Model #101: Credit Card Default Model

Model Development Guide

Tim Crnkovic

1. Introduction

The problem of credit risk modelling was examined. A set of consumer credit information was used to predict whether or not a customer would default. This data set was derived from the one used by Yeh and Lien (2009) in their research of techniques used to predict credit card default.

This was a binary classification problem with a target variable and multiple potential predictor variables. The raw predictor variable candidates were engineered in an attempt to create predictors which would provide insight into the classification task. These then were examined through exploratory data analysis to determine their relevance to the task.

Four different types of predictive models were tried in this analysis: a random forest, the boosted random forest model XGBoost, a logistic regression and a deep neural network constructed with TensorFlow and Keras for R. These four models were trained on a subset of the full data and their performances were assessed and compared.

The best results came from a simple logistic regression as opposed to powerful non-linear models, although none of the models performed particularly well on this task.

2. The Data

2.1. Data Description

The data for this analysis project consist of observations of features used to attempt to predict a credit consumer default. There are 30,000 total observations in the data set, divided into training, test and validation sets. The specific counts of the three sets are shown in section 2.4 below.

The dependent variable is a binary categorical variable, indicating either a default by the customer or not. There are 23 independent variables in the original data set. Three of the

independent variables are categorical; the remainder are continuous. There is also one index and five utility columns in the data, none of which are used to predict the response.

2.2. Data Dictionary

Shown below in Table 1 is the data dictionary for this data set. Note that this dictionary refers to the data as originally constituted. It does not include any engineered features.

Element	Description	Data type	Acceptable values
ID	Unique identifier.	Integer, sequential	> 0
LIMIT_BAL	The amount of credit given.	Integer, continuous	> 0
SEX	Gender.	Integer, categorical	1 = male, 2= female
EDUCATION	Level of education.	Integer, categorical	1 = graduate school, 2 = university, 3 = high school, 4 = others
MARRIAGE	Marital status.	Integer, categorical	1 = married, 2 = single, 3 = others
AGE	Age (years).	Integer, continuous	> 0
PAY_1 - PAY_6	History of past 6 payments.	Integer, categorical	-1 = paid in full, 1 = payment delay for 1 month, 2 = payment delay for 2 months, 3 = payment delay for 3 months, 4 = payment delay for 4 months, 5 = payment delay for 5 month, 6 = payment delay for 6 months, 7 = payment delay for 7 months, 8 = payment delay for 8 months, 9 = payment delay for 9 months or more
BILL_AMT_1 - BILL_AMT_6	Amount of billing statement for months 1 -6.	Integer, continuous	Any integer is theoretically acceptable.
PAY_AMT_1 - PAY_AMT_6	Amount of payment for previous billing period for months 1 -6.	Integer, continuous	>= 0
DEFAULT	Binary indicator of whether or not the consumer defaulted. Target variable.	Integer, categorical	1 = Yes, did default; 0 = No, did not default
u	Uniform random number used for utility purposes only, not a predictor.	Real, continous	[0,1)
train	Binary indicator of whether or not the observation is in the training set. Used for utility purposes only. Not a predictor.	Integer, categorical	1 = Yes, 0 = No
test	Binary indicator of whether or not the observation is in the test set. Used for utility purposes only. Not a predictor.	Integer, categorical	1 = Yes, 0 = No
validate	Binary indicator of whether or not the observation is in the validation set. Used for utility purposes only. Not a predictor.	Integer, categorical	1 = Yes, 0 = No
data.group	Indicator of which set the observation is in. Used for utility purposes only. Not a predictor.	Integer, categorical	1 = training set; 2 = test set; 3 = valiation set

Table 1. Data Dictionary.

2.3. Data Quality Check

In order to assess the quality of the data, a summary of the entire data set (excluding the index and utility columns) was produced and compared to the data dictionary. That summary can be seen in Table 2 below.

Table 2: Summary Statistics for Credit Card Default Data.

Statistic	N	Mean	St. Dev.	Min	Median	Max
LIMIT_BAL	30,000	167,484.30	129,747.70	10,000	140,000	1,000,000
SEX	30,000	1.60	0.49	1	2	2
EDUCATION	30,000	1.85	0.79	0	2	6
MARRIAGE	30,000	1.55	0.52	0	2	3
AGE	30,000	35.49	9.22	21	34	79
PAY_1	30,000	-0.02	1.12	-2	0	8
PAY_2	30,000	-0.13	1.20	-2	0	8
PAY_3	30,000	-0.17	1.20	-2	0	8
PAY_4	30,000	-0.22	1.17	-2	0	8
PAY_5	30,000	-0.27	1.13	-2	0	8
PAY_6	30,000	-0.29	1.15	-2	0	8
BILL_AMT1	30,000	51,223.33	73,635.86	-165,580	22,381.5	964,511
BILL_AMT2	30,000	49,179.08	71,173.77	-69,777	21,200	983,931
BILL_AMT3	30,000	47,013.15	69,349.39	-157,264	20,088.5	1,664,089
BILL_AMT4	30,000	43,262.95	64,332.86	-170,000	19,052	891,586
BILL_AMT5	30,000	40,311.40	60,797.16	-81,334	18,104.5	927,171
BILL_AMT6	30,000	38,871.76	59,554.11	-339,603	17,071	961,664
PAY_AMT1	30,000	5,663.58	16,563.28	0	2,100	873,552
PAY_AMT2	30,000	5,921.16	23,040.87	0	2,009	1,684,259
PAY_AMT3	30,000	5,225.68	17,606.96	0	1,800	896,040
PAY_AMT4	30,000	4,826.08	15,666.16	0	1,500	621,000
PAY_AMT5	30,000	4,799.39	15,278.31	0	1,500	426,529
PAY_AMT6	30,000	5,215.50	17,777.47	0	1,500	528,666
DEFAULT	30,000	0.22	0.42	0	0	1

It can be seen immediately that the EDUCATION feature has unexpected values. According to the data dictionary, the only valid values are 1-4. The frequency table for this feature (Table 3

below) shows that the values of 0, 5 and 6 are uncommon and thus will indeed be considered invalid. They will be mapped to a value for “Unknown”.

Table 3: Frequency Table of Education.

	Education	Freq
1	0	14
2	1	10,585
3	2	14,030
4	3	4,917
5	4	123
6	5	280
7	6	51

MARRIAGE also shows the unexpected value of 0. Its valid values are 1-3 and its frequency table below (Table 4) indicates that 0 is very uncommon. It will be considered invalid and mapped to a value of “Unknown”.

Table 4: Frequency Table of Marriage.

	Marriage	Freq
1	0	54
2	1	13,659
3	2	15,964
4	3	323

PAY_1 through PAY_6 all exhibit values that are undefined according to the data dictionary, which states that their valid values are -1 and 1-9. The frequency table of PAY_1 below (Table 5) shows thousands of cases of values of -2 and 0. The value of 0 is in fact the most frequent value. The other five PAY features show similar frequency distributions.

Table 5: Frequency Table of PAY_1.

	Pay_1	Freq
1	-2	2,759
2	-1	5,686
3	0	14,737
4	1	3,688
5	2	2,667
6	3	322
7	4	76
8	5	26
9	6	11
10	7	9
11	8	19

Figure 1 below is a stacked column chart of PAY_1 versus the value of DEFAULT, showing the influence of the value of DEFAULT on the distribution of the values of PAY_1.

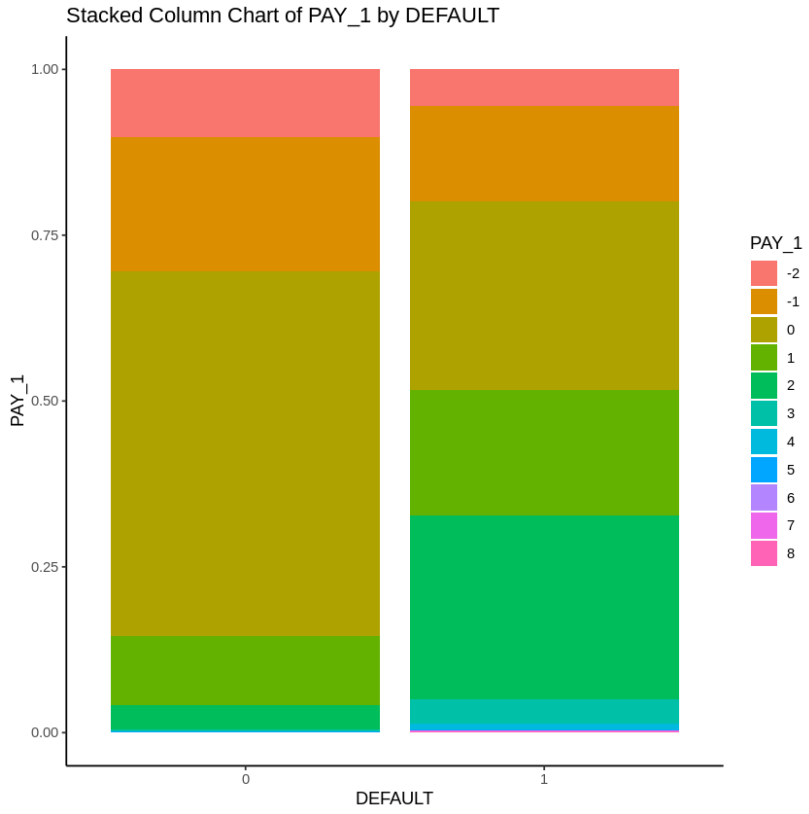

Figure 1. Stacked Column Chart of PAY_1 by DEFAULT.

The question then is, are the values of -2 and 0 truly invalid? Figure 1 shows that the proportion of 0 in PAY_1 is greatly influenced by the value of DEFAULT, indicating that it could have predictive value. It also seems unlikely that the most common value – 0 - would be invalid, so it will be considered valid for PAY_1-PAY_6, although its meaning is unclear.

The validity of the value of -2 is also in question. It is definitely not infrequent either and again, Figure 1 shows the influence that the value of DEFAULT has on the proportion of -2 in PAY_1, so it also may have predictive power. It will therefore not be discarded and will be considered a valid value, even though the meaning of the value of -2 is not at all apparent.

BILL_AMT1 through BILL_AMT6 all contain instances of negative values but this is entirely acceptable because a customer can have negative balances if they pay off more than the previous month's balance.

2.4. Counts of Observations

As mentioned, the data is divided into training, test and validation sets. The counts of observations in each of these three sets is shown in Table 6 below.

Table 6 : Number of Training, Test and Validation Observations.

	Train Obs	Test Obs	Val Obs
1	15,180	7,323	7,497

3. Feature Engineering

The feature AGE was discretized into three separate features. The intervals were determined using Weight of Evidence binning. The results of this binning are shown below in Table 7.

Table 7: WOE Binning Table.

Bin	Total_Count	Total_Distr.	0_Count	1_Count	0_Distr.	1_Distr.	1_Rate	WOE	IV
1 < = 25	3,871	12.9%	2,839	1,032	12.2%	15.6%	26.7%	-24.7	0.008
2 < = 40	17,855	59.5%	14,227	3,628	60.9%	54.7%	20.3%	10.8	0.007
3 < = Inf	8,274	27.6%	6,298	1,976	27.0%	29.8%	23.9%	-10.0	0.003
5 Total	30,000	100.0%	23,364	6,636	100.0%	100.0%	22.1%	NA	0.018

The WOE binning produced three bins: <=25, 26-29 and >40. Given that the minimum observed age in the data set is 21 and the maximum is 79, the new features are named **Age_21_25, Age_26_40** and **Age_41_79**. In Figure 2 below, a plot of the Weight of Evidence for the three bins is shown.

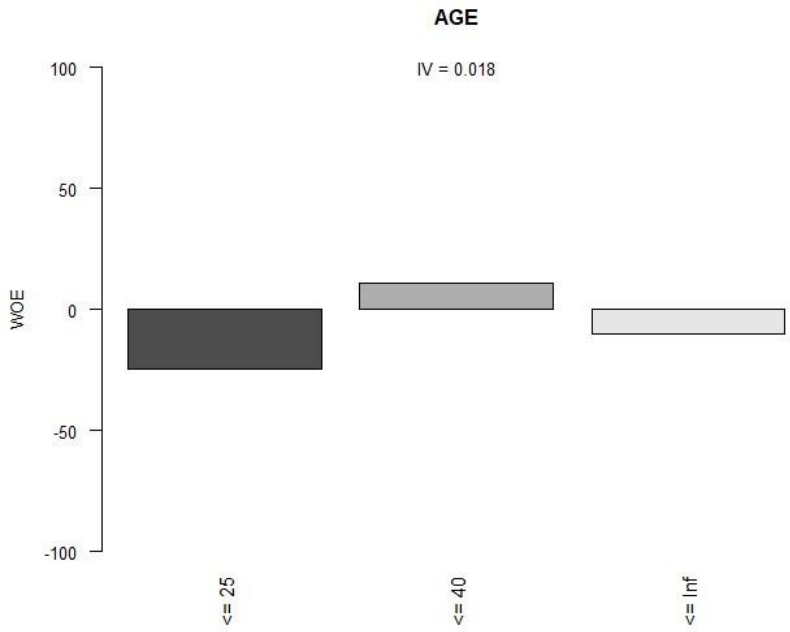

Figure 2. Weight of Evidence for Age Bins.

