MS4610: INTRODUCTION TO DATA ANALYTICS

Group 27

Group Members:	Rishabh Shah	NA17B116
	Mansi Khandelwal	NA17B004
	Manish Patidar	NA17B111
	Harshita Ojha	BS17B012
	Landge Hrushikesh Sandip	NA17B109

Project Report: Loan Default Prediction

Table of Contents

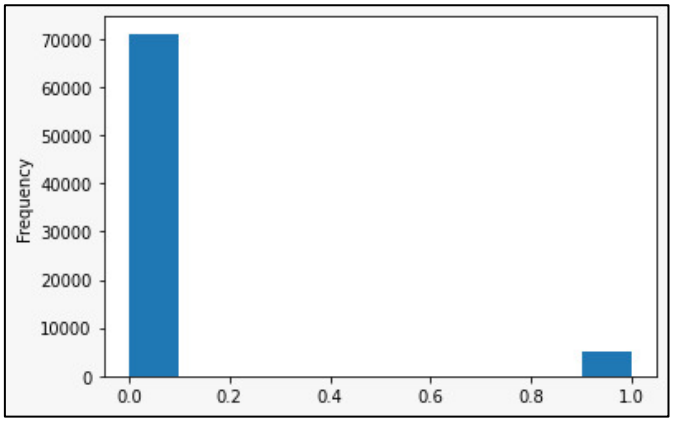
Data Information:

The dataset consists of the following details on loans taken by different customers:

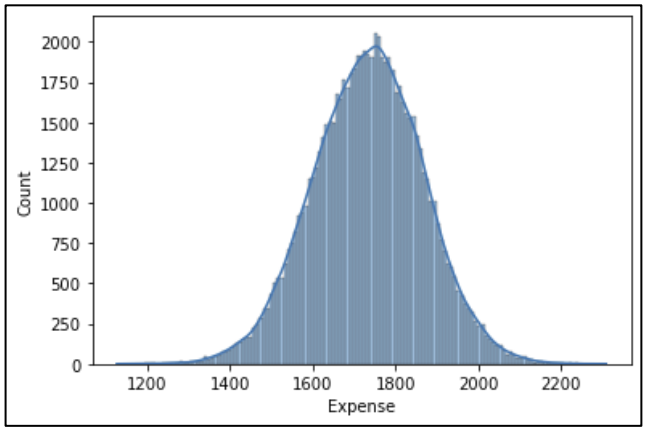
- **ID**: A unique identifier for every financial loan that is being considered.
- **Loan type**: Type of loan taken (Two types, 'A' or 'B').
- **Occupation type**: Occupation of the customer (Three occupation types, 'X', 'Y', 'Z').
- **Income**: A continuous variable that is indicative of the annual income of the customer. This is not the exact income value.
- **Expense**: A continuous variable that is indicative of the annual expense of the customer. This is not the exact expense value.
- **Age**: Age of customer – Value of '0' is considered as below 50, and the value of '1' is considered as above 50.
- **Score1, Score2, Score3, Score4, Score5**: Represents five different metrics calculated by the organization, about the customer and the loan that is being considered.
- **Label**: '0' means non-default, and '1' means default on that loan.

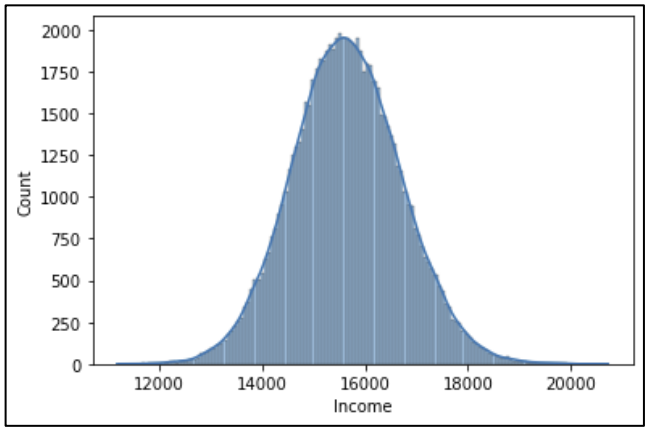
Data set description:

