%]*

Here you are going to visualize the (empirical) distribution of each attribute for a given class in both datasets. More specifically, for each attribute of each class in each dataset

1) *[60%]* Calculate and plot equi-width histograms for the values of the attribute across the given class. You should parametrize your implementation with the number of histogram bins and create plots for b = 5, 10, 50, and 100 bins. In your report you should show all the plots organized by 1) dataset 2) attribute, 3) class, 4) bin size.
2) *[20%]* For the same data (organized in the same way as above), plot their Box-plots. You may use a library function for this.
3) *[20%]* For each distribution (using the histograms), under the figure you produce in your report also write whether it is 1) {mostly symmetric or mostly skewed}, and 2) {mostly unimodal, mostly bimodal/multimodal, mostly uniform}.

Question 2: Relations between features and data points *[60%]*

In this question you will work in understanding basic relations between different features and different data points. Those insights will be very useful when you are going to implement classification and clustering techniques later on in the class. The following plots and calculations should be carried out for each dataset.

1) *[20%]* Correlation Plots:
    a) Compute the Pearson correlation coefficient between pairs of features by implementing a "correlation(x,y)" function according to the definition.
    b) Plot the Feature-by-Feature correlation matrix
    c) What is the absolute minimum number of calls to the correlation(x,y) function in order to fill in this matrix?

d) Do you observe any correlated features? How can this information be useful? Compute the pearson correlation across features and plot the correlogram (i.e., the heatmap of the Feature-by-Feature correlation matrix).

2) *[20%]* Scatterplots *[only for the "Iris" dataet]*:
   a) Plot scatterplots for pairs of attributes, and color the points according to their label.
   b) Do you observe any pairs of features being discriminative? By "discriminative" we mean pairs of features that show good separation of the two classes in the 2D space defined by those features.
   c) Do you observe any pairs of features being non-discriminative? To what extent does this agree with the set of correlated features from #1?
3) *[60%]* Distances:
   a) Implement a distance function "distance(x,y,p)" that computes the Lp norm.
   b) For each dataset compute the distance using p=1 and 2 between all data points and fill in an Data Point x Data Point matrix.
   c) What is the absolute minimum number of calls to the distance function you need to do to fill in this matrix?
   d) Plot the matrix as a heatmap where the intensity is proportional to the distance
   e) For each data point, find its non-trivial nearest data point (e.g., the point that is not the same point). What is the label of the nearest point? Is it the same? Does the answer change for different values of p?

PLEASE READ BELOW - IMPORTANT INFORMATION

Programming language: For this assignment you are going to use *Matlab* or *Python*. Unless noted otherwise, you may ***not*** use library functions that already implement the questions that you have to implement in this assignment. As an indication, "mean", "max", "min" etc are standard functions and you may use them, but "norm" or "dist" are not (especially if you are asked to implement some distance function). Regarding non-standard Python libraries, you are allowed to use functionality of NumPy (as long it is not exactly what you need to implement.)

Deliverables:

1. A report in PDF format describing the approach taken in each question and containing the answers and figures required in each question.
2. **In your PDF you should include**: a) Name, b) UCR ID, c) Late days used for this assignment, d) Total late days used so far.
3. Your code, organized by question, and properly commented.

Both deliverables described above should be uploaded as a single archive on iLearn.

Deadline: Please check the class website for the most current date. The deadline is on 11:59pm on the day specified.

Late policy: Please check the class website for the late policy.

Grade distribution: Unless noted otherwise, the number of points for each question are equally distributed to each sub-question.

Academic Integrity: Each assignment should be done *individually.* You may discuss general approaches with other students in the class, and ask questions to the TAs, but *you must only submit work that is yours*. If you receive help by any external sources (other than the TA and the instructor), you must properly credit those sources, and if the help is significant, the appropriate grade reduction will be applied. If you fail to do so, the instructor and the TAs are obligated to take the appropriate actions outlined at http://conduct.ucr.edu/policies/academicintegrity.html. Please read carefully the UCR academic integrity policies included in the link.