Numerous other features were engineered from the original features in the data set and these are described below.

- **Average Bill Amount** (Avg_Bill_Amt): this is the arithmetic mean of BILL_AMT1 through BILL_AMT6. Negative billing amounts are not changed to zero when computing this average

because that would cause loss of information. There may be some relationship between consistent negative balances and fewer defaults.

- **Total Negative Balances** (Tot_Neg_Balances): this is the total number of negative values in BILL_AMT1 through BILL_AMT6 and thus has a maximum value of 6. Again, this feature may indicate the relation between negative balances and fewer defaults.
- **Average Payment Amount** (Avg_Pmt_Amt): this is the arithmetic mean of PAY_AMT1 through PAY_AMT6 and may function as a proxy for income or the ability to pay.
- **Payment Ratio 1** (Pay_Ratio1): this is the ratio of PAY_AMT1 to BILL_AMT2. If BILL_AMT2 is zero, then this ratio is defined as 1, indicating full payment.
- **Payment Ratio 2** (Pay_Ratio2): this is the ratio of PAY_AMT2 to BILL_AMT3. If BILL_AMT3 is zero, then this ratio is defined as 1, indicating full payment.
- **Payment Ratio 3** (Pay_Ratio3): this is the ratio of PAY_AMT3 to BILL_AMT4. If BILL_AMT4 is zero, then this ratio is defined as 1, indicating full payment.
- **Payment Ratio 4** (Pay_Ratio4): this is the ratio of PAY_AMT4 to BILL_AMT5. If BILL_AMT5 is zero, then this ratio is defined as 1, indicating full payment.
- **Payment Ratio 5** (Pay_Ratio5): this is the ratio of PAY_AMT5 to BILL_AMT6. If BILL_AMT6 is zero, then this ratio is defined as 1, indicating full payment.
- **Average Payment Ratio** (Avg_Pmt_Ratio): this is the arithmetic mean of Pay_Ratio1 through Pay_Ratio5.
- **Payment Ratio Weighted Mean** (Pmt_Ratio_Weighted_Mean): this is an attempt to quantify payment ratio change over time. More recent payment ratios are given more weight than more distant ones. The precise definition is:

(Pay_Ratio1*0.6 + Pay_Ratio2*0.8 + Pay_Ratio3 + Pay_Ratio4*1.2 + Pay_Ratio5*1.4) / 5

- **Utilization 1** (Util1): this is the ratio of BILL_AMT1 to LIMIT_BAL.
- **Utilization 2** (Util2): this is the ratio of BILL_AMT2 to LIMIT_BAL.
- **Utilization 3** (Util3): this is the ratio of BILL_AMT3 to LIMIT_BAL.
- **Utilization 4** (Util4): this is the ratio of BILL_AMT4 to LIMIT_BAL.
- **Utilization 5** (Util5): this is the ratio of BILL_AMT5 to LIMIT_BAL.
- **Utilization 6** (Util6): this is the ratio of BILL_AMT6 to LIMIT_BAL.
- **Average Utilization** (Avg_Util): this is the arithmetic mean of Util1 through Util6.
- **Balance Growth Over 6 Months** (Bal_Growth_6mo): this is the percentage change in the balance over 6 months. It is defined as the difference between BILL_AMT6 and BILL_AMT1, divided by BILL_AMT1. If both BILL_AMT1 and BILL_AMT6 are <= 0, this value is defined to be 1. If BILL_AMT1 is <= 0 but BILL_AMT6 > 0, then this value is assigned the maximum value of Bal_Growth_6mo in the data set to avoid having a value of infinity.
- **Utilization Growth Over 6 Months** (Util_Growth_6mo): this is the difference between Util6 and Util1.
- **Utilization Increase from Minimum** (Util_Incr_From_Min): this is the difference between Util6 and the minimum of Util1 through Util6. This is another attempt to capture the change in utilization over time.
- **Utilization Weighted Mean** (Util_Weighted_Mean): this is yet another attempt to quantify utilization change over time. More recent utilizations are given more weight than more distant ones. The precise definition is:

  (Util1*0.5 + Util2*0.7 + Util3*0.9 + Util4*1.1 + Util5*1.3 + Util6*1.5) / 6

- **Maximum Bill Amount** (Max_Bill_Amt): this is the maximum of BILL_AMT1 through BILL_AMT6.

- **Maximum Pay Amount** (Max_Pmt_Amt): this is the maximum of PAY_AMT1 through PAY_AMT6.
- **Maximum Delinquency** (Max_DLQ): this is the maximum delinquency period as defined by the maximum of PAY_1 through PAY_6.
- **Total Delinquency** (Tot_DLQ): this is the total of delinquency period as defined by the total of PAY_1 through PAY_6, discounting negative values.

4. Exploratory Data Analysis

4.1. Traditional EDA

In order to perform exploratory data analysis, all engineered predictor variables were used but all variables from which they were engineered were excluded in order to avoid redundancy as much as possible. For example, the three binned age variables were used but the original AGE was not. Similarly, average and maximum billing amounts were used but the original BILL_AMT1 through BILL_AMT6 were excluded.

First, a matrix of Pearson Correlation was created among all of the included predictor variables and the target variable DEFAULT. This matrix is shown in Figure 3 below.

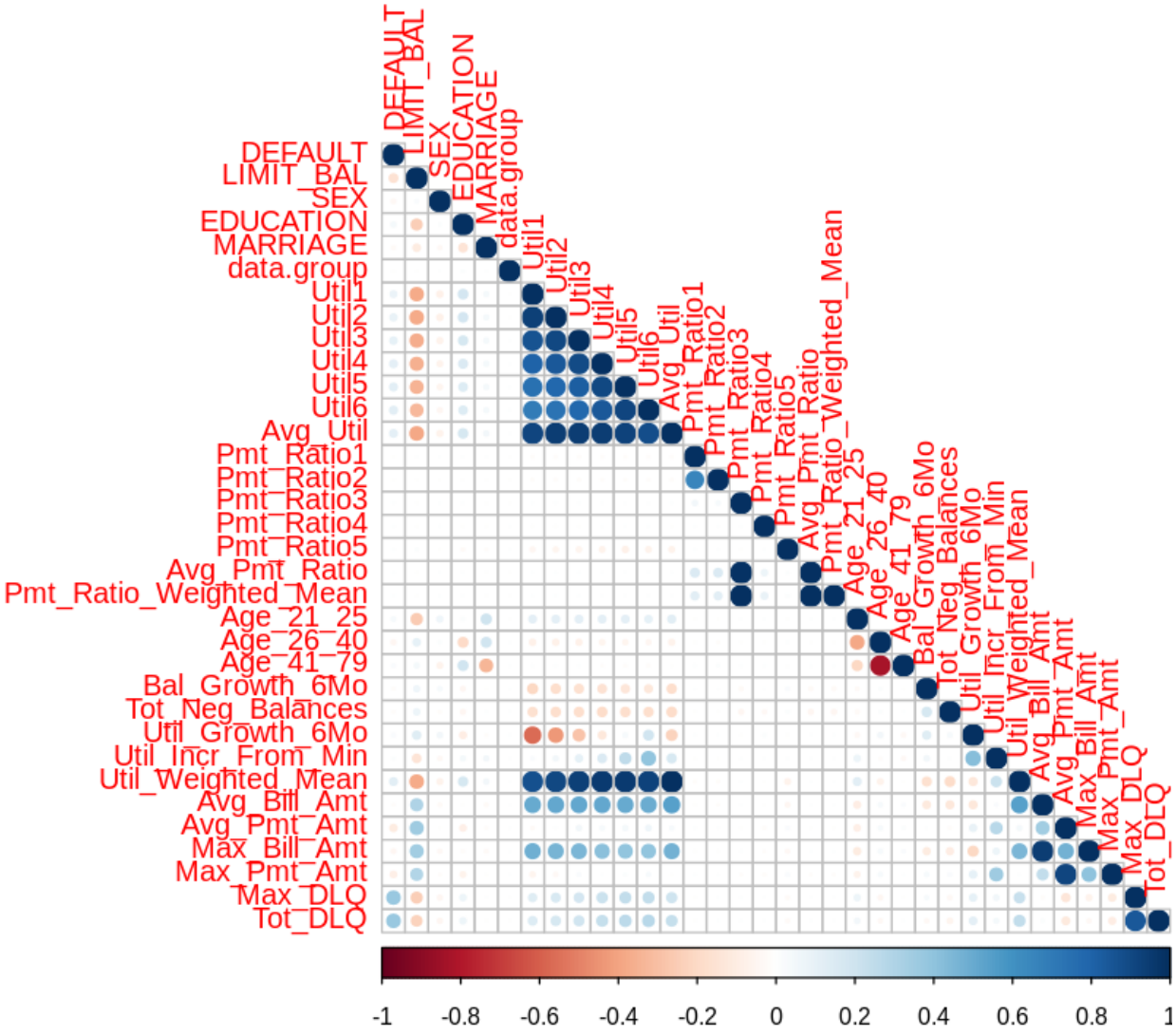

Figure 3. Matrix of Pearson Correlations.

Because DEFAULT is a categorical variable, the correlation matrix is less useful than it would have been if DEFAULT were continuous, but it is a place to start to look for interesting predictor variables. The predictor variables showing the greatest correlation with the target variable are Max_DLQ, the maximum delinquency period and Tot_DLQ, the total of all delinquency period values.

There are few other predictors that show any notable correlation with DEFAULT, but the ones that do are related to utilization and payment amount, as well as the credit limit. These variables are: LIMIT_BAL, Avg_Util, Util_Weighted_Mean, Avg_Pmt_Amt and Max_Pay_Amt. Each of the aforementioned predictor variables will be examined further. In particular, to examine how the values of these variables change with the value of DEFAULT, boxplots and density plots or histograms of each of the variables for each value of DEFAULT were created. These visualizations are all shown below.

For Max_DLQ, its boxplots and histograms varied by DEFAULT are shown in Figures 4 and 5 below.