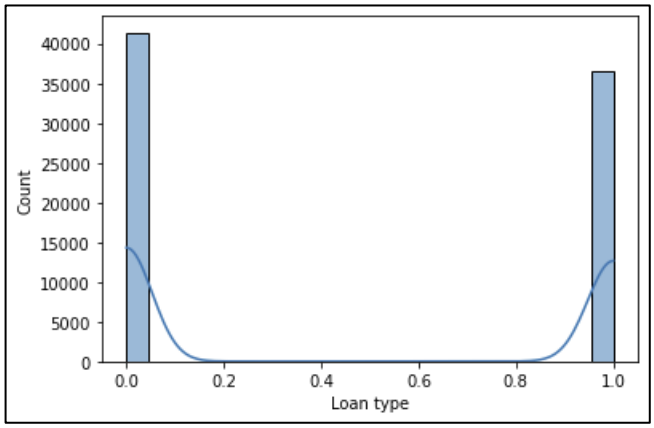
- The given training data has $80{,}000$ samples.
- There are $3{,}903$ missing labels in training data.
- Binary classification problem, the target variable is Label for defaulters and non-defaulters
- Unbalanced classification problem: The given training data is unbalanced as it has unequal instances of different classes. 1 is a minority class and 0 label has a significantly larger number of instances.

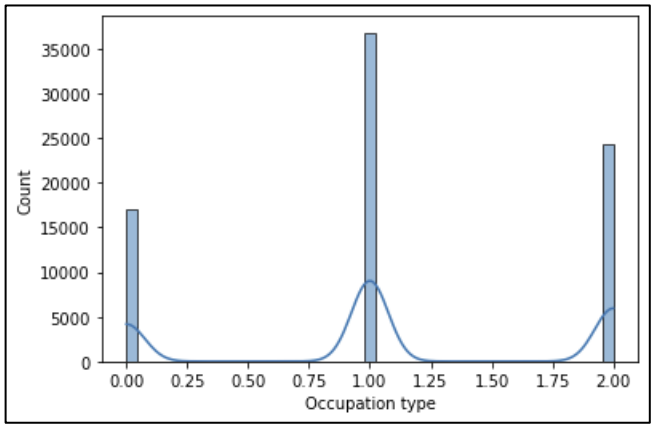


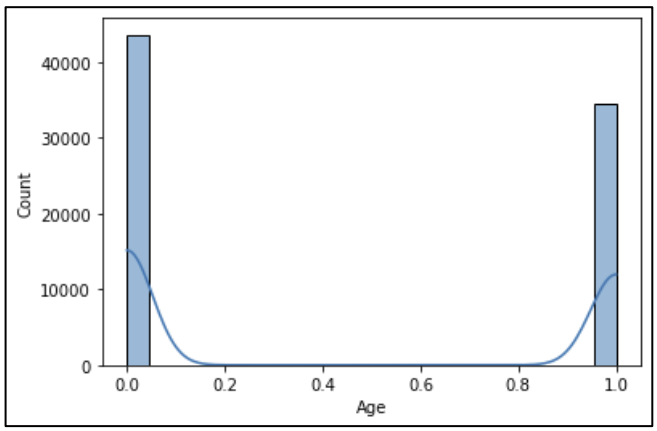
- Features:

