

Loan Default Prediction

Project (Group 8)

TABLE OF CONTENTS

- 1. Project Goal**
- 2. Overview of the data**
- 3. Data Exploration**
- 4. Model Selection**
- 5. Model Development**
- 6. Insights**
- 7. Conclusion**

Project Goal

The dataset was used in this project, which contained information such as Id, loss, and other relevant data. The dataset was preprocessed by performing data cleaning, transformation, and feature engineering techniques to ensure the data was suitable for model training.

This project aims to develop a machine-learning model that can accurately predict loan repayment and identify the factors that contribute to loan repayment. This will enable the lending industry to make informed decisions and minimize the risk of loan defaults, benefiting lenders and borrowers. The project aims to determine whether a loan will default, as well as the loss incurred if it does default, and to anticipate and incorporate both the default and the severity of the resulting losses. The project seeks to build a bridge between traditional banking, where the focus is on reducing the consumption of economic capital, to an asset-management perspective, where the goal is to optimize the risk to the financial investor.

Overview of the data:

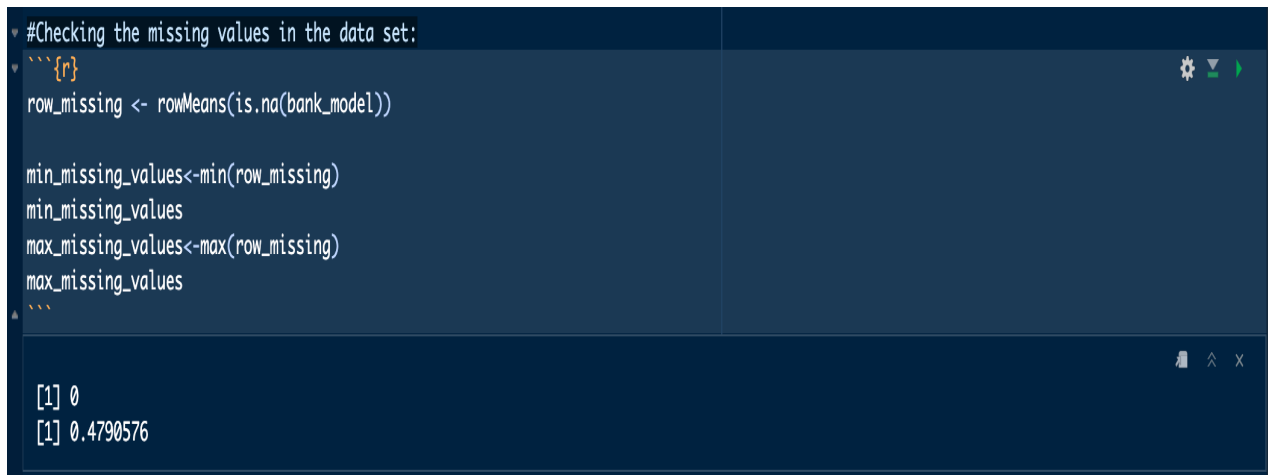
There are two datasets, train_v3.csv and test__no_lossv3.csv. train_v3.csv, which consists of 80000 observations and 764 attributes. The dataset consists of the following attributes :

- Id: This represents a unique customer ID.
- Loss: Amount of loss by a defaulting customer.
- F1- f777: Various factors affect the customer's ability to repay the loan.

test__no_lossv3 consists of 25471 rows and 762 attributes. The dataset consists of the same columns as the training dataset, with an exception being no “loss” column that needs to be predicted from the model.

DATA EXPLORATION:

- We checked for missing values in the train_v3.csv dataset. Below is the representation of the missing values in the dataset.