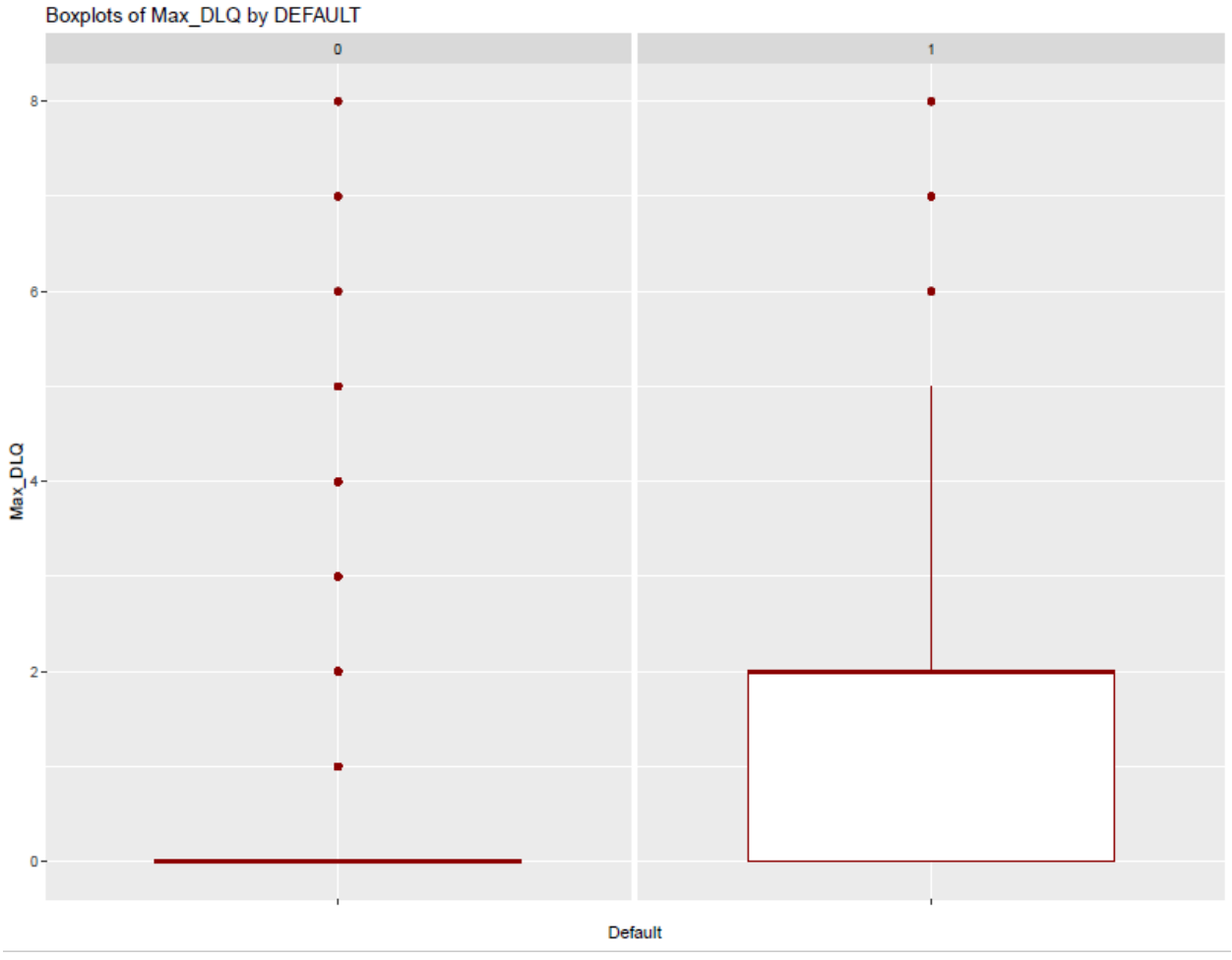

Figure 4. Boxplots of Max_DLQ by DEFAULT.

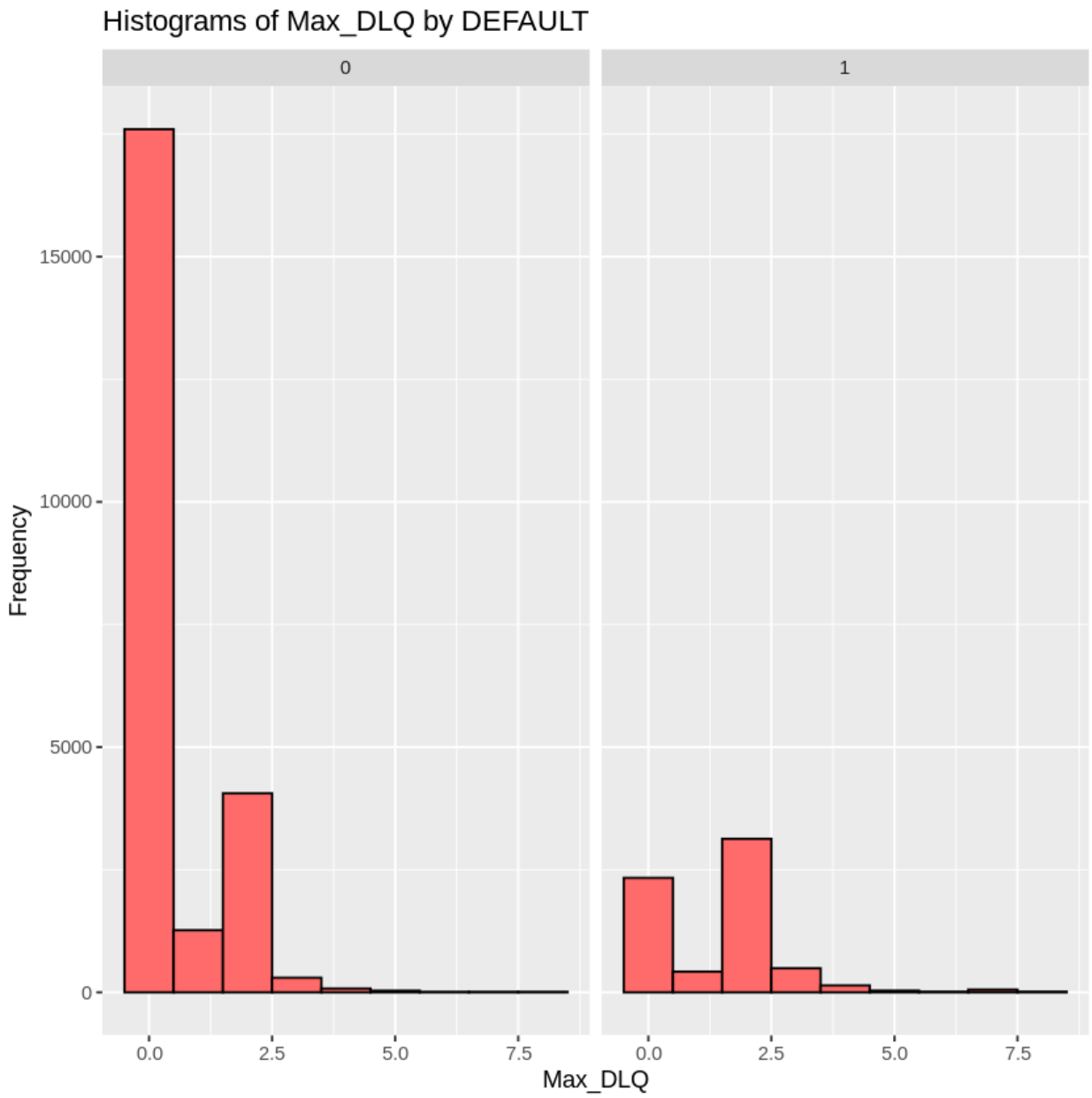

Figure 5. Histograms of Max_DLQ by DEFAULT.

The boxplots and histograms for Max_DLQ indicate that it may have considerable discriminatory power for DEFAULT as most of its values are lower for non-default than they are for default.

For Tot_DLQ, its boxplots and histograms varied by DEFAULT are shown in Figures 6 and 7 below.

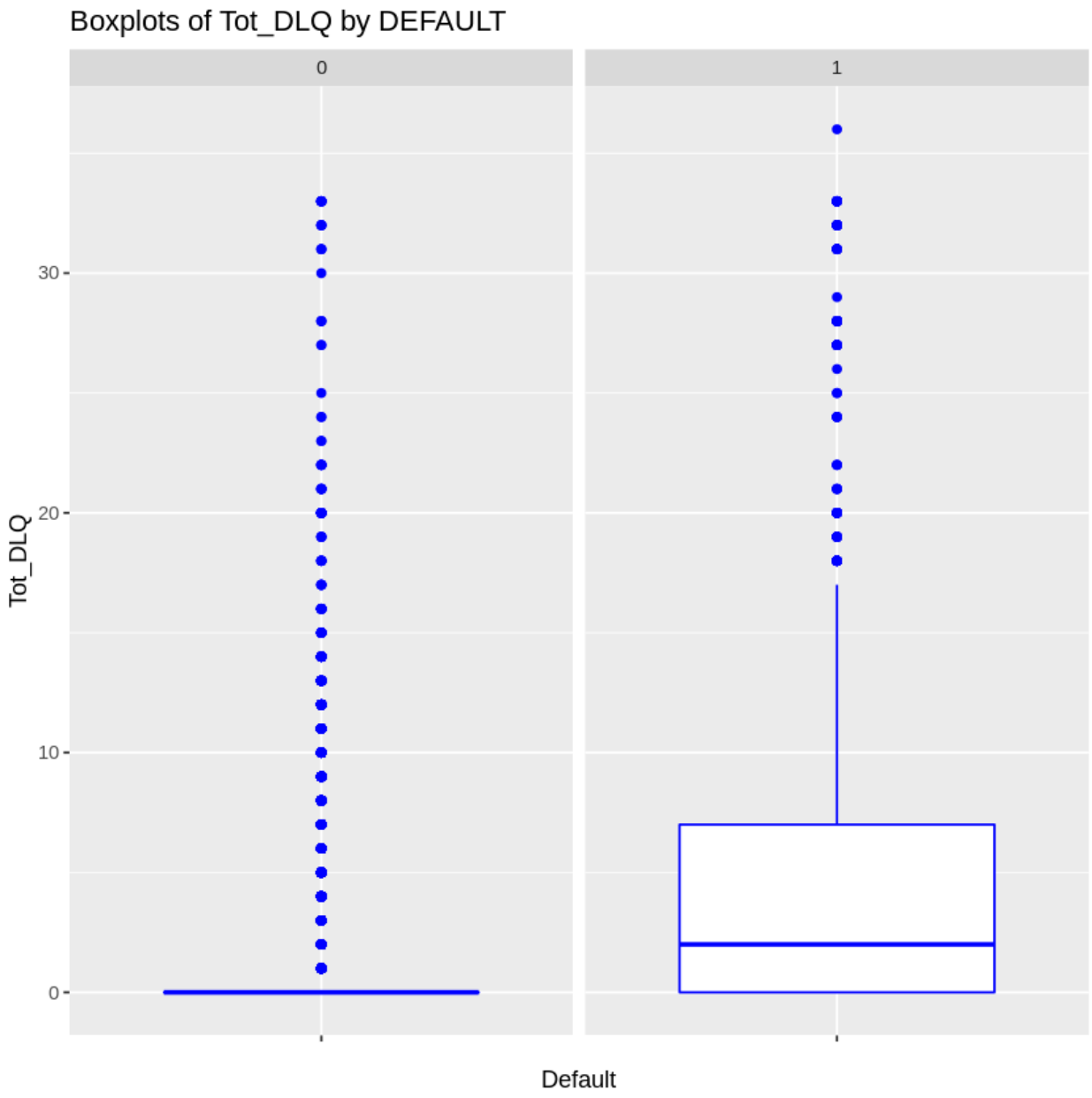

Figure 6. Boxplots of Tot_DLQ by DEFAULT.

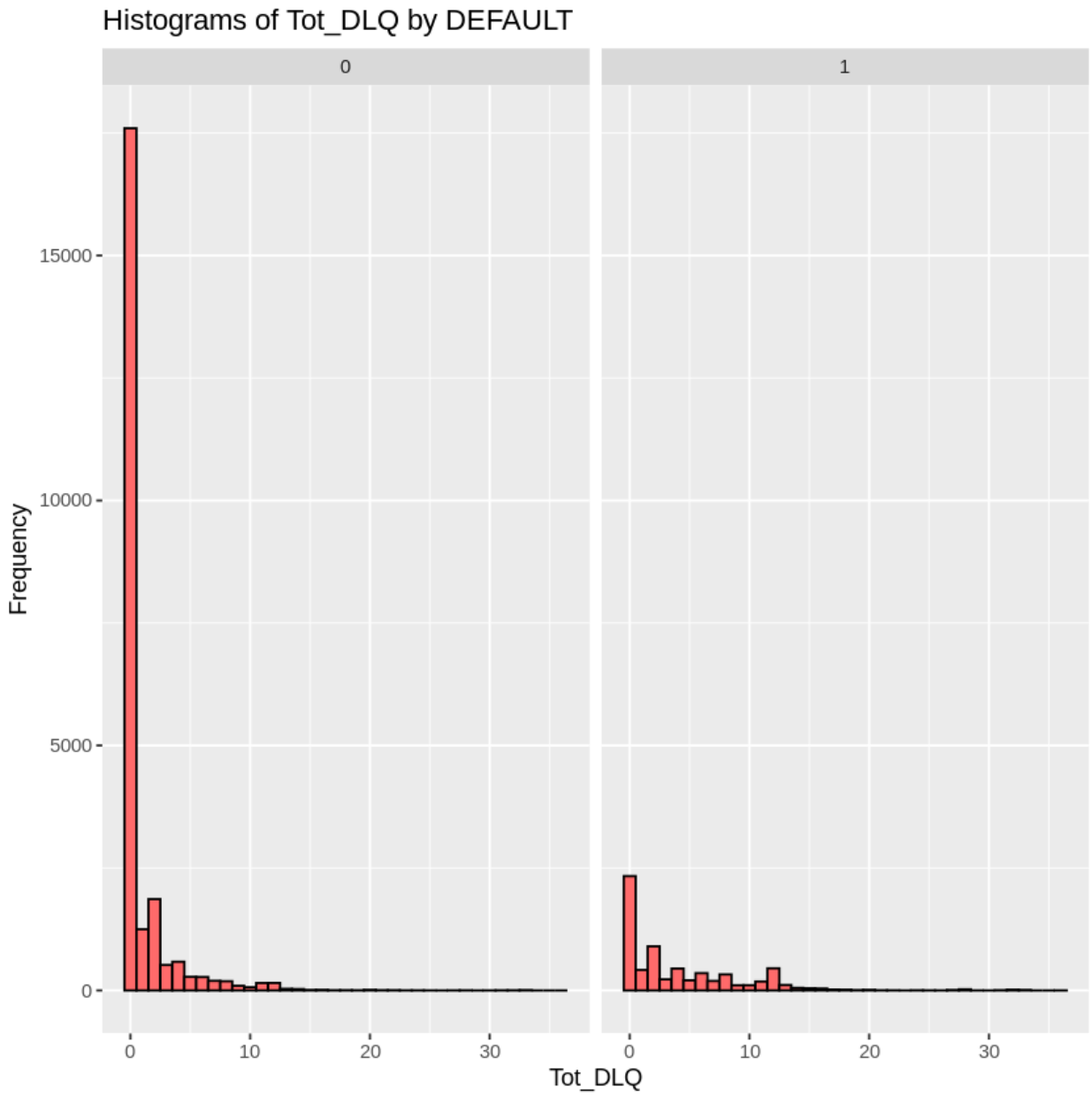

Figure 7. Histograms of Tot_DLQ by DEFAULT.