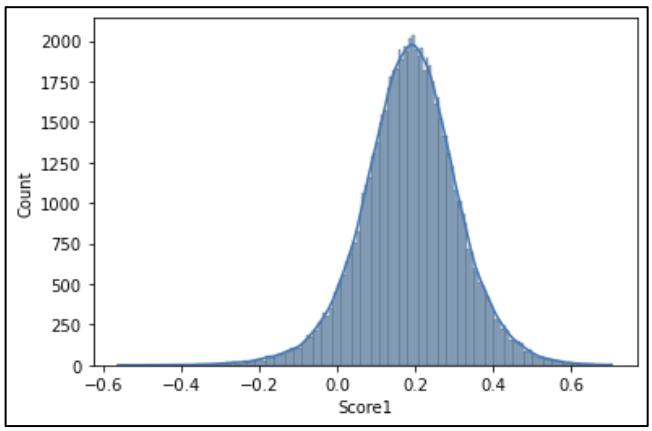


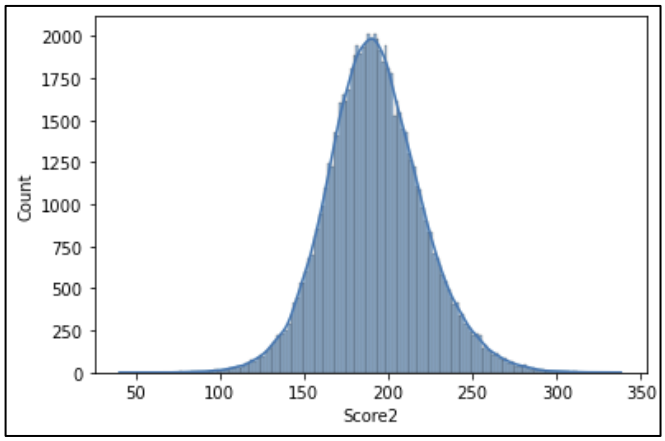


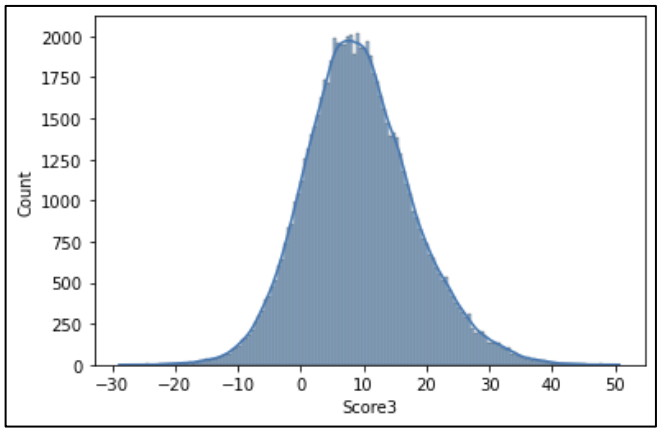

Count
Loan type

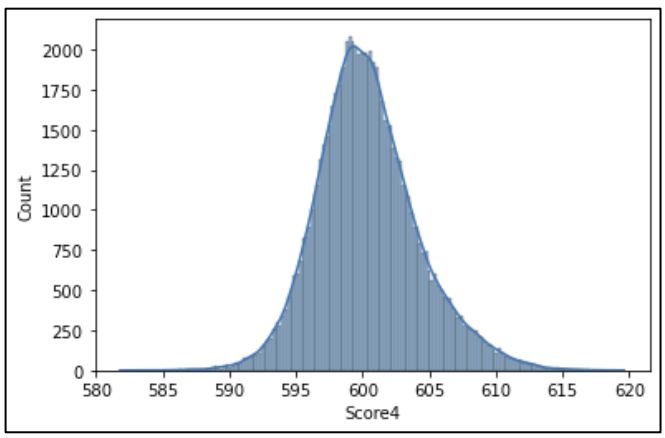

Count
Occupation type

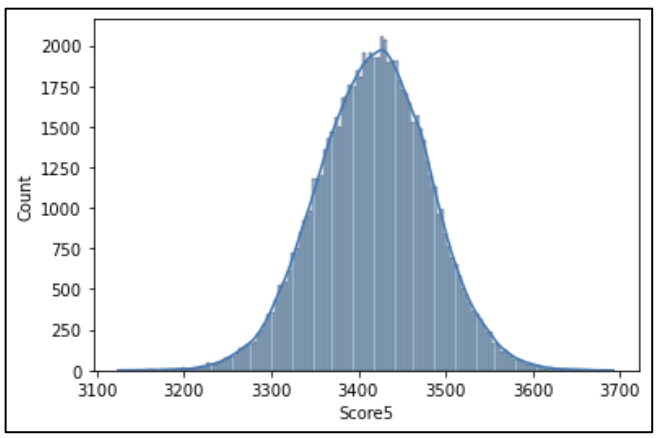

Count
Age


Count
Score1

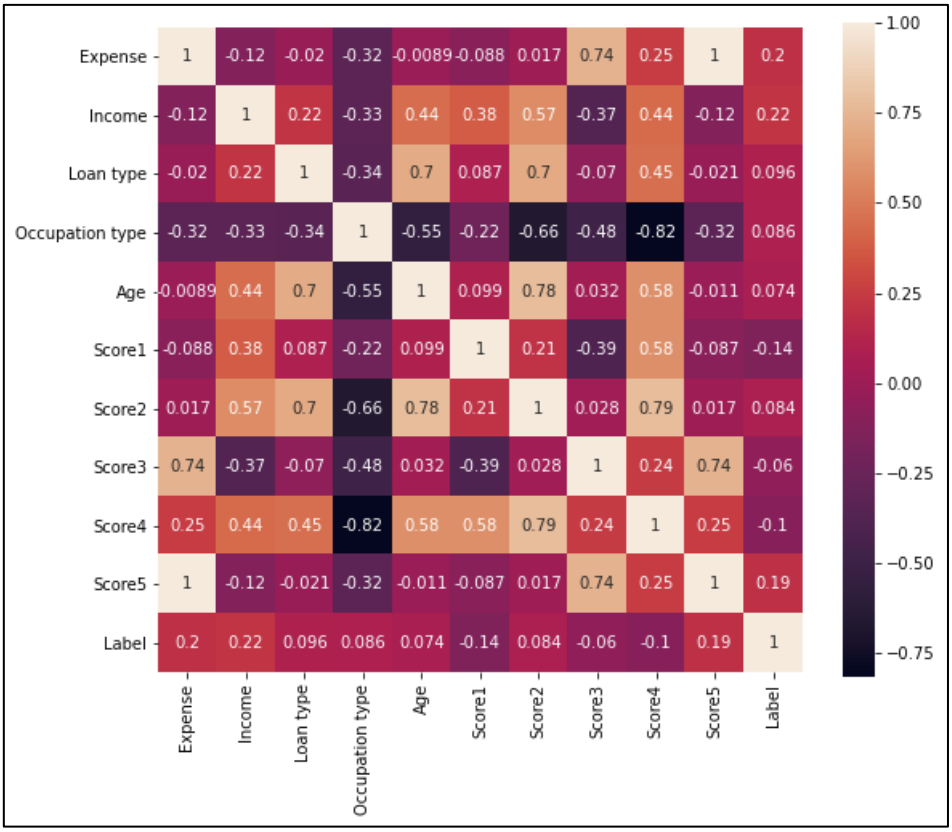

Count
Score2


Count
Score3


Count
Score4


Count
Score5

- Feature Correlation:
  - One way to try and understand the data is by looking for correlations between the features and the target. We can calculate the Pearson correlation coefficient between every variable and the target using the $.corr$ data frame method.
  - Positive correlation implies that as the value of a feature increases the person is less likely to default on a loan.
  - From the correlation matrix shown below, we can infer that the features expense, income and score 5 are highly correlated to the target variable while the features score 1, score 3 and score 4 have a negative correlation with it.



• Data Pre-processing:

▪ Removing unrelated features:

The ID column in the dataset is dropped as it is assumed that it does not directly impact the default/non-default tendency of the users.

▪ Encoding Categorical Features:

In our dataset, the features 'Occupation' and 'Loan Type' are Label encoded to make them compatible with machine learning models which are mathematical models.

▪ Imputation of Missing Values:

The given training data contains missing or NaN values in the features which need to be imputed before machine learning models can be applied to the data. We have used the MissForest Imputation technique which operates on the Random Forest Algorithm. The imputation technique is used with the 'balanced' class weight parameter to account for the unbalanced nature of the data.

- Removing Samples with Missing Labels:

  The training dataset contains 3903 samples which do not have a label associated with them. At this stage, we drop these rows from our training data and it would later be reused after finding a suitable model for the rest of the training data (Self-training method, semi-supervised learning).