A screenshot of an R console window with a dark blue background. The console shows the following R code and its output:

```
#Checking the missing values in the data set:
```{r}
row_missing <- rowMeans(is.na(bank_model))

min_missing_values<-min(row_missing)
min_missing_values
max_missing_values<-max(row_missing)
max_missing_values
```
```

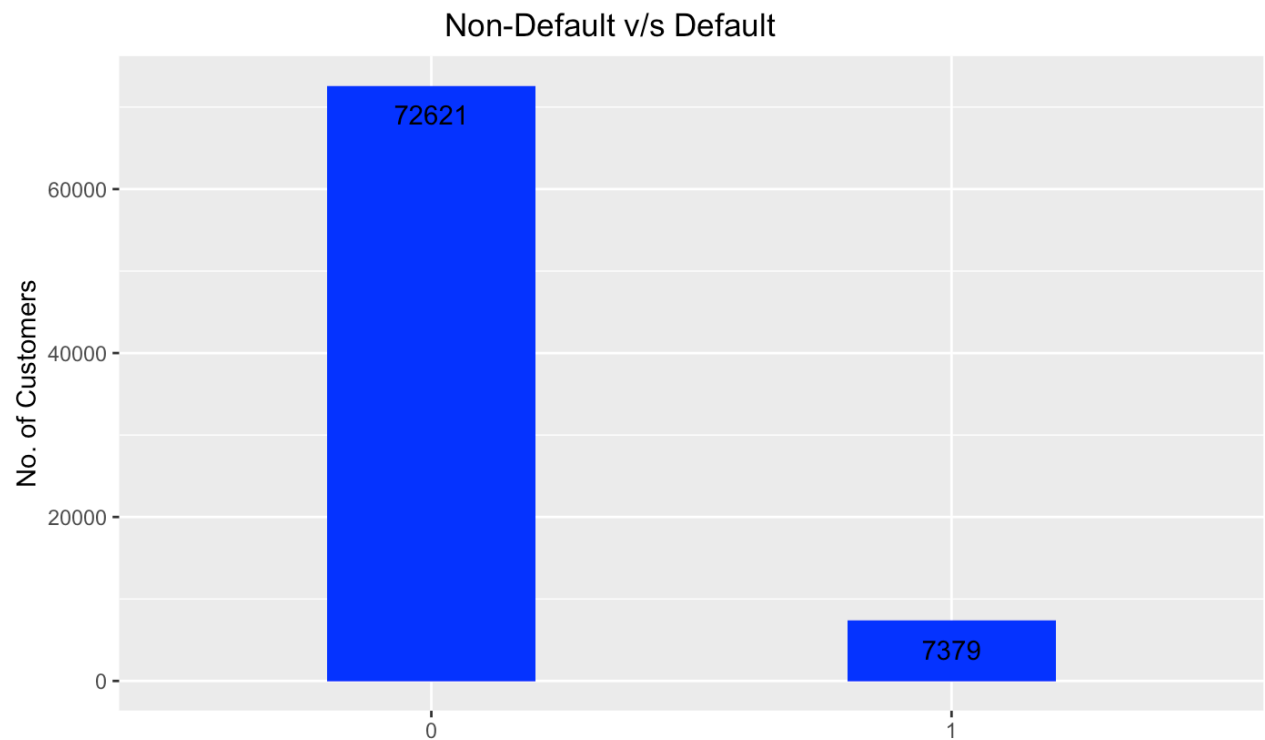
The output at the bottom of the console shows:

```
[1] 0
[1] 0.4790576
```

We can observe that the dataset consists of maximum missing values of 47%.

Further, we also observed that the dataset consisted of zero variance and highly correlated variables.

- We used a ggplot2 library to plot the number of customers defaulting versus those not.



From the above, we can observe that the dataset consists of non-defaulting customers: 72,621 and defaulting customers: 7379.

DATA PREPARATION:

- To proceed with model building, handling the missing values, zero variance variables, and highly correlated variables is important.
- Therefore, we processed the dataset using the “preProcess” function in the caret package to handle data preparation.

```
#Removing the zero-variances variables and Preprocessing the dataset by removing highly correlated and imputing missing values using "corr" and "medianimpute":
```{r}
zero_var_indices <- nearZeroVar(bank_model[, -c(763,764)])

data_cleaned <- bank_model[, -zero_var_indices]

bank_preprocess <- preProcess(data_cleaned[, -c(739,740)], method = c("corr", "medianImpute"))

new_bank_model <- predict(bank_preprocess, data_cleaned)
```

#Removing zero-variance,highly corr variables and imputed missing values : we have new data set "new_bank_model" with 248 attributes.
```