The boxplots and histograms for Tot_DLQ indicate that it may also have considerable discriminatory power for DEFAULT as most of its values are lower for non-default than they are for default.

For LIMIT_BAL, its boxplots and density plot varied by DEFAULT are shown in Figures 8 and 9 below. These plots were made using the log10 of LIMIT_BAL in an attempt to show the distributions more clearly.

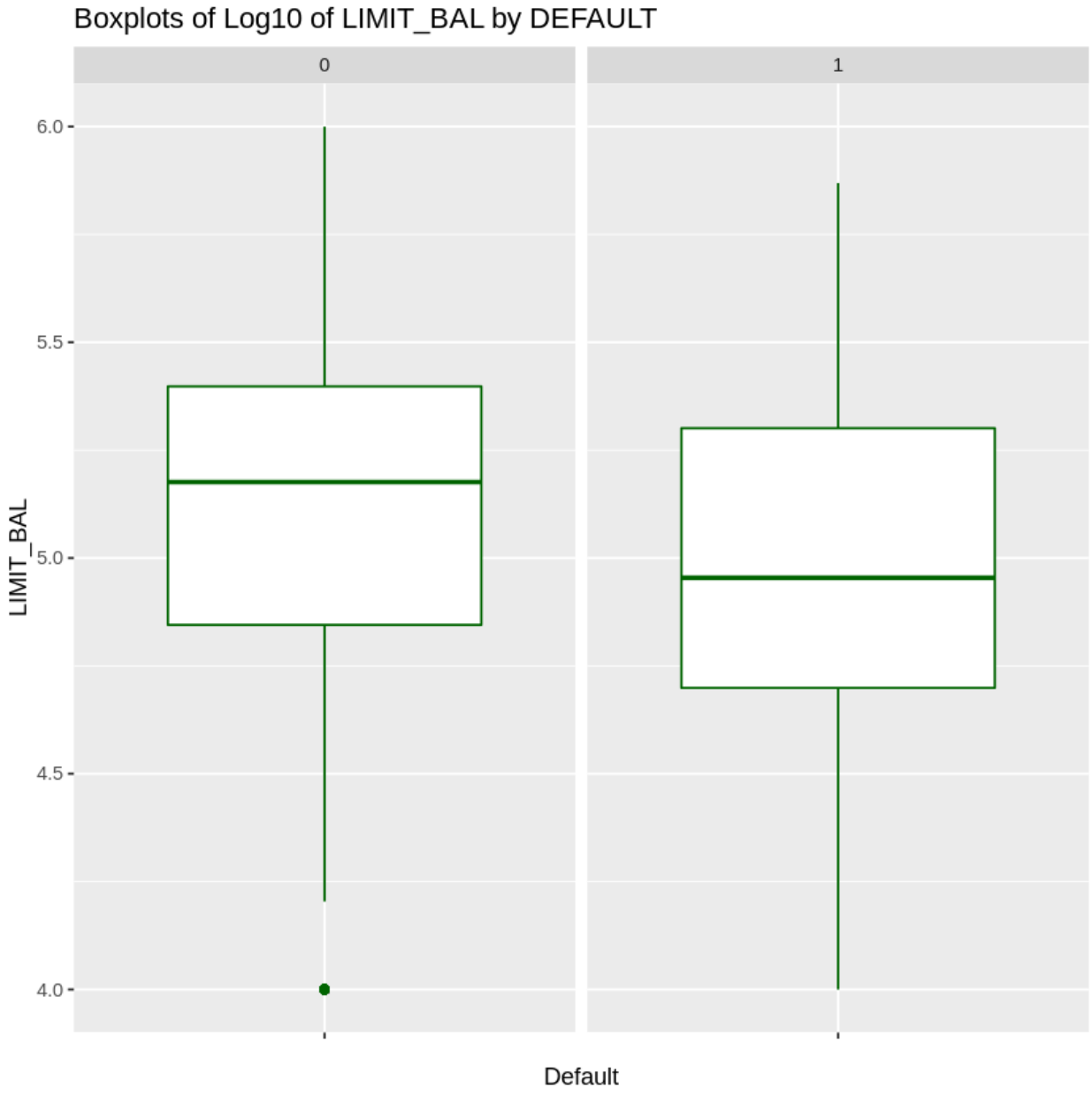

Figure 8. Boxplots of Log10 of LIMIT_BAL by DEFAULT.

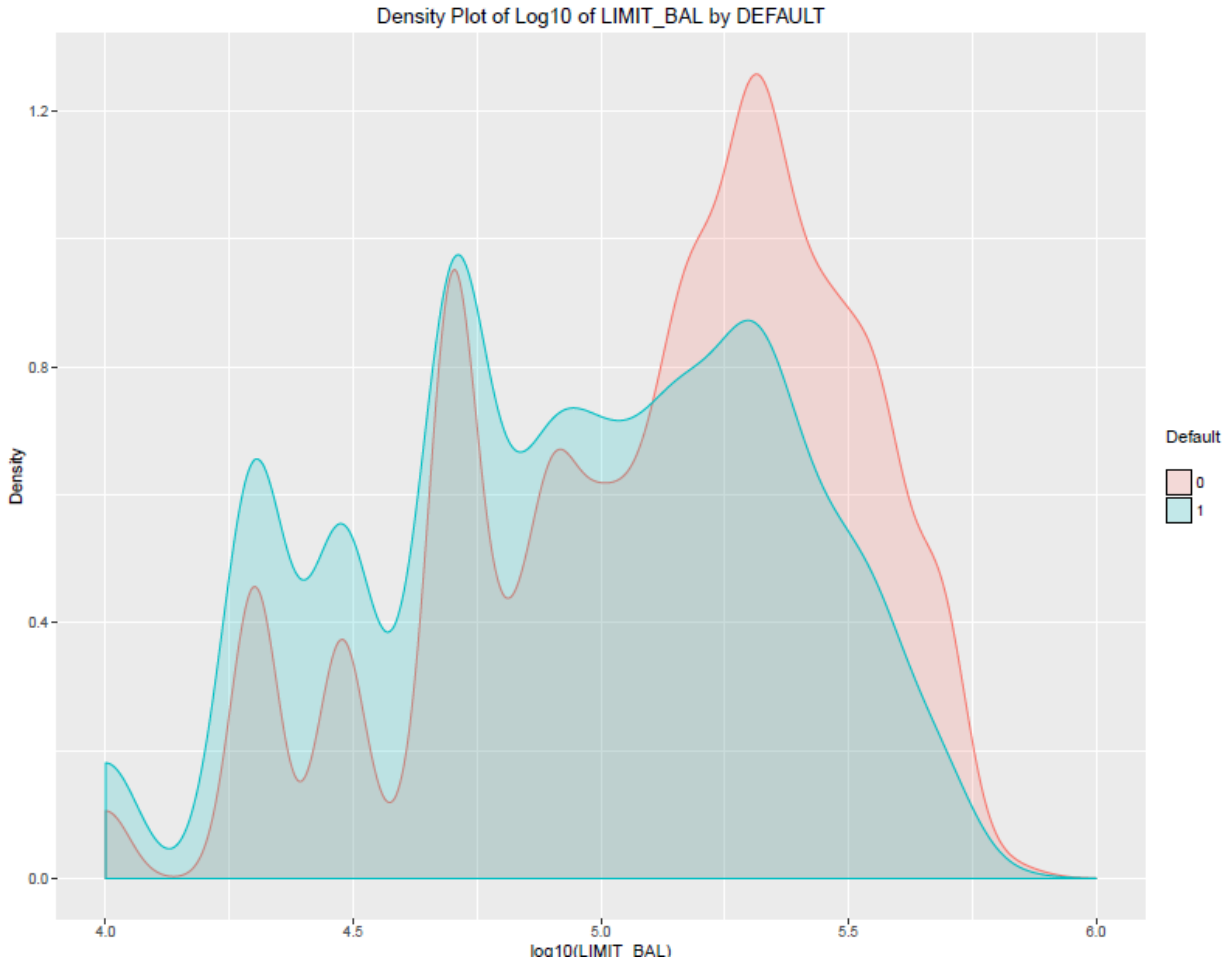

Figure 9. Density Plot of Log10 of LIMIT_BAL by DEFAULT.

The boxplots and density plot for log10 of LIMIT_BAL show that it may have some discriminatory power for DEFAULT as its median value and distribution is higher for non-default than for default.

For Avg_Util, its boxplots and density plot varied by DEFAULT are shown in Figures 10 and 11 below.

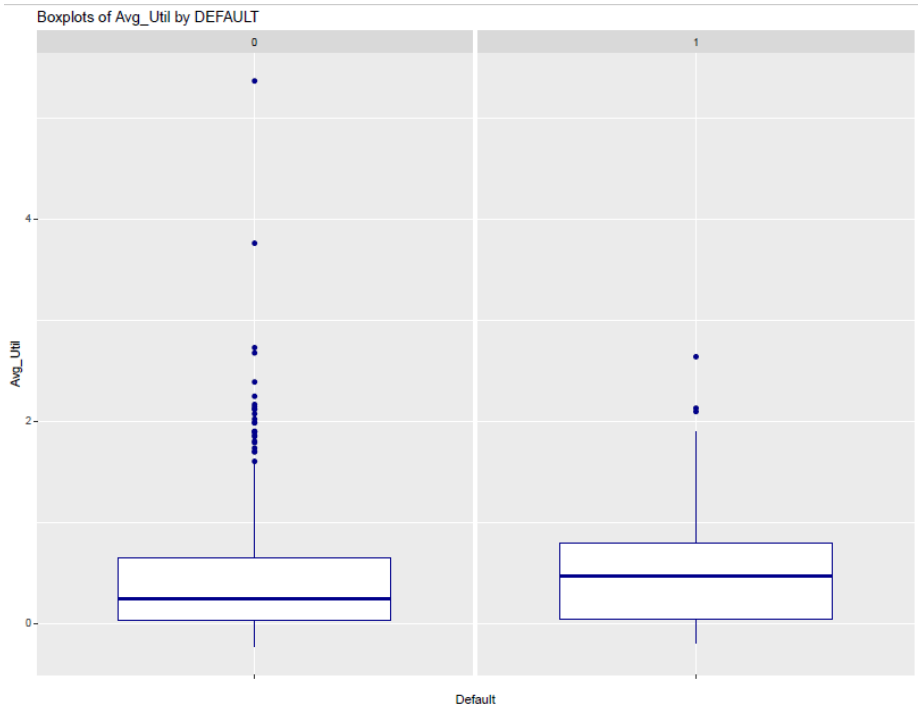

Figure 10. Boxplots of Avg_Util by DEFAULT.