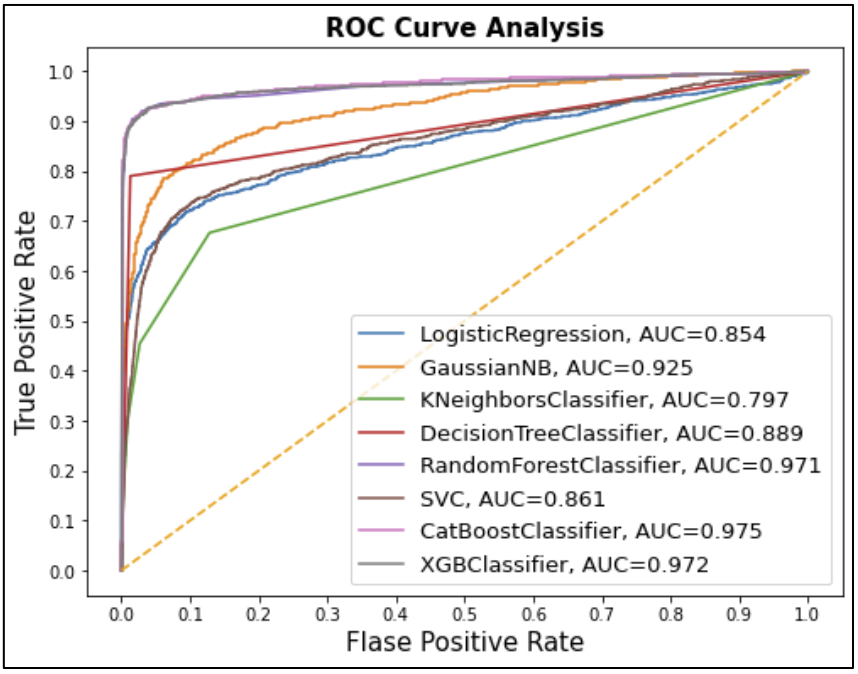
Model selection:

We implemented various models on the training dataset to find a suitable model for our binary classification problem keeping in mind the unbalanced nature of the training dataset.

- Models Implemented:
  - Logistic Regression
  - GaussianNB Classification
  - KNN Classification
  - Decision Tree Classification
  - SVC Classification
  - Random Forest Classification
  - XGBoost Classification
  - CatBoost Classification

- Best Model:

  To compare and select the best model out of the models implemented we use the AUC ROC curves as shown below:



  As evident from the AUC ROC curve above, the CatBoost Classification model works best with the training dataset.

- CatBoost Model:

  CatBoost is a method based on Gradient Boosting. It divides the dataset into random permutation and applies ordered boosting on those random permutations. It can

outperform XGBoost both in simplicity of application and prediction results. Some of the advantages of CatBoost:

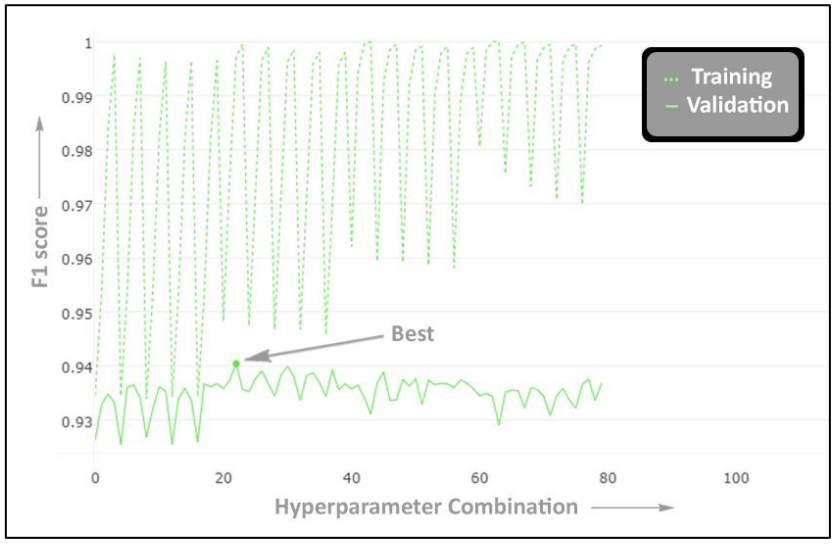
- It is easier to implement and very powerful as compared to other Boosting algorithms like XGBoost and Light GBM, as CatBoost implements symmetric trees which helps in decreasing prediction time.
- CatBoost is more resilient to overfitting compared to other popular algorithms, which makes our model more generalised

- Training:
  - Model Tuning:

After model selection, we fine-tune the hyperparameters of the CatBoost Classification model using Grid Search technique to obtain the best set of parameters. The evaluation metric used for hyperparameter training was F1 score. In order to handle class imbalance, we set the class_weight parameter to $[1, 14]$, i.e., the majority class is ~ 14 times the size of minority class. The optimal values found for the following hyperparameters are:

- depth $-$ 6
- l2_leaf_reg (L2 regularisation parameter) $-$ 1
- learning_rate $-$ 0.1

Using these hyperparameter, we obtain an F1 score of 0.94 as seen in the graph below.



