**Project Overview**

As part of the project for the Machine Learning Techniques for AI and Data Analytics course, you need to **attempt** **one of the following four** **assignments and submit the response file through the staff-graded assignment.** You may consider attempting the remaining three for practice. 

**Note**: To complete the assignment, each team can perform the analysis using any of the two programming languages covered in the course, Python or R.

**Assignment 1**

In this assignment, you are required to train a machine learning (ML) model for credit card fraud detection using the anonymized credit card transactions dataset, which is provided in a CSV file—C*reditcard.csv*.

Feature **'Class'** is the responsevariable, and it takes the value ‘1’ in case of fraud and ‘0’ otherwise.

Features V1, V2, …, and V28 are the principal components obtained with PCA. The only features that have not been transformed with PCA are 'Time' and 'Amount.' The feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount.

**Assignment 2**

In this assignment, you are required to train a machine learning (ML) model for heart attack classification using the Heart Attack Analysis and Prediction dataset, which is provided in a CSV file — *Heart.csv*.

Feature ‘**target’** is the responsevariable, and it takes the value ‘0’ when there is less chance of heart attack and ‘1’ when there is more chance of heart attack.

The predictor variables are ‘Age’ of the patient, ‘Sex’ of the patient, ‘exang’ or the exercise-induced angina (1 = yes; 0 = no), ‘ca’ or the number of major vessels (0-3), ‘cp’ or the  Chest Pain type (here, Value 1 represents typical angina, Value 2 represents atypical angina, Value 3 represents non-anginal pain, and Value 4 represents the asymptomatic), ‘trtbps’ or the resting blood pressure (in mm Hg), ‘chol‘ or the cholesterol in mg/dl fetched via BMI sensor, ‘fbs’ or the (fasting blood sugar > 120 mg/dl) (1 = true; 0 = false), ‘rest\_ecg’ or the resting electrocardiographic results (here, Value 0 represents normal, Value 1 represents having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV), Value 2 represents showing probable or definite left ventricular hypertrophy by Estes' criteria, and ‘thalach’ or the maximum heart rate achieved.

**Assignment 3**

In this assignment, you are required to train a machine learning (ML) model to predict the house price using the Boston Housing Data dataset, which is provided in a CSV file—*HouseData.csv*.

Feature **“MEDV’’** is the response variable, representing the median value of owner-occupied homes in $1000s.

Features used for predicting the value of a home in a particular locality are as follows:

1. ZN           → proportion of residential land (for lots over 25000 sq. ft.)
2. INDUS      → proportion of non-retail business acres per town
3. NOX         → nitric oxides concentration in the locality (parts per 10 million)
4. RM           → average number of rooms per dwelling
5. AGE         → proportion of owner-occupied units built prior to 1940
6. DIS           → weighted distances to five Boston employment centers
7. RAD         → index of accessibility to radial highways
8. TAX          → full-value property-tax rate per $10,000
9. PTRATIO  → pupil-teacher ratio by town

**Assignment 4**

In this assignment, you are required to train a machine learning (ML) model for wine type classification using the Wine dataset, which is provided in a CSV file—*Wine-Clusterring.csv*.

This data results from a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the **three types of wines**.

The attributes are the contents of :

* Alcohol
* Malic acid
* Ash
* Alkalinity of ash
* Magnesium
* Total phenols
* Flavonoids
* Non-flavonoid phenols
* Proanthocyanidins
* Color intensity
* Hue
* OD280/OD315 of diluted wines
* Proline