DATA MINING TECHNIQUES

Project Review

Topic: Potential Loan defaults Prediction

Course Code: ITE2006

Faculty: Mr. Ramkumar T

Member 1: Kunj Patel (19BIT0256)

Member 2: Narayanan (19BIT0259)

Member 3: Robin Singh (19BIT0265)

Data mining functionalities focused and platforms used:

Classification was the main data mining functionality implemented in our project. It is the problem of identifying the class of an observation, on the basis of a training set of data containing observations and whose classes are already known.

There are two stages in classification. First, we build the model, and train it to predict accurate results and then, we test the built model using test data to find its accuracy.

Classification is applied in many fields. Recognising handwritten digits, identifying spam mails and image segmentation are few examples where classification is applied.

We built the model and trained it on Google Colab.

Colab link: https://colab.research.google.com/drive/1v-sVCaBNWpbO16grCEcSSaU_ZoBITmU8

Benchmarking Dataset used:

We used a Loan defaults dataset from Kaggle, that had 13 columns and 614 rows. The columns present in the dataset were Loan_ID, Gender, Married, Dependents, Education, Self_Employed, ApplicantIncome, Coapplicant income, LoanAmount, Loan_Amount_Term, Credit_History, Property_Area and Loan_Status. Of all the columns, ApplicantIncome, CoapplicantIncome, LoanAmount, Loan_Amount_Term and Credit_History were continous variables, and the rest were categorical variables. Loan_Status was taken as the target class for our analysis.

Link: https://drive.google.com/drive/folders/11go418mxPSdazU3fWku2ySINr_Ni4aNE?usp=sharing

Data Pre-processing techniques:

Data Cleaning

- The presence of duplicate values in the dataset was checked using the duplicated.any() function, and we obtained the result as FALSE.
- The number of null values that were present in each column of the dataset was obtained using the isnull.sum() and the result was TRUE.
- For categorical data, the missing values were filled with the most frequently used values
- For continous data, the missing values were filled with values in the previous row of the same column.

Data Transformation

- The values in target column, Loan_Status were transformed from Y and N to 1 and 0 respectively.
- Credit_History was changed to Object data type.

Data Reduction

- The Loan_ID column was removed as it was not required for our analysis.

Data mining Algorithms:

Decision Tree classifer:

Decision tree is one of many supervised learning algorithms, which takes decisions on the basis of a set of rules. The key concept behind Decision tree is that Yes/No questions are created within the dataset, which splits the data into branches. This in turn, organizes the data in the form of a tree, hence the name Decision Tree is obtained.

This algorithm is easy to implement as we do not need to focus a lot on data pre-processing. It is one of the best algorithms for visual representation, and feature selection happens automatically.

But, it takes a long time to train the model, and is relatively expensive as the complexity and time taken are more.

Naive Bayes:

The Naive Bayes is a probabilistic classification algorithm, which assumes that the presence of any feature in a class is not related to the presence of any other feature.

The main advantages of this algorithm is that it is easy to build, and is very useful when we have large datasets. It can perform very well even with less training data, when compared to other models.

However, it assumes that all features are independent, but in reality, we can hardly find a set of independent features. Another disadvantage is that if the test data set has a categorical variable that wasn't present in the training data set, the Naive Bayes model will assign it zero probability and will not make any predictions. This is known as 'Zero Frequency'.

KNN:

KNN, also known as the K-Nearest Neighbours algorithm, is a supervised machine learning algorithm. It stores all the data and classifies a new data point based on the similarity. So when new data appears, it can be easily classified into a category that best suits it.

The KNN algorithm is simple to implement and can be used when we have large datasets. It is also robust to noisy data.

But, determining the value of K might not be an easy task always. Moreover, as we are calculating the distance between data points for training samples, the cost of computation tends to be high.

Result:

We evaluated each of the models, and obtained the following results:

Decision Tree classifier

Precision- 0.7838

Recall- 0.9775

F1 Score- 0.87

Loss- 6.9615

Accuracy- 0.7984

KNN

Precision- 0.7706

Recall- 0.9438

F1 Score- 0.8485

Loss- 8.0324

Accuracy- 0.7674

Naive Bayes

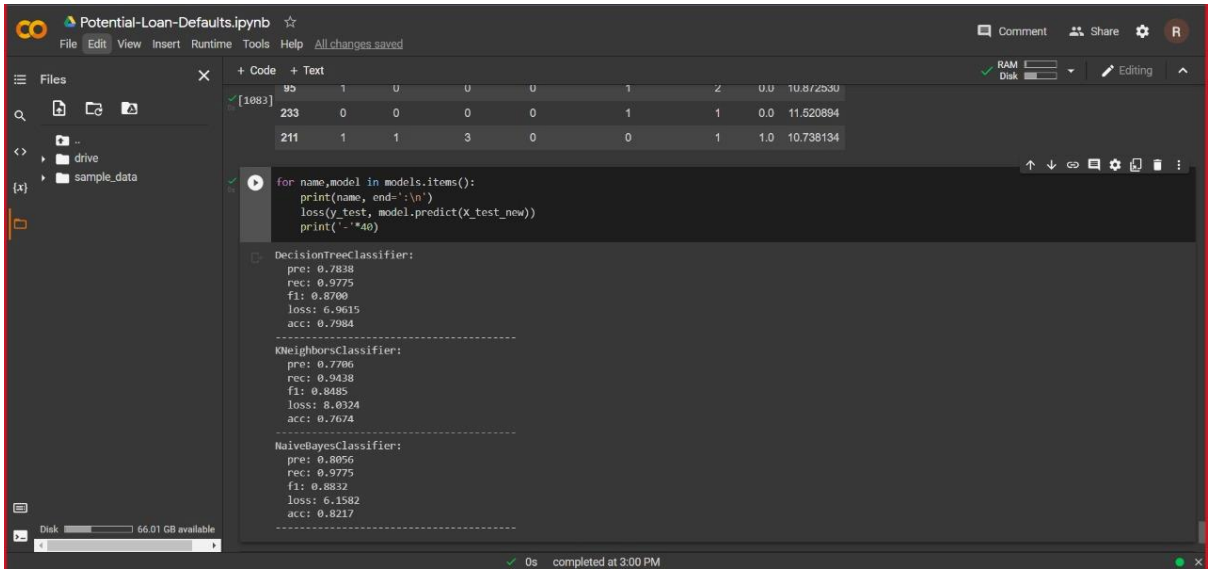
Precision- 0.8056

Recall- 0.9775

F1 Score- 0.8832

Loss- 6.1582

Accuracy- 0.8217



Among the three algorithms, the Naïve Bayes performed well, with a higher accuracy.

Exposure gained through J Component:

We came across various methods that are used to clean data and handle missing values. We plotted several graphs to visualize and understand the data set better. We were also able to train our data set based on various data mining algorithms that come under classification such as - Decision Trees, K-Nearest Neighbors and Naive-Bayes. We then performed cross validation to evaluate our model using different data that we trained the model on. Dimensionality Reduction, Outlier Removal and other Feature Selection methods were implemented in order to fine tune and enhance the performance metrics of the models.