  - Self-Training the model:
    - Predicting missing labels in training data:

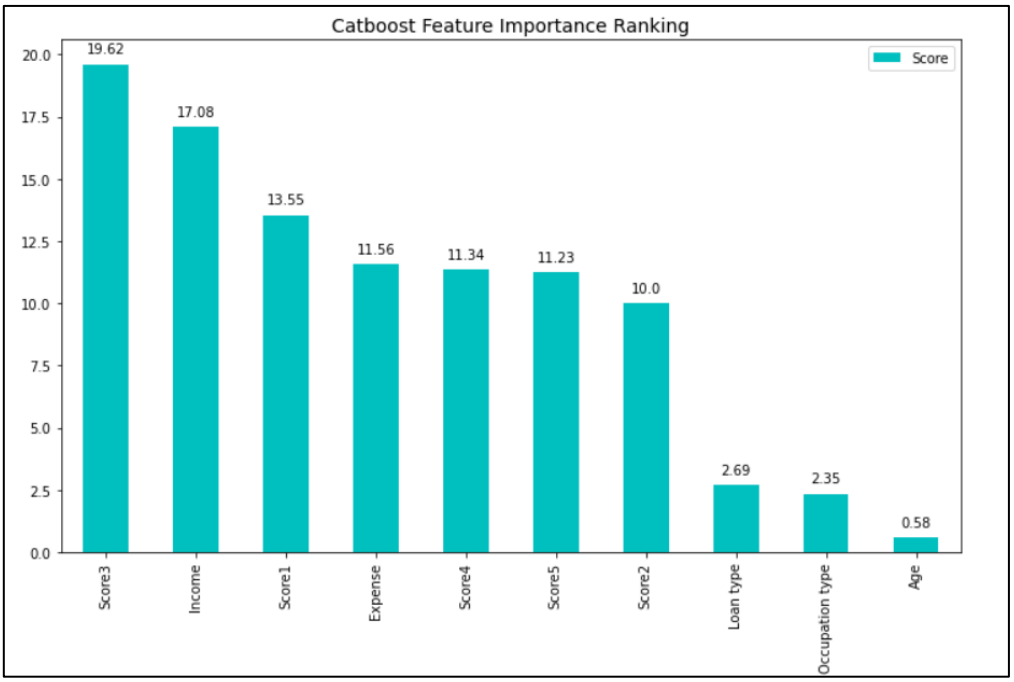
We used the tuned CatBoost Classifier model obtained in the last step to predict the missing labels in the training data. After predicting these labels, the labels which have a prediction probability greater than 0.95 are appended to the training dataset. There are 3725 such samples whose labels are predicted with more than a 0.95 prediction probability (confidence). The rest of the samples with a prediction probability lesser than 0.95 are discarded which is only $\sim 0.26\%$ of the entire training data.

    - Re-training the model:

The model is trained on the updated training dataset obtained in the last step.

- ## Feature Importance:

  For each feature, there is individual importance value (the default importance feature calculation method is for non-ranking metrics), PredictionValuesChange displays the average prediction changes against changes in the feature value. Larger the value of importance, larger on average is the change to the prediction value if this feature is changed. Using the CatBoost Feature Importance Ranking technique we got the following data on the feature importance-



- ## Predicting Labels for test data:

  After pre-processing of the given test data (Removing ID column and label encoding categorical features), the model finally obtained in the last step is used to predict the labels of the test dataset provided in the problem statement.

- ## Results and Conclusions:

  - We predicted the test data to contain 13,562 samples of minority class (1) and the rest as majority class (0).
  - Feature importance was calculated using the PredictionValueChange and the feature 'Score3' was found to be of significant importance while the feature 'Age' was found to be the least significant
  - CatBoost performs best on our training dataset and gives us an AUC value of 0.975.
  - Accuracy isn't a good evaluation metric for hyperparameter tuning; therefore, we have made use of F1 scores and AUC ROC curves. For instance, accuracy of models with an F1 score of 0.81 was coming out to be 0.98.
  - Class imbalance constraints the learning capability of any classification model and the prediction accuracy of the minority class is compromised.
  - The Pearson correlation factor of multiple features in our dataset with the target variable was not close to 1 or -1 yet the feature importance was significant, implying that there is a non-linear relationship between the target and these variables. Thus, we need a more complex model than simple linear regression to perform the classification task more accurately.