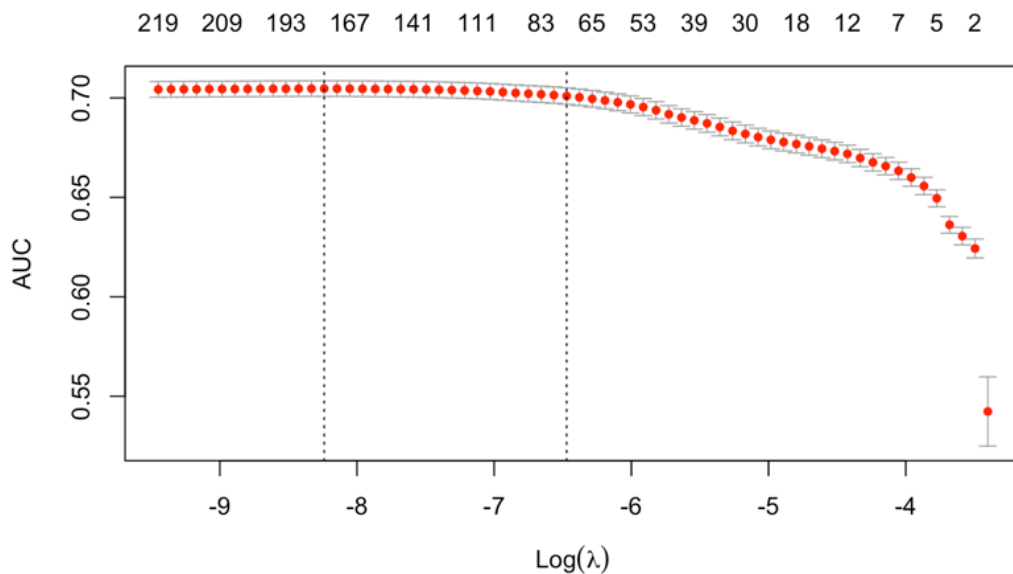
From the above, we can observe that the training dataset was cleaned, resulting in a dataset with 80,000 observations and 248 attributes.

MODEL SELECTION:

- Model selection is a crucial step in machine learning projects, as it involves choosing the most appropriate algorithm to fit the data and make accurate predictions. This project evaluated four machine learning algorithms: Lasso, Principal Component Analysis(PCA), Random Forest model, and Ridge regression model. Lasso and Principal Component Analysis(PCA) were used for variable selection. The dataset consisted of 248 attributes and 80,000 observations, so reducing it for better computation was important.
- The random Forest model was used to classify the dataset to predict how many customers defaulted in the test dataset.
- The Ridge Regression was used for the regression model to predict the loss by the customers to the institution when they defaulted.

MODEL DEVELOPMENT:

- Earlier, we have prepared the data for further processing. We will use the dataset with 80,000 observations and 248 attributes.
- We will apply the Lasso model to identify important attributes in the dataset.
- Following are the results from the Lasso model: The optimal lambda value for the Lasso model was “0.0002640483.”



- The lasso model also resulted in reducing the attributes to 180 attributes.

| Description: df [181 × 2] | | |
|---------------------------|---------------|----------------------|
| | name
<chr> | coefficient
<dbl> |
| 1 | (Intercept) | 7.211311e+00 |
| 28 | f129 | 2.418453e+00 |
| 59 | f268 | 1.679735e+00 |
| 116 | f471 | 1.051069e+00 |
| 178 | f768 | 1.034926e+00 |
| 13 | f57 | 8.977787e-01 |
| 136 | f604 | 8.537344e-01 |
| 25 | f99 | 6.190194e-01 |
| 71 | f298 | 4.921528e-01 |
| 74 | f306 | 4.318747e-01 |

1-10 of 181 rows

Previous 1 2 3 4 5 6 ... 19 Next

We have used the values of the significant coefficients from the Lasso model to reduce the attributes from 248 attributes to 180 attributes.

