National University of Computer & Emerging Sciences, Karachi
Fast School of Computing
Mid Term-II Spring-2021
19th April 2021, 08:15 AM – 09:30 AM

Course Code: CS481	**Course Name:** Data Science
Instructor Name / Names: Dr. Muhammad Nouman Durrani / Muhammad Sohail Afzal	
Student Roll No:	**Section:**

Instructions:

- **For getting Datasets and other necessary files:**
  Press Windows key and type \\filestorage in the Run App.
  You can also directly access the file storage folder by typing 172.16.5.41 in the browser or \\172.16.5.41 in the Run App.
  Copy all necessary files from the **"Data_Science"** folder.
- **For Mid II Exam. submission:**
  Copy all your code files in one folder. You **MUST** rename this folder as your Student ID.
  Now open this IP address: 172.16.25.30 in browser or \\172.16.25.30 in the Run App.
  Now open the folder with name **"DS_Submissions"** and submit your folder in your respective section folder.
- **Files submitted after the Due Time (9:30 AM) will not be considered.**
- Attempt all the questions.
- After completion of the exam, return the question paper.
- Your Student ID **MUST** be written on the paper.
- In case of any ambiguity, you may make an assumption. But your assumption should not contradict any statement in the question paper.

Time: 75 minutes. **Total Marks:** 12.5

Question 1: **[Marks: 3.5 + 3 = 6.5]**

Suppose you are working on the Iris dataset that contains 150 observations and 5 variables. Variables "Sepal_length, Sepal_width, Petal_length, Petal_width" are quantitative variables describing the length and widths of parts of flowers in cm. Variable "Species" is a categorical variable that consists of three different species namely, Setosa, Versicolor, and Virginica.

We want to see if our classifiers (K-NN in part (a) and Decision Tree in part (b)) are correctly able to predict the Species class a flower belongs to based on the **Sepal_length, Sepal_width, and Petal_width only**.

a) Apply **K-NN classifier** on above three selected attributes (where K or n_neighbors = 1-10) if Manhattan distance (as the distance metric) = $|(x_2 - x_1)| + |(y_2 - y_1)|$ and 5-fold Cross-validation is used. You will apply KNN multiple times putting K = 1-10 (where K is the number of nearest neighbors). Print "highest accuracy" and also print "value of K" (number of nearest neighbors) on which it has the highest accuracy.

b) Apply **DecisionTree classifier** on the above three selected attributes using Entropy with pruning = 0.014 and 10-fold Cross-validation. Print the classifier accuracy and classification report.

Question 2: **[Marks: 3.5 + 2.5]**

Consider the **diabetes dataset**. The dataset consists of eight (8) baseline variables "Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction, Age" and one categorical output variable "Outcome".

a) Suppose we want to apply **K-Means** clustering on the the baseline variables **"Pregnancies, Glucose, BloodPressure, Insulin, BMI"** ONLY to find possible clusters in the data.
Also, in the given dataset, there is a lot of variation in the magnitude of the data. Variables like Pregnancies have low magnitude whereas variables like Glucose BloodPressure etc. have a higher magnitude. So you are supposed to **standardize** the dataset as well.

Apply KMeans with the following criterion:

- Pick the initial centroids using kmeans++.
- Draw elbow curve after applying Kmeans algorithm choosing the number of clusters N =1 to 10.
- After analyzing the elbow curve, apply KMeans algorithm on the appropriate(best possible) value of N, and visualize the clusters.

b) In this problem, we want to see if there is any linear relationship between the two attributes "Glucose and Blood Pressure" in the **diabetes dataset**.

Apply Linear Regression with the following criterion:

- Use hold out Cross-Validation with 70 / 30 split.
- Find root mean square error (RMSE).
- Visualize the relationship between "Glucose and Blood Pressure".

Appendix

You may use these libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.formula.api as smf

from sklearn.model_selection import cross_val_score, StratifiedShuffleSplit
from sklearn.model_selection import  train_test_split,KFold

from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import LinearRegression
from sklearn.cluster import KMeans

from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import LabelEncoder

from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score
```