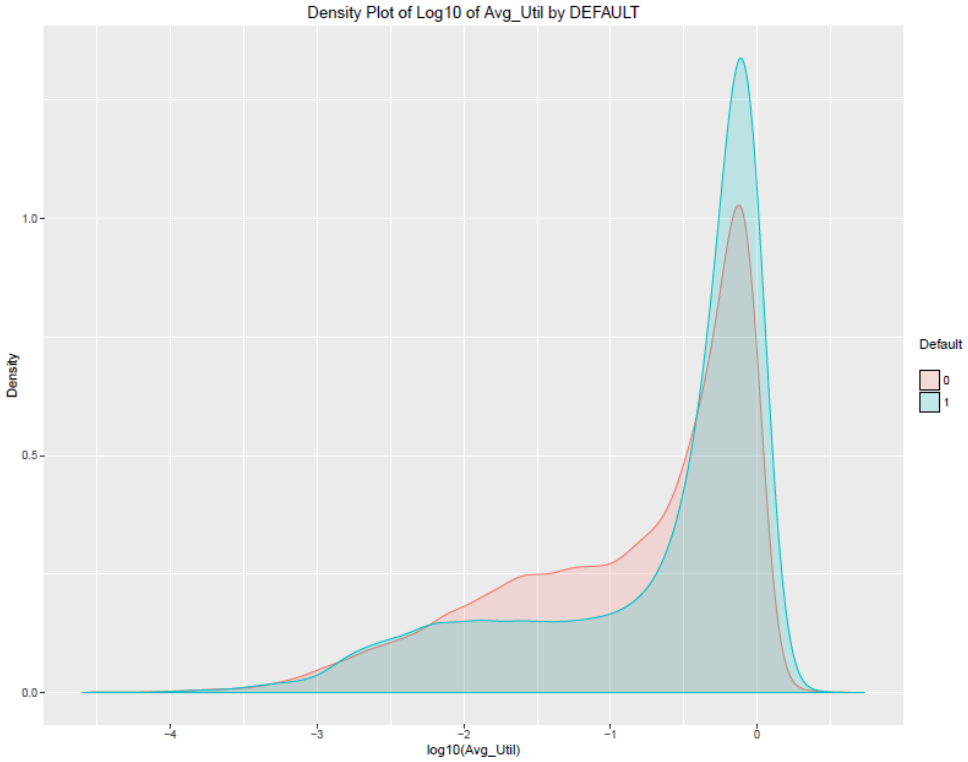

Figure 11. Density Plot of Avg_Util by DEFAULT.

The boxplots and density plot for Avg_Util indicate that it may have some discriminatory power for DEFAULT as its median value is lower for non-default than they for default.

For Util_Weighted_Mean, its boxplots and density plot varied by DEFAULT are shown in Figures 12 and 13 below.

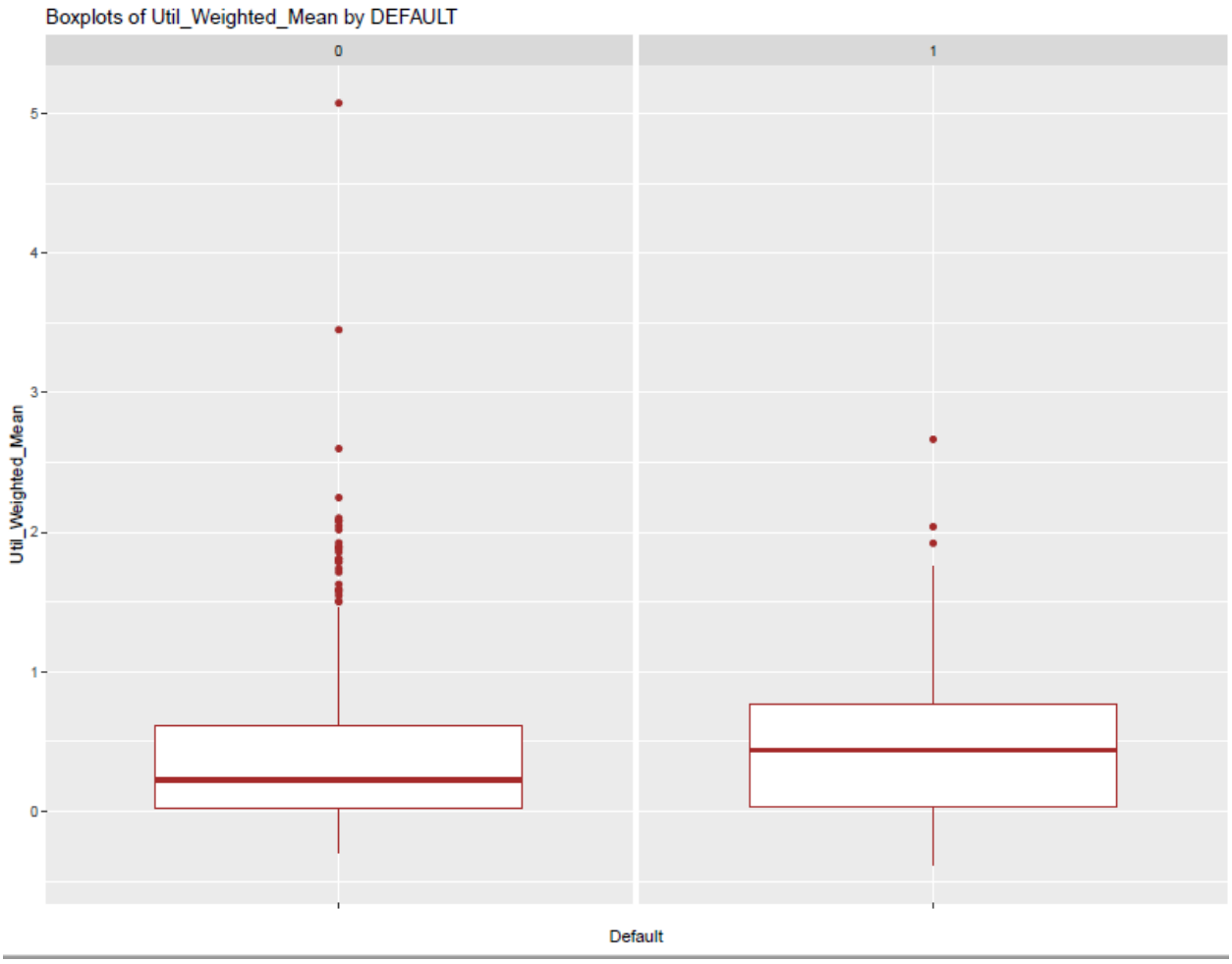

Figure 12. Boxplots of Util_Weighted_Mean by DEFAULT.

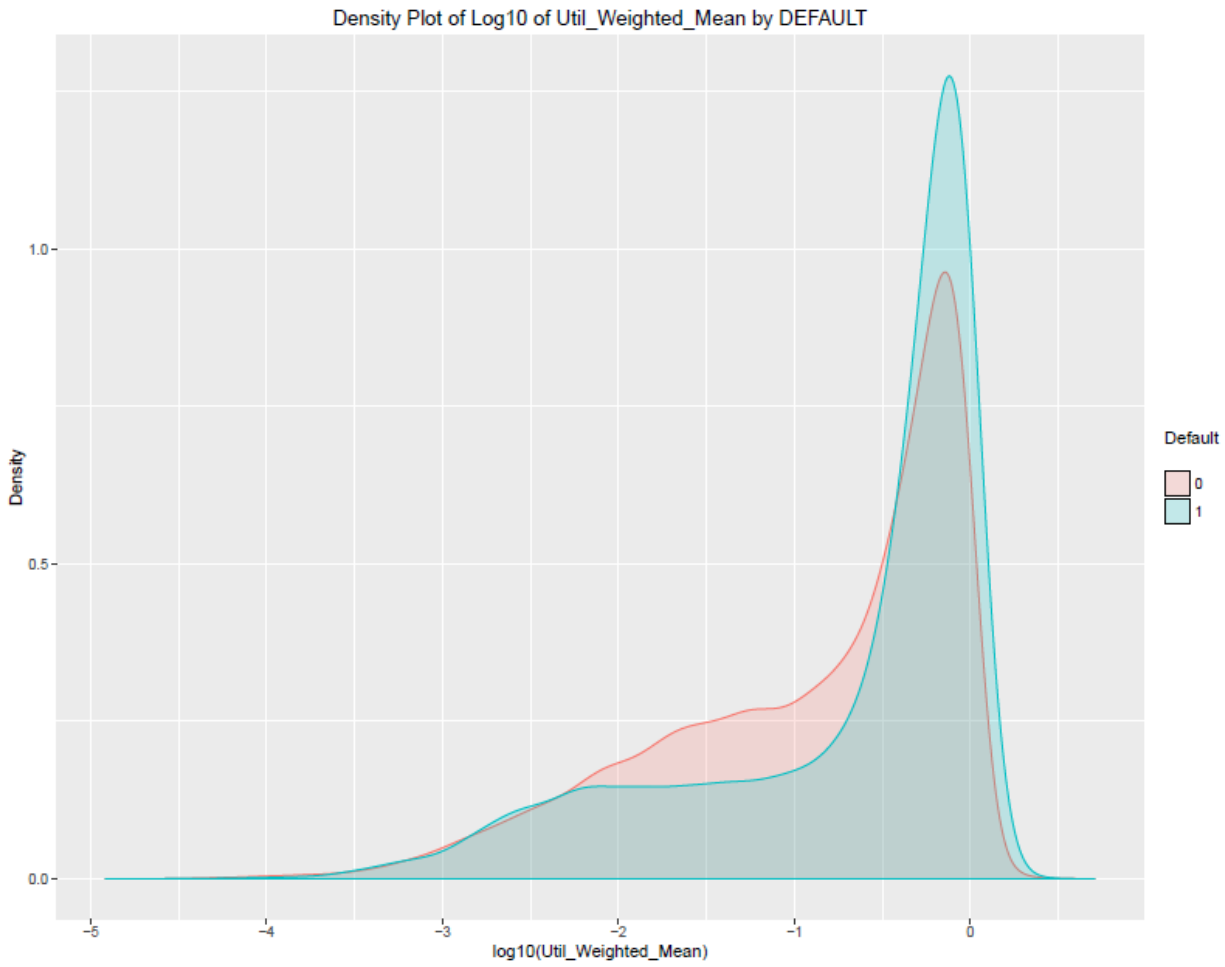

Figure 13. Density Plot of Util_Weighted_Mean by DEFAULT.

The boxplots and density plot for Util_Weighted_Mean indicate that it may have some discriminatory power for DEFAULT as its median value is lower for non-default than they for default.

For Avg_Pmt_Amt, its boxplots and density plot varied by DEFAULT are shown in Figures 14 and 15 below. These plots were made using the log10 of Avg_Pmt_Amt in an attempt to show the distributions more clearly.

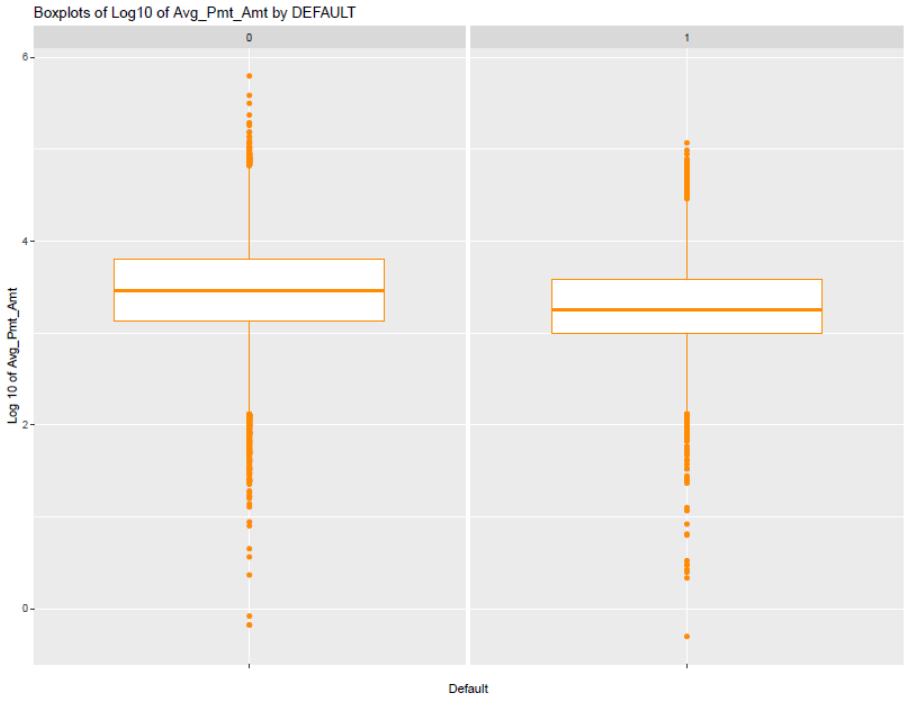

Figure 14. Boxplots of Log10 of Avg_Pmt_Amt by DEFAULT.