- We use the PCA model to reduce further the attributes derived from the Lasso model.
- The following are the results derived from the PCA model:

```
#We have 180 variables returned from Lasso model that are stored in "bank_lasso". Now, we further process the variables using PCA.

####{r}
pca_model <- preProcess(bank_lasso[, -c(181)], method = c("center", "scale", "pca"), thresh = 0.80)
pca_model_1 <- predict(pca_model, bank_lasso)
pca_model
####

Created from 80000 samples and 180 variables

Pre-processing:
- centered (180)
- ignored (0)
- principal component signal extraction (180)
- scaled (180)

PCA needed 69 components to capture 80 percent of the variance
```

The PCA model was used with a threshold of 0.80 to capture 80% of the variance of the dataset derived from the Lasso model. It showed that PCA captured 80% of the variance in 69 components.

- The results from PCA were stored in a dataset to perform a classification model using a random forest model.
- The random forest model was applied to the dataset derived from the PCA model.
- The random forest was applied to the validation data set to check the model's performance. The following are the results:

| Confusion Matrix and Statistics | | |
|----------------------------------|-----------|------|
| Prediction | Reference | |
| | 0 | 1 |
| 0 | 14510 | 1462 |
| 1 | 14 | 13 |
| Accuracy : 0.9077 | | |
| 95% CI : (0.9032, 0.9122) | | |
| No Information Rate : 0.9078 | | |
| P-Value [Acc > NIR] : 0.5178 | | |
| Kappa : 0.014 | | |
| McNemar's Test P-Value : <2e-16 | | |
| Sensitivity : 0.0088136 | | |
| Specificity : 0.9990361 | | |
| Pos Pred Value : 0.4814815 | | |
| Neg Pred Value : 0.9084648 | | |
| Prevalence : 0.0921933 | | |
| Detection Rate : 0.0008126 | | |
| Detection Prevalence : 0.0016876 | | |
| Balanced Accuracy : 0.5039248 | | |
| 'Positive' Class : 1 | | |

- The random forest model predicted true positives of 13 on the validation dataset.
- The random forest was then applied to the test data set to predict the number of customers defaulting.
- The predictions on the test dataset result showed that “33” customers belonged to class “1”, and 33 customers defaulted.

Filtering the number defaulting customers that was predicted by our Random Forest model.

```

{r}
predictions_pca_filtered<-predictions_pca %>% filter(predicted_default == 1)
predictions_pca_filtered

```

| Description: df [33 x 4] | | | | |
|--------------------------|-------------|-------------|-------------|----------------------------|
| | id
<int> | X0
<dbl> | X1
<dbl> | predicted_default
<dbl> |
| 1086 | 34664 | 0.580 | 0.420 | 1 |
| 1342 | 20578 | 0.382 | 0.618 | 1 |
| 3621 | 90934 | 0.564 | 0.436 | 1 |
| 5105 | 98853 | 0.554 | 0.446 | 1 |
| 6837 | 4702 | 0.568 | 0.432 | 1 |
| 7645 | 1180 | 0.586 | 0.414 | 1 |
| 7715 | 13944 | 0.598 | 0.402 | 1 |
| 8647 | 57274 | 0.412 | 0.588 | 1 |
| 8882 | 99336 | 0.388 | 0.612 | 1 |
| 9128 | 74118 | 0.530 | 0.470 | 1 |

1-10 of 33 rows

Previous 1 2 3 4 Next

- After predicting the customers defaulting, we created a Ridge Regression model using “MAE” as metrics to predict the loss generated from the defaulting customers.
- The “MAE” for the Ridge Regression model was at 0.0575126.

INSIGHTS:

- The Lasso and PCA effectively reduced the dataset and selected important variables.
- The random forest classifiers were evaluated based on their classification accuracy.
- The loss, as per the final Ridge regression model for the “33” defaulting customers, is:
 - Least loss for customer default – 2
 - Most loss for customer default – 33

CONCLUSION:

In conclusion, this project demonstrated the effectiveness of machine learning in predicting default and loss when there was a default. The random forest classifier provided the best performance for predicting defaulting customers. The insights gained from this analysis can be used to improve the lending decision-making process and reduce the risk of loan default.

Contributions of Each team member:

- Harish Kumar uddandi: Deciding the Threshold limits and visualizations.
- Abhinav Thupili: Understanding the relationship of the variables, deciding on the model to be used, and conclusions to be drawn per the model.
- Harish Varma Kunaparaju & Supriya Vengala: Worked on cleaning the data in the dataset and preparing the dataset.