Deliverable 3

We began the project by importing the necessary libraries and loading the dataset , we have no missing data, so we did not need to apply any code for the fill missing data or fill Na.

Next, we split the dataset into training and testing sets. This is a crucial step in evaluating the performance of a machine learning model. We used an 80/20 split — 80% of the data for training and 20% for testing.

In Question 13, we analyzed the relationship between two variables using simple linear regression by selecting the Unit Price as the independent variable and Total as the dependent variable. We configured the linear regression model and trained it on the training data. Then, we used it to predict the test results. The model predicted a value of 494.24, while the actual value was 507. We visualized the training and test results and derived the linear regression equation Total = 1.338910 + 5.9 × Unit Price

Following that, we applied multiple linear regression, which allows us to analyze the relationship between one dependent variable and multiple independent variables. In our case, the independent variables are Unit Price, Quantity, and Tax (5%), and the dependent variable remained Total. We reused the initial preprocessing steps and trained the multiple linear regression model on the training set. We then predicted values for the test set and for specific inputs, which gave us accurate results. The final equation from the multiple linear regression model is Total = (-4.60e-15 × Unit Price) + (-2.49e-14 × Quantity) + (2.10e+01 × Tax 5%) + 2.27e-13

After completing regression analysis, we moved on to classification, where we used Logistic Regression to predict a categorical dependent variable — in this case, Customer Type. The independent variables used were Total, Gross Income, Rating, Unit Price, Quantity, and Tax. Logistic regression models the probability of a class using a linear combination of the independent variables.

We followed the standard steps: importing libraries, training the model, predicting the test set, and evaluating the results. To assess model performance, we used a confusion matrix and calculated the accuracy score. We also tested the model’s ability to predict a specific input case.

We have repeated the same process with other classification models: K-Nearest Neighbours (KNN), Naïve Bayes and Decision Tree

Evaluate model performance using metrics like accuracy, precision, regression and F1-score where we select the best-performing model based on these metrics. Each model produced accuracy scores between 54% and 57%. Among them, the Decision Tree model gave the highest accuracy, making it the best fit for our classification task.