CS 4661: Introduction to Data Science
Dr. M. Pourhomayoun
Homework3
Due Date: Fri, Nov 1

Write and submit your python codes in “Jupyter Notebook”. **<u>Please upload 2 separate ipynb files (One for Question1, and One for Question2).</u>**

<u>Question1: Predicting Heart Disease</u>

In this question, we work with a dataset from the great textbook of "An Introduction to Statistical Learning."

a- Read the data file “Hearts_s.csv” (from github using the following command), and assign it to a Pandas DataFrame:
df = pd.read_csv("https://github.com/mpourhoma/CS4661/raw/master/Heart_s.csv")

b- Check out the dataset. As you see, the dataset contains a number of features including both contextual and biological factors (e.g. age, gender, vital signs, …). The last column “AHD” is the label with “Yes” meaning that a human subject has Heart Disease, and “No” meaning that the subject does not have Heart Disease.

c- As you see, there are at least 3 categorical features in the dataset (Gender, ChestPain, Thal). Let’s ignore these categorical features for now, only keep the numerical features and build your feature matrix and label vector.

d- Split the dataset into testing and training sets with the following parameters: test_size=0.25, random_state=6.

e- Use **KNN (with k=3), Decision Tree (with random_state=5),** and **Logistic Regression** Classifiers to predict Heart Disease based on the training/testing datasets that you built in part (d). Then check, compare, and report the accuracy of these 3 classifiers. Which one is the best? Which one is the worst?

f- Now, we want to use the categorical features as well! To this end, we have to perform a feature engineering process called OneHotEncoding for the categorical features. To do this, each categorical feature should be replaced with dummy columns in the feature table (one column for each possible value of a categorical feature), and then encode it in a binary manner such that only one of the dummy columns can take “1” at a time (and zero for the rest). For example, “Gender” can take two values “m” and “f”. Thus, we need to replace this feature (in the feature table) by 2 columns titled “m” and “f”. Wherever we have a male subject, we can put “1” and ”0” in the columns “m” and “f”. Wherever we have a female subject, we can put “0” and ”1” in the columns “m” and “f”. (Hint: you will need 4 columns to encode “ChestPain” and 3 columns to encode “Thal”).

g- Repeat parts (d) and (e) with the new dataset that you built in part (f). How does the prediction accuracy change for each method?

h- Now, repeat part (e) with the new dataset that you built in part (f), but this time using ***Cross-Validation***. Thus, rather than splitting the dataset into testing and training, use 10-fold Cross-Validation (as we learned in Lab4) to evaluate the classification methods and report the final prediction accuracy.

Question2: Debt Prediction

In this question, we work with another dataset from the textbook of "An Introduction to Statistical Learning."

a- Read the dataset file “Credit.csv” (from github using the following command), and assign it to a Pandas DataFrame:
df = pd.read_csv("https://github.com/mpourhoma/CS4661/raw/master/Credit.csv")

a- Check out the dataset. The “Credit” dataset includes the “balance” column (average credit card debt for a number of individuals) as target, as well as several features: age, cards (number of credit cards), education (years of education), income (in thousands of dollars), limit (credit limit), marital status, and rating (credit rating).

b- Generate the feature matrix and target vector (target is “balance” in this dataset). Then, **normalize** (scale) the features (**note**: **don’t normalize the target vector!**). To normalize the data, you can simply use ***preprocessing.scale*(X)** from sklearn.

c- Split the dataset into testing and training sets with the following parameters: test_size=0.24, random_state=9.

d- Use Linear Regression to train a linear model on the training set. Check the coefficients of the linear regression model. Which feature is the most important? Which feature is the least important?

e- Predict “balance” for the users in testing set. Then, compare the predicted balance with the actual balance by calculating and reporting the **RMSE** (as we saw in lab tutorial 4).

f- Now, use 10-fold Cross-Validation to evaluate the performance of a linear regression in predicting the balance. Thus, rather than splitting the dataset into testing and training, use Cross-Validation to evaluate the regression performance. What is the **RMSE** when you use cross validation?