Final Project Write Up

Group 8: Kanivalavan Venkatesan and Bindi Shakya

DSS 680 Predictive Analytics

Prof. Eddie Amadie

Saint Joseph’s University

**Objective**

The objective of our project is to predict how capable each applicant is of repaying a loan. The target is binary. 0: the loan was repaid(No for Default) or 1: the loan was not repaid (Yes for Default).

Doing so will ensure that clients capable of repayment are not rejected and that loans are given with a principal, maturity, and repayment calendar that will empower their clients to be successful.

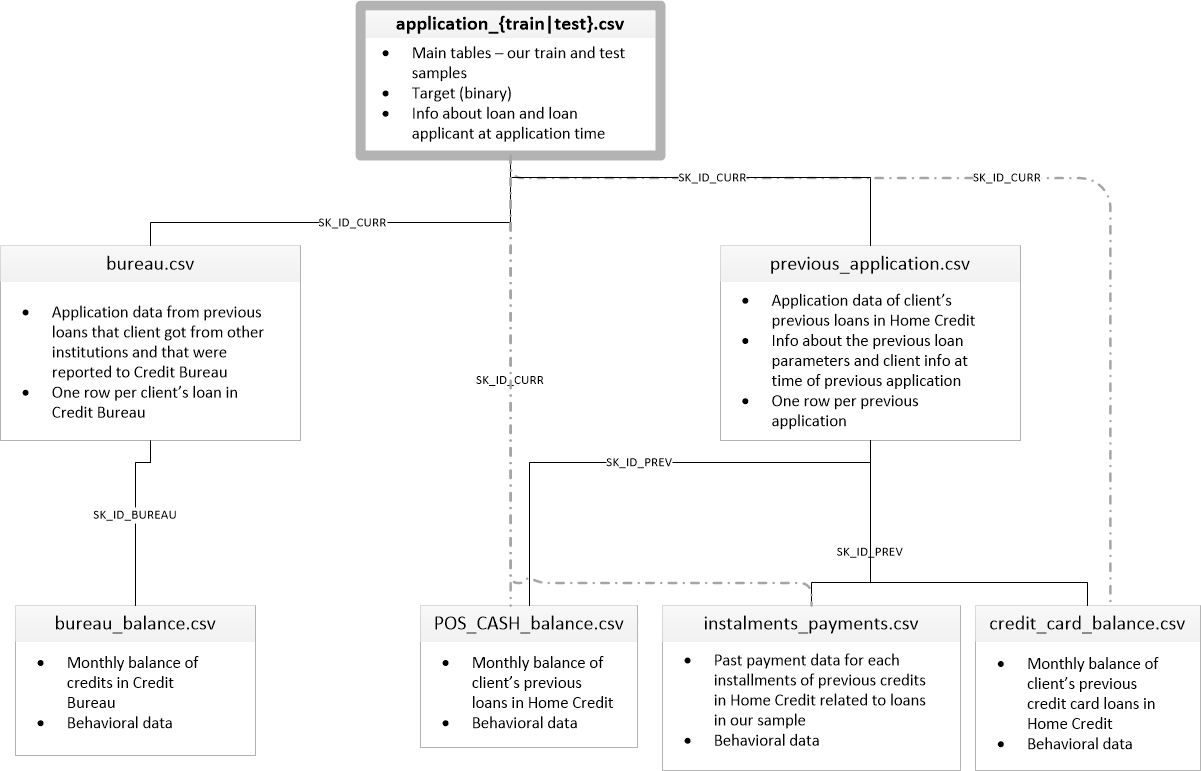
**About the dataset**

Many people struggle to get loans due to insufficient or non-existent credit histories. And, unfortunately, this population is often taken advantage of by untrustworthy lenders.

Founded in 1997, Home Credit Group is an international consumer finance provider with operations in 10 countries. They focus on responsible lending primarily to people with little or no credit history. Home Credit strives to broaden financial inclusion for the unbanked population by providing a positive and safe borrowing experience. In order to make sure this underserved population has a positive loan experience; Home Credit makes use of a variety of alternative data-including telco and transactional information-to predict their clients' repayment abilities.