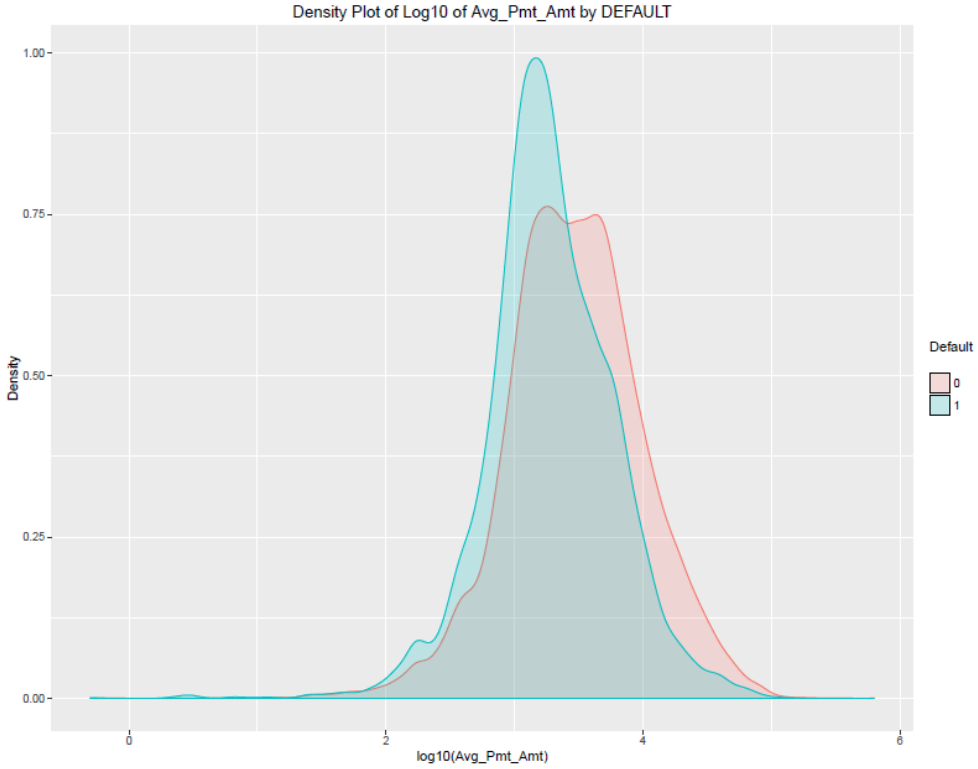

Figure 15. Density Plot of Log10 of Avg_Pmt_Amt by DEFAULT.

The boxplots and density plot for log10 of Avg_Pmt_Amt indicate that it may only have slight discriminatory power for DEFAULT as its distribution, median and range differ only slightly for non-default than for default.

For Max_Pmt_Amt, its boxplots and density plot varied by DEFAULT are shown in Figures 16 and 17 below. These plots were made using the log10 of Max_Pmt_Amt in an attempt to show the distributions more clearly.

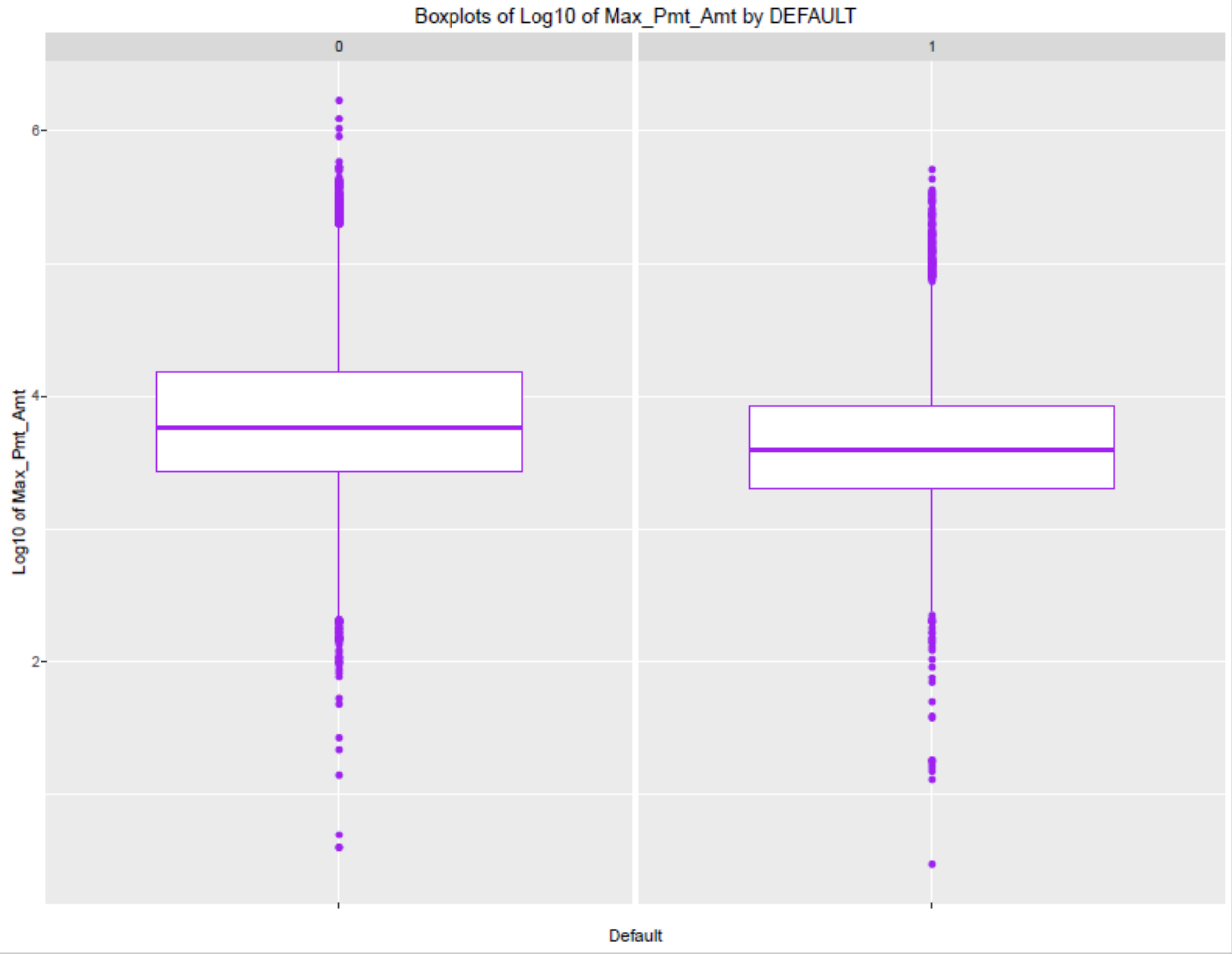

Figure 16. Boxplots of Log10 of Max_Pmt_Amt by DEFAULT.

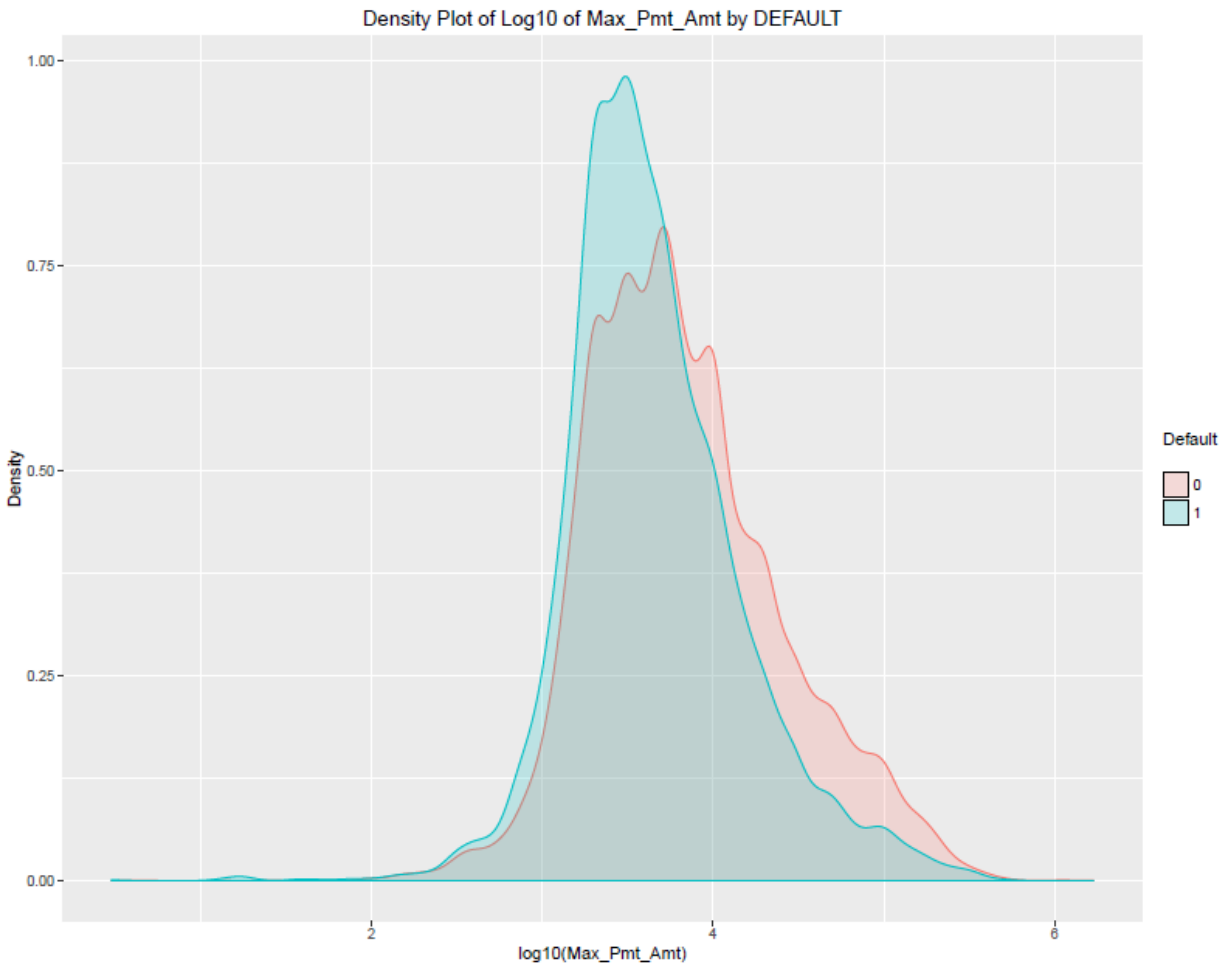

Figure 17. Density Plot of Log10 of Max_Pmt_Amt by DEFAULT.

The boxplots and density plot for log10 of Max_Pmt_Amt indicate that it may only have slight discriminatory power for DEFAULT as its distribution, median and range differ only slightly for non-default than for default.

4.1. Model Based EDA

A decision tree was fit for the credit card default data, with DEFAULT as the target variable. As with the traditional EDA, all engineered predictor variables were used in this analysis but all variables from which they were engineered were excluded.

Several different levels of complexity of the decision tree were tried, seeking to make the tree deep enough to discriminate between the two values of the target value but not so deep that the

lower layers provide little additional information. The results of the fitting of the decision tree are shown below in Figure 18.

