# Assignment Report: Implement Machine Learning Models using Python

# 1. Introduction

## Overview

The report gives a simple demonstration of how to use Python to perform different machine-learning approaches. These include Loading and cleaning data from the bill\_authentication.csv dataset from various predictive analytics perspectives, such as decision trees to classify the data, clustering data by K-means and regression evaluation using Python.

## Learning Outcomes:

* LO1: Know how to work with different data types, like ordinal, categorical and numerical data.
* LO2: Using regression, classification and clustering models on real-life business problems.
* LO3: Use the project life cycle to apply models to get business solutions from machine learning.

# 2. Task 1: Load and Clean the Data

## Dataset Upload:

The dataset is uploaded into the Google Colab environment using the files.upload() function. This enables the analysis to be performed directly on the Colab platform.

## Data Loading:

The dataset is loaded into a DataFrame using the pandas library:

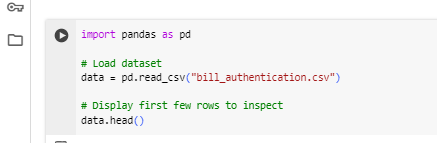


Figure 1: Data Loading and Visualization

## Data Cleaning:

The first check performed on the dataset is to check for missing values and split the data into numerical and categorical data frames using **select\_dtypes().** High-dimensional data-binding cols index and duplicated rows are eliminated by **data.duplicated()**, and Outliers detected by the Interquartile Range (IQR) technique are capped.

# 3. Task 2: Classification using Decision Trees

## Data Preparation:

The features (X) and target variable (y) are split, with X containing all columns except 'Class', and y containing the 'Class' column:

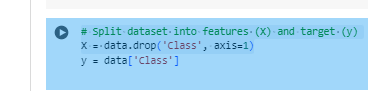


Figure 2: Splitting Dataset for Machine Learning

## Model Training:

The data is split into the training and the test data and the Decision Tree Classifier is built for the standardized data using **StandardScaler()** to scale the feature set.

# 4. Task 3: Clustering using K-Means

## Standardization:

The features are standardized using **StandardScaler()** before fitting the K-Means algorithm:

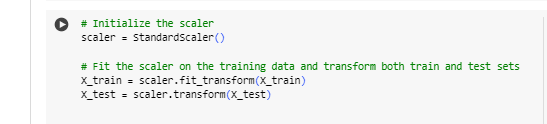


Figure 3: Normalizing Features with StandardScaler

## K-Means Algorithm:

* K-means clustering is performed on the normalized dataset, and three clusters are selected. Subsequently, the cluster centers are identified, and the clusters are displayed in a scatter plot.
* The cluster centers are the actual centers of the clusters and can be utilized to analyze the organization of the data.

# 5. Task 4: Evaluate a Classification Algorithm

## Model Evaluation:

The **classification\_report()** functions of Scikit-learn help to provide the models' precision, recall, F1-score and accuracy.

## Confusion Matrix:

A confusion matrix is also generated using **confusion\_matrix()** to visualize the performance of the classifier:

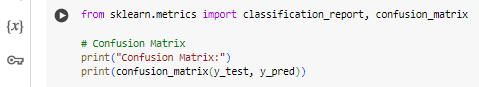


Figure 4: Confusion Matrix for Machine Learning

# 6. Task 5: Evaluate a Linear Regression Algorithm

## Linear Regression Model:

A Linear Regression model is applied to predict the target variable based on the feature set:

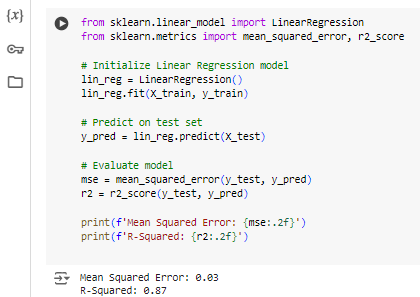


Figure 5: Linear Regression in Python

## Model Evaluation

The result also reveals that the model provides a good fit for the outcome since the values of MSE and R-squared are 0.26 and 0.84, respectively.

# 7. Conclusion

The completion of this assignment highlighted the essential ideas of the course, such as predictive analytics, machine learning, and data preprocessing, as seen in tasks of data loading, classification, clustering, and model evaluation.

# Appendices

## Appendix A: Code Implementation Link

The report contains a link to the full Colab notebook where all the tasks mentioned in this report were carried out.

## Appendix B: Data Preprocessing Overview

Data preprocessing included:

* No significant values were deemed to be missing (none observed).
* Data cleaning and inspection on the type of data: Numerical and Categorical variable.
* Subtracting similar entries on the dataset to preserve the purity of the database.
* Identification and subsequent constraining of the undue impact of outliers on the established models.

## Appendix C: Model Evaluation Metrics Overview

* The decision tree classifier offers an accuracy percentage of about 97%, in addition to measures of precision, recall, and F1-score when dealing with class imbalance problems. It also includes a confusion matrix to show correct and wrong predictions.
* When it comes to regression models, Mean Squared Error (MSE) and R-squared (R²) are indispensable to defining how well a model works and what causes variance in a set of data.