While Home Credit is currently using various statistical and machine learning methods to make these predictions, they issued a challenge on Kaggle to help them unlock the full potential of their data.

The dataset has the following information:



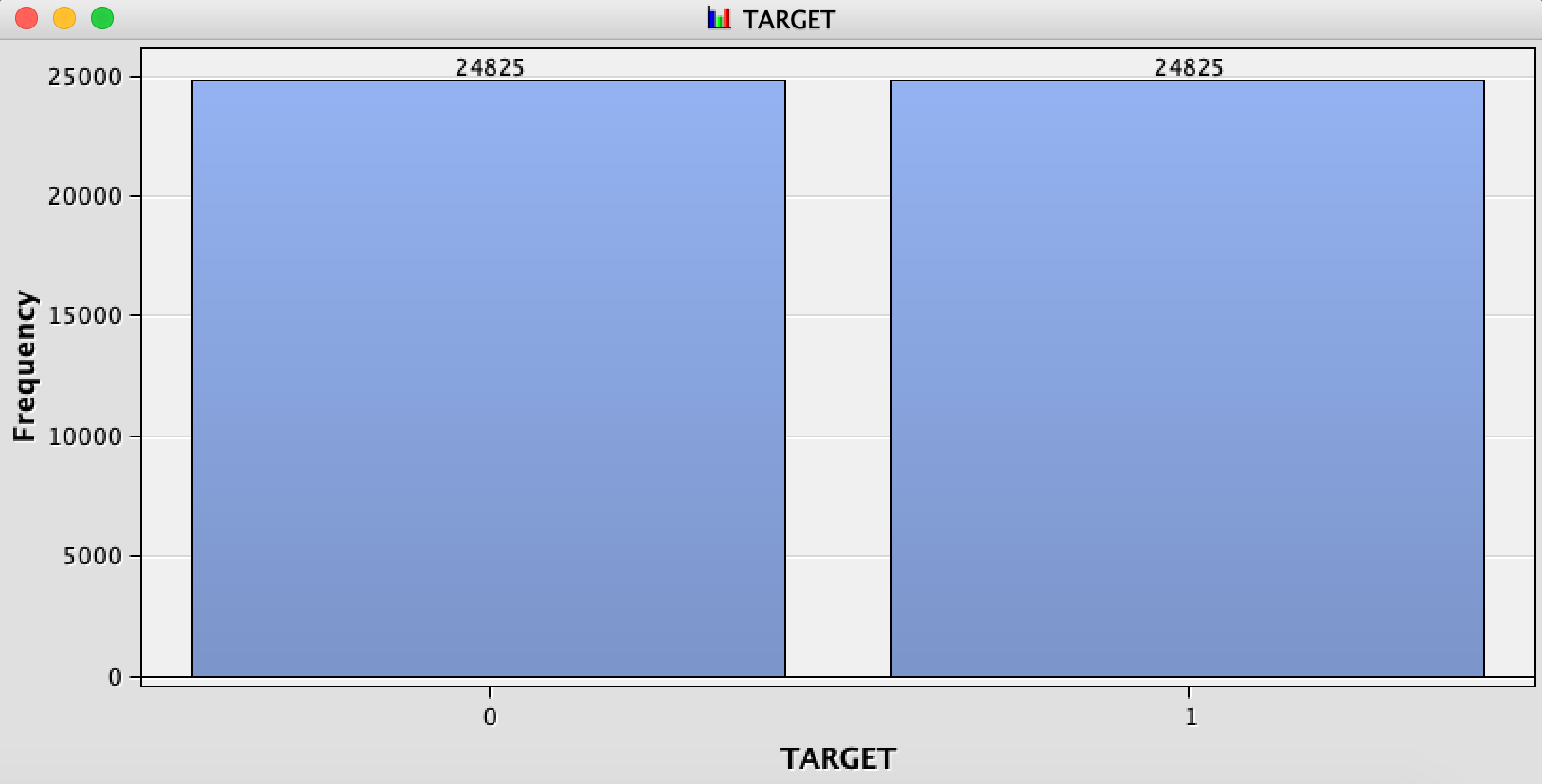
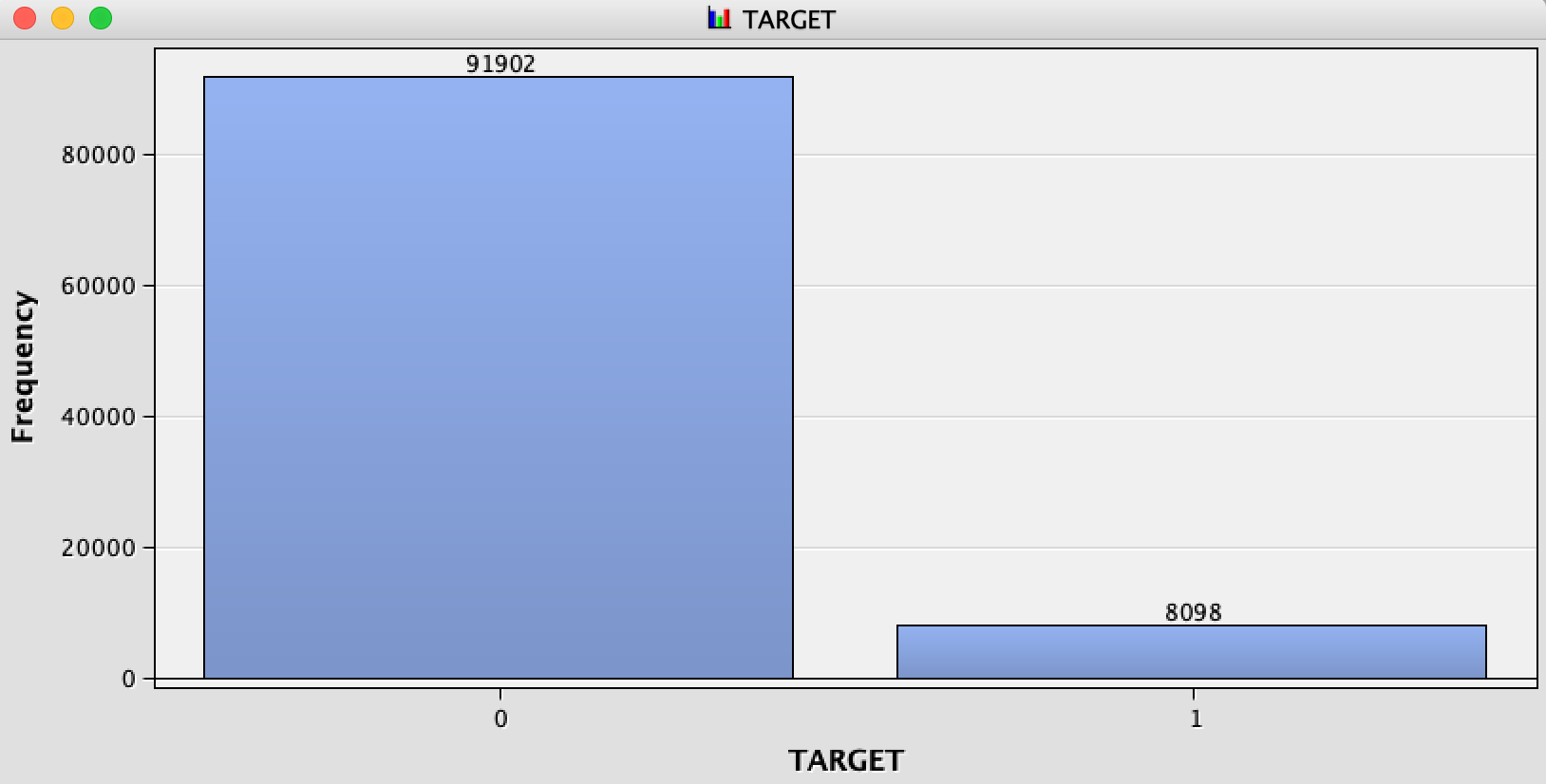
* application\_{train|test}.csv : This is the main table, broken into two files for Train (with TARGET) and Test (without TARGET). Static data for all applications. One row represents one loan in our data sample.
* bureau.csv: All client's previous credits provided by other financial institutions that were reported to Credit Bureau (for clients who have a loan in our sample). For every loan in our sample, there are as many rows as number of credits the client had in Credit Bureau before the application date.
* bureau\_balance.csv: Monthly balances of previous credits in Credit Bureau. This table has one row for each month of history of every previous credit reported to Credit Bureau – i.e the table has (#loans in sample \* # of relative previous credits \* # of months where we have some history observable for the previous credits) rows.
* POS\_CASH\_balance.csv: Monthly balance snapshots of previous POS (point of sales) and cash loans that the applicant had with Home Credit. This table has one row for each month of history of every previous credit in Home Credit (consumer credit and cash loans) related to loans in our sample – i.e. the table has (#loans in sample \* # of relative previous credits \* # of months in which we have some history observable for the previous credits) rows.
* credit\_card\_balance.csv: Monthly balance snapshots of previous credit cards that the applicant has with Home Credit. This table has one row for each month of history of every previous credit in Home Credit (consumer credit and cash loans) related to loans in our sample – i.e. the table has (#loans in sample \* no of relative previous credit cards \* no of months where we have some history observable for the previous credit card) rows.
* previous\_application.csv: All previous applications for Home Credit loans of clients who have loans in our sample. There is one row for each previous application related to loans in our data sample.
* installments\_payments.csv: Repayment history for the previously disbursed credits in Home Credit related to the loans in our sample. There is a) one row for every payment that was made plus b) one row each for missed payment. One row is equivalent to one payment of one installment OR one installment corresponding to one payment of one previous Home Credit credit related to loans in our sample.
* HomeCredit\_columns\_description.csv: This file contains descriptions for the columns in the various data files.

**What are some independent variables?**

This dataset is very large. The dataset has around 308,000 rows and 122 columns (121 independent variables) and has a wide range of information related to the loan application. Some of the independent variables are Gender, Days employed, Education Type, Amount Credit, Family status, etc.

**Issues faced during data exploration and preparation**

* This was an imbalanced class problem. There are far more loans that were repaid on time than loans that were not repaid. Around 91% of the loans were repaid, whereas only around 9% defaulted. We used Stratified Random Sampling to fix this issue. In stratified random sampling, we divide the population into smaller subgroups, or strata, based on the members’ shared attributes and characteristics. Then we take a random sample from each stratum in a number that is proportional to the size of the stratum and finally pool the subsets of the strata together to form a random sample.
* There were many missing values. We imputed the missing values. We used mean for interval variables and mode for categorical variable. We had to increase the cut off in the Impute Node to 65% missing values, as one of our important variable (as indicated by Decision Tree with Missing Values) was being left out.
* Building the model with 121 variables would make the model more complex and harder to interpret. So, we had to reduce the features. Used Decision tree and stepwise regression to reduce the features.
* Detailed Attribute information is missing. We have searched through the Kaggle competition to find the description/Metadata for all the variables, but we could not find it. We know what type of information is in the dataset from the file descriptions, but we do not know what each independent variable description is ( for instance, we do not know what EXT\_SOURCE\_1 means).
* Many of the independent variables were not normally distributed and so we applied logarithmic transformation to fix this issue.
* Some of our categorical independent variables had too many levels (eg occupation) which made the model very complex. Therefore, we chose to omit that variable.

**Before and after sampling**