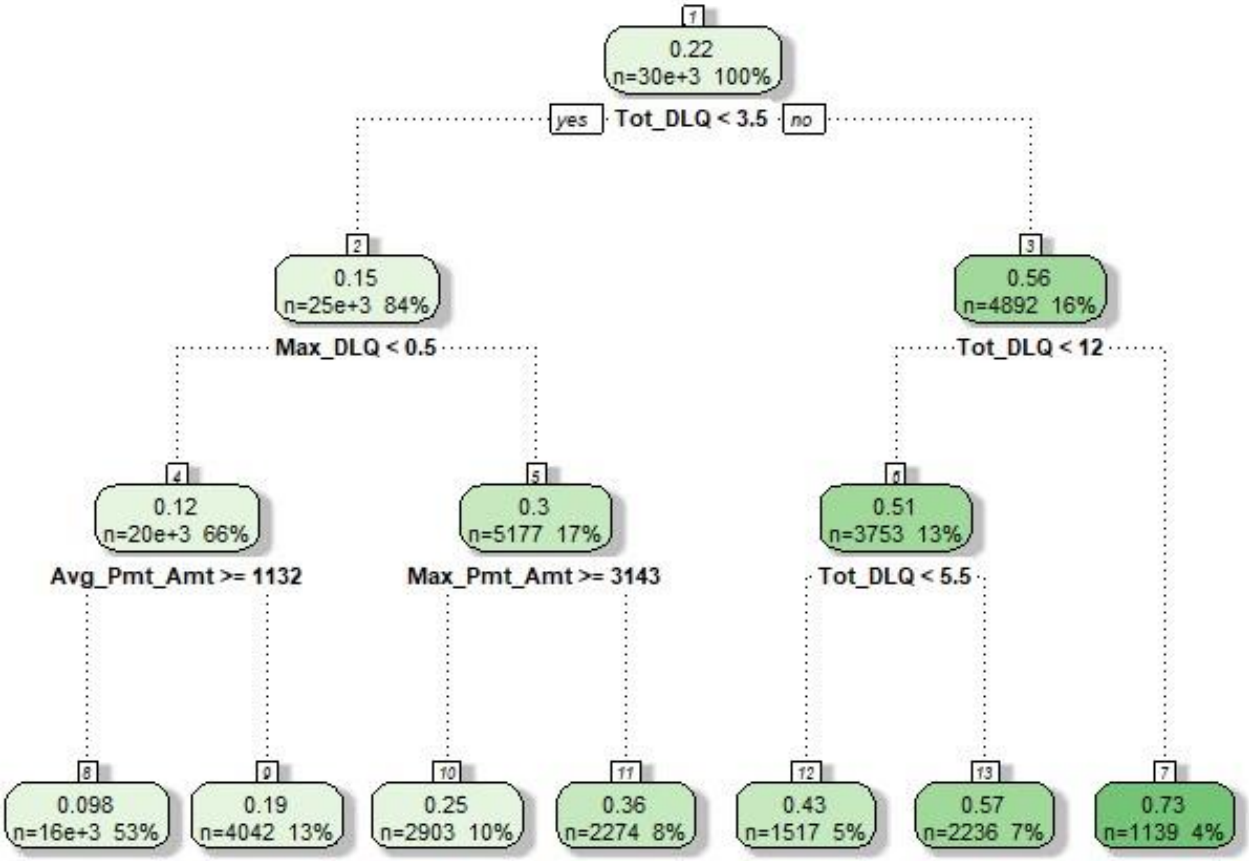

Figure 18. Decision Tree Model of Credit Card Default Data.

Figure 18 shows that two variables identified in the traditional EDA as possibly being useful for the classification of the DEFAULT variable were also used in the decision tree model: Max_DLQ and Tot_DLQ, both of which document the severity of delinquencies. These two variables were clearly the most important ones in this decision tree. It also shows that the next most important variables were also previously identified by traditional EDA as being potentially important to the classification: Avg_Pmt_Amt and Max_Pmt_Amt. This decision tree did not identify anything else that may be of interest and accordingly, no additional variables were examined with traditional EDA.

5. Predictive Modelling: Methods and Results

5.1. Random Forest

The credit card default data set was modelled with a random forest model, using the "randomForest" R package. This model was tuned by fixing the number of trees at 500 and varying the value for "mtry", which is the number of predictor variables considered at each split in a tree. Figure 19 below shows both the test error and the out of bag error as a function of the value of "mtry". The test error was lowest at a value of 7 and that value was used in the final model.

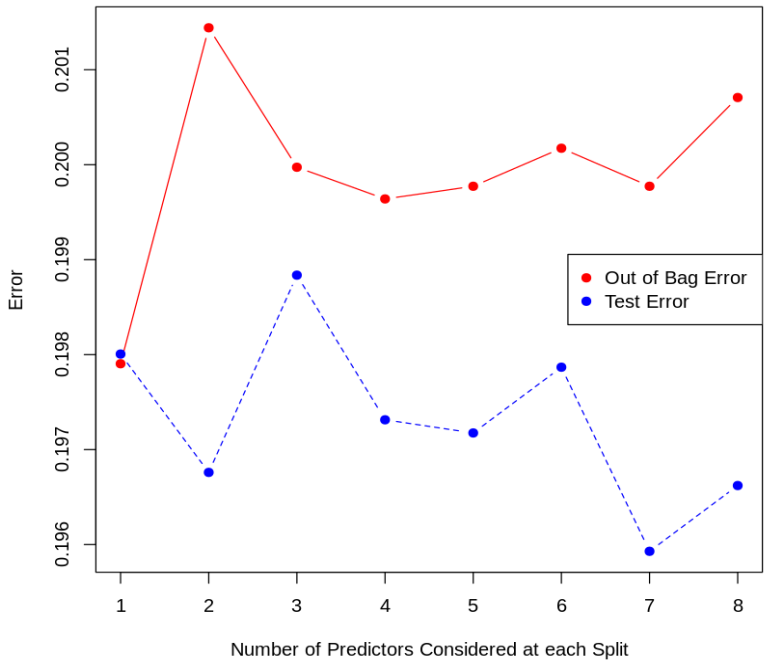

Figure 19. Test and OOB Error by "mtry" for Random Forest Model.

Table 7 below shows the feature importance of the predictor variables, as measured by mean decrease in Gini index. As expected from the results of EDA, the most important feature was Tot_DLQ. Many of the other important feature were also predicted by EDA.

Variable	MeanDecreaseGini
Tot_DLQ	632.2662
Avg_Pmt_Amt	375.88914
Max_Bill_Amt	356.28915
Max_Pmt_Amt	347.9669
Avg_Util	346.77896
Util_Growth_6Mo	344.62912
Util_Weighted_Mean	341.92392
Avg_Bill_Amt	330.61455
Max_DLQ	318.32772
Pmt_Ratio_Weighted_Mean	309.54713
Avg_Pmt_Ratio	304.07785
Bal_Growth_6Mo	292.63074
LIMIT_BAL	263.77118
Util_Incr_From_Min	238.70099
EDUCATION	98.4821
MARRIAGE	63.39035
SEX	49.32008
Age_41_79	46.72595
Age_26_40	46.41252
Age_21_25	30.12687
Tot_Neg_Balances	19.1687

Table 7. Feature Importance in Random Forest Model.

The full results of the training and testing with the random forest model are summarized in Table 8 below. The model produced a true positive rate of 0.34, a false positive rate of 0.07, an overall accuracy of 0.80, an F1 score of 0.48 and an AUC of 0.63 against the test set.

Model #1: Random Forest Model - Training Set												
Actual Class	Predicted Class 0	Predicted Class 1	Totals		Actual Class	Predicted Class 0	Predicted Class 1	TP	0.98	TP+TN	1.98	AUC	0.99
0	11,562	13	11,575		0	1.00	0.00	TN	1.00	Precision	1.00	Sensitivity	0.98
1	66	3,341	3,407		1	0.02	0.98	Type I Error	0.00	Recall	0.98	Specificity	1.00
								Type II Error	0.02	F1	0.99		
Model #1: Random Forest Model - Test Set												
Actual Class	Predicted Class 0	Predicted Class 1	Totals		Actual Class	Predicted Class 0	Predicted Class 1	TP	0.34	TP+TN	1.26	AUC	0.63
0	5,262	412	5,674		0	0.93	0.07	TN	0.93	Precision	0.56	Sensitivity	0.34
1	1,026	522	1,548		1	0.66	0.34	Type I Error	0.07	Recall	0.34	Specificity	0.93
								Type II Error	0.66	F1	0.48		

Table 8. Confusion Matrix and Classification Metrics for Random Forest Model.

5.2. Boosted Random Forest with XGBoost

The data set was then modelled with a boosted random forest model with XGBoost, using the "xgboost" R package. This model was tuned by varying the values of the hyperparameters "eta", "nrounds" and "max_depth". Best performance was achieved with values for those of 0.25, 17 and 3, respectively.

The importance of features used in this model are shown in Table 9 below. Once again, Tot_DLQ is the most important, followed by Max_DLQ and several others that the random forest model and EDA also identified as important.

Feature	Gain
Tot_DLQ	0.420
Max_DLQ	0.259
Max_Bill_Amt	0.045
Util_Growth_6Mo	0.042
Avg_Pmt_Amt	0.041
Avg_Util	0.034
Max_Pmt_Amt	0.034
LIMIT_BAL	0.027
Avg_Bill_Amt	0.022
Bal_Growth_6Mo	0.018
Util_Incr_From_Min	0.018
Util_Weighted_Mean	0.013
Avg_Pmt_Ratio	0.009
MARRIAGE	0.006
Pmt_Ratio_Weighted_Mean	0.006
EDUCATION	0.003
Tot_Neg_Balances	0.003
Age_41_79	0.000

Table 9. Feature Importance in the Boosted Random Forest with XGBoost.

The full results of the training and testing with the boosted random forest model with XGBoost are summarized in Table 10 below. The model produced a true positive rate of 0.32, a false positive rate of 0.06 and an overall accuracy of 0.81, an F1 score of 0.47 and an AUC of 0.64 against the test set.

Model #2: Boosted Random Forest with XGBoost Model - Training Set												
Actual Class	Predicted Class		Totals		Actual Class	Predicted Class		TP	0.33	TP+TN	1.28	AUC	0.63
	0	1				0	1	TN	0.94	Precision	0.64	Sensitivity	0.33
0	10,932	643	11,575		0	0.94	0.06	Type I Error	0.06	Recall	0.33	Specificity	0.94
1	2,277	1,130	3,407		1	0.67	0.33	Type II Error	0.67	F1	0.48		
Model #2: Boosted Random Forest with XGBoost Model - Test Set												
Actual Class	Predicted Class		Totals		Actual Class	Predicted Class		TP	0.32	TP+TN	1.26	AUC	0.64
	0	1				0	1	TN	0.94	Precision	0.60	Sensitivity	0.32
0	5,337	337	5,674		0	0.94	0.06	Type I Error	0.06	Recall	0.32	Specificity	0.94
1	1,046	502	1,548		1	0.68	0.32	Type II Error	0.68	F1	0.47		

Table 10. Confusion Matrix and Classification Metrics for XGBoost Model.

5.3. Logistic Regression with L1 (Lasso) Regularization Used for Variable Selection

The data set was modelled with a logistic regression model, using the "glm" R package. In order to perform feature selection for this model, L1 (Lasso) regularization was used. The results of the regularization are shown in Table 11 below. The variables whose weights were zeroed-out by regularization were excluded from the subsequent logistic regression modelling.

Variable	L1 Regularized Weight
(Intercept)	1.53E-01
LIMIT_BAL	-1.71E-07
SEX	-1.58E-02
EDUCATION	0
MARRIAGE	-2.22E-02
Avg_Util	1.68E-03
Avg_Pmt_Ratio	0
Pmt_Ratio_Weighted_Mean	0
Age_21_25	0
Age_26_40	-9.82E-03
Age_41_79	0
Bal_Growth_6Mo	-6.79E-07
Tot_Neg_Balances	0
Util_Growth_6Mo	-1.29E-02
Util_Incr_From_Min	0
Util_Weighted_Mean	0
Avg_Bill_Amt	7.06E-08
Avg_Pmt_Amt	-1.41E-06
Max_Bill_Amt	0
Max_Pmt_Amt	0
Max_DLQ	5.99E-02
Tot_DLQ	2.63E-02

Table 11. Features Selected for Logistic Regression by L1 Regularization.

Table 12 below shows coefficients, z and P values and significance for the included features.

Feature	Estimate	z value	Pr(>|z|)	Significance
(Intercept)	-1.39	-17.663	< 2.00E-16	***
LIMIT_BAL	0.00	-5.209	1.90E-07	***
Max_DLQ1	13.30	0.058	0.953784	
Max_DLQ2	15.03	0.065	0.94781	
Max_DLQ3	15.23	0.066	0.947105	
Max_DLQ4	14.82	0.065	0.948511	
Max_DLQ5	13.51	0.059	0.953064	
Max_DLQ6	14.38	0.063	0.950036	
Max_DLQ7	27.54	0.069	0.9448	
Max_DLQ8	14.01	0.043	0.96559	
Tot_DLQ1	-12.26	-0.053	0.957421	
Tot_DLQ2	-13.83	-0.06	0.951965	
Tot_DLQ3	-14.23	-0.062	0.950585	
Tot_DLQ4	-13.44	-0.059	0.953297	
Tot_DLQ5	-13.48	-0.059	0.953178	
Tot_DLQ6	-12.84	-0.056	0.955383	
Tot_DLQ7	-13.27	-0.058	0.953913	
Tot_DLQ8	-12.63	-0.055	0.95614	
Tot_DLQ9	-13.06	-0.057	0.954619	
Tot_DLQ10	-12.80	-0.056	0.955545	
Tot_DLQ11	-12.95	-0.056	0.955021	
Tot_DLQ12	-11.95	-0.052	0.958488	
Tot_DLQ13	-12.27	-0.053	0.957363	
Tot_DLQ14	-12.54	-0.055	0.956441	
Tot_DLQ15	-11.77	-0.051	0.959101	
Tot_DLQ16	-12.23	-0.053	0.957511	
Tot_DLQ17	-10.90	-0.047	0.962131	
Tot_DLQ18	-10.91	-0.048	0.962105	
Tot_DLQ19	-11.98	-0.052	0.958385	
Tot_DLQ20	-11.73	-0.051	0.959261	
Tot_DLQ21	-13.57	-0.059	0.952861	
Tot_DLQ22	-13.38	-0.058	0.953509	
Tot_DLQ24	-11.61	-0.051	0.959647	
Tot_DLQ27	-13.31	-0.032	0.974676	
Tot_DLQ28	-23.22	-0.058	0.953443	
Tot_DLQ31	-13.27	-0.041	0.967407	
Tot_DLQ32	-24.36	-0.061	0.951166	
Tot_DLQ33	-11.63	-0.036	0.971429	
MARRIAGE2	-0.18	-3.968	7.25E-05	***
MARRIAGE3	-0.11	-0.566	0.571414	
SEX2	-0.16	-3.665	0.000248	***
Util_Growth_6Mo	-0.25	-3.318	0.000906	***
Age_26_40	-0.09	-1.902	0.057234	.
Avg_Util	-0.18	-1.8	0.07185	.
Bal_Growth_6Mo	0.00	-2.103	0.035434	*
Avg_Pmt_Amt	0.00	-5.733	9.86E-09	***
Avg_Bill_Amt	0.00	3.373	0.000743	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				

Table 12. Coefficients, z and P Values for Logistic Regression Features.

Table 12 itemizes the categorical features by level. By convention, the first level of each categorical variable is defined to have an estimate of zero and that level is not displayed. The table shows that Avg_Bill_Amt is the only feature significantly correlated with an increased likelihood of default, meaning that an increase in its value corresponds to increased probability of default with very high confidence. Note that this is true even though its estimate shows as 0. This is a result of the great range in values for this feature. Even a small increase in Avg_Bill_Amt will likely lead to a greater chance of default.

Table 12 also shows several features significantly related to a decreased probability of default: Avg_Pmt_Amt, Util_Growth_6Mo, LIMIT_BAL, MARRIAGE2 (Single) and SEX2 (Female). Increases in these should lead to a smaller chance of default. This is true for LIMIT_BAL and Avg_Pmt_Amt even with their estimates of 0. Again, that is a result of their large ranges of values. Even small changes in their value could lead to significant effects on chance of default.

The full results of the training and testing with the logistic regression model are summarized in Table 13 below. The model produced a true positive rate of 0.36, a false positive rate of 0.07 and an overall accuracy of 0.81, an F1 score of 0.51 and an AUC of 0.65 against the test set.

Model #3: Logistic Regression Model - Training Set												
Actual Class	Predicted Class 0	Predicted Class 1	Totals		Actual Class	Predicted Class 0	Predicted Class 1	TP	0.37	TP+TN	1.30	AUC	0.63
0	10,762	813	11,575		0	0.93	0.07	TN	0.93	Precision	0.61	Sensitivity	0.37
1	2,144	1,263	3,407		1	0.63	0.37	Type I Error	0.07	Recall	0.37	Specificity	0.93
								Type II Error	0.63	F1	0.51		
Model #3: Logistic Regression Model - Test Set												
Actual Class	Predicted Class 0	Predicted Class 1	Totals		Actual Class	Predicted Class 0	Predicted Class 1	TP	0.36	TP+TN	1.29	AUC	0.65
0	5,253	421	5,674		0	0.93	0.07	TN	0.93	Precision	0.57	Sensitivity	0.36
1	983	565	1,548		1	0.64	0.36	Type I Error	0.07	Recall	0.36	Specificity	0.93
								Type II Error	0.64	F1	0.51		

Table 13. Confusion Matrix and Classification Metrics for Logistic Regression Model.

5.4. Deep Neural Network

The data set was then modelled with a deep neural network (DNN), using the "keras" R package which runs TensorFlow. Several different architectures were experimented with. The final model used for training and testing was composed of 5 densely connected layers. The full results of the training and testing with this and the other models are summarized in the Table 14 in below.

Figure 20 below shows the structure of the DNN. It starts with 24 nodes in the first dense layer and ends with 1 in the final. The output classification is produced by a sigmoid activation function.

Layer (type)	Output Shape	Param #
dense_46 (Dense)	(None, 24)	528
dense_47 (Dense)	(None, 16)	400
dense_48 (Dense)	(None, 8)	136
dense_49 (Dense)	(None, 4)	36
dense_50 (Dense)	(None, 1)	5

Total params: 1,105
Trainable params: 1,105
Non-trainable params: 0

Figure 20. Structure of Deep Neural Network Model.

The metrics of accuracy, binary cross-entropy loss and AUC were tracked against both the training and the test set for each epoch of the model training. Testing revealed that the best combination of AUC and accuracy against the test set occurred after 5 epochs, so that was chosen as the best model. Figure 21 below shows a plot of the various metrics per epoch.

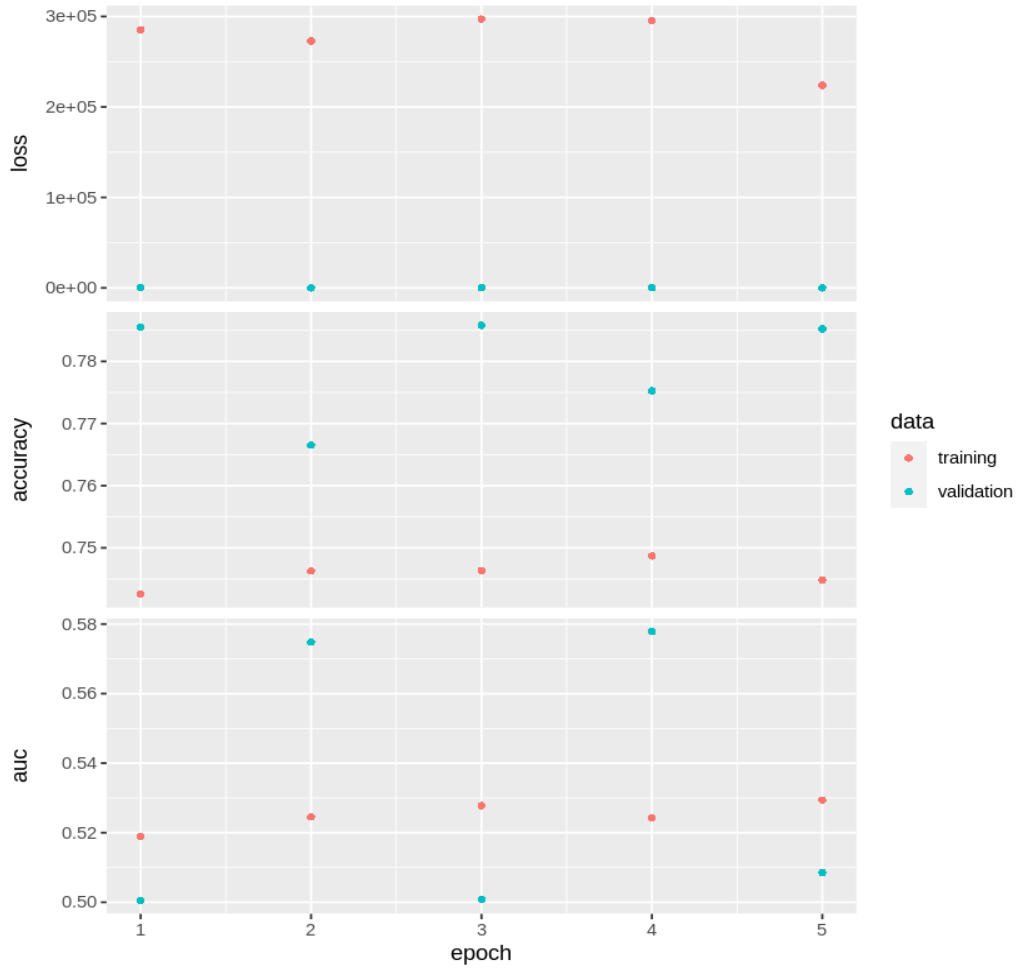

Figure 21. Training and Validation Metrics by Epoch.

The full results of the training and testing with the deep neural network model are summarized in Table 14 below. The model produced a true positive rate of 0.02, a false positive rate of 0.01, an overall accuracy of 0.79, an F1 score of 0.03 and an AUC of 0.51 against the test set.

Model #4: Deep Neural Network Model - Training Set													
Actual Class	Predicted Class		Totals		Actual Class	Predicted Class		TP	0.02	TP+TN	1.02	AUC	0.51
	0	1				0	1	TN	1.00	Precision	0.64	Sensitivity	0.02
0	11,537	38	11,575		0	1.00	0.00	Type I Error	0.00	Recall	0.02	Specificity	1.00
1	3,339	68	3,407		1	0.98	0.02	Type II Error	0.98	F1	0.04		
Model #4: Deep Neural Network Model - Test Set													
Actual Class	Predicted Class		Totals		Actual Class	Predicted Class		TP	0.02	TP+TN	1.01	AUC	0.51
	0	1				0	1	TN	0.99	Precision	0.47	Sensitivity	0.02
0	5,645	29	5,674		0	0.99	0.01	Type I Error	0.01	Recall	0.02	Specificity	0.99
1	1,522	26	1,548		1	0.98	0.02	Type II Error	0.98	F1	0.03		

Table 14. Confusion Matrix and Classification Metrics for Deep Neural Network Model.

6. Comparison of Results

Table 15 below summarizes relevant performance metrics of each of the four models against the test set.

Performance Against Test Set					
Model	**Accuracy**	**Precision**	**Recall**	**F1**	**AUC**
Random Forest	**0.80**	**0.56**	**0.34**	**0.48**	**0.63**
RF with XGBoost	**0.81**	**0.60**	**0.32**	**0.47**	**0.64**
Logisitic Regression	**0.81**	**0.57**	**0.36**	**0.51**	**0.65**
Deep Neural Network	**0.79**	**0.47**	**0.02**	**0.03**	**0.51**

Table 15: Summary of Performance Metrics for All Models Against Test Set.

The first thing of interest to note is that the deep neural network performed the worst. It classified nearly everything into the same class, resulting in F1 and recall scores that approached zero. This may be a case where the principle of parsimony should apply. This is a data set with relatively few features, especially when compared to the number of observations. In addition, some of those features appear to be correlated with each other, resulting in some redundancy in the feature set. It may be a case of overkill to apply a DNN to such a data set and the performance of the DNN seems to bear that out.

Another curiosity is that although the random forest model appears to have memorized the training set, it still did not overfit the test set much, if at all. It performance on the test set was nearly as good as the two best performers.

Those two best performers on the test set were the logistic regression and the XGBoost-augmented random forest. The logistic regression model had the best AUC, F1 and recall against the test set, making it the best overall performer by a slight margin. Less may indeed have been more in this classification task because a simple linear model outperformed powerful non-linear ones like the DNN and XGBoost.

The final observation to make is that none of the models classified the test set very well. The best accuracy was 81%, the best F1 was 0.51 and the best AUC only 0.65. It may well be that the target variable is simply not all that well explained by the predictor variables in these data.

7. Conclusions

A binary classification problem of credit risk modelling was examined. Raw predictor variables were engineered in an attempt to create effective predictors for the classification task at hand. The engineered features were examined through exploratory data analysis to determine their importance to the task. Four different types of predictive models were then tried in this analysis: a random forest, the boosted random forest model XGBoost, a logistic regression and a deep neural network.

The consumer credit data proved to be a challenging set to classify for both linear and non-linear models. Among the four different models, none achieved greater than 81% accuracy, an F1 of 0.51 or an AUC of 0.65.

A logistic regression model was the most performant, outpacing several non-linear models including a deep neural network. This looks to have been an instance where the simplest solution was the best, although only by a small margin. The target variable may be inherently resistant to classification by the features present in this particular set of data.

8. Bibliography

Yeh, I., & Lien, C. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications*. 36, 2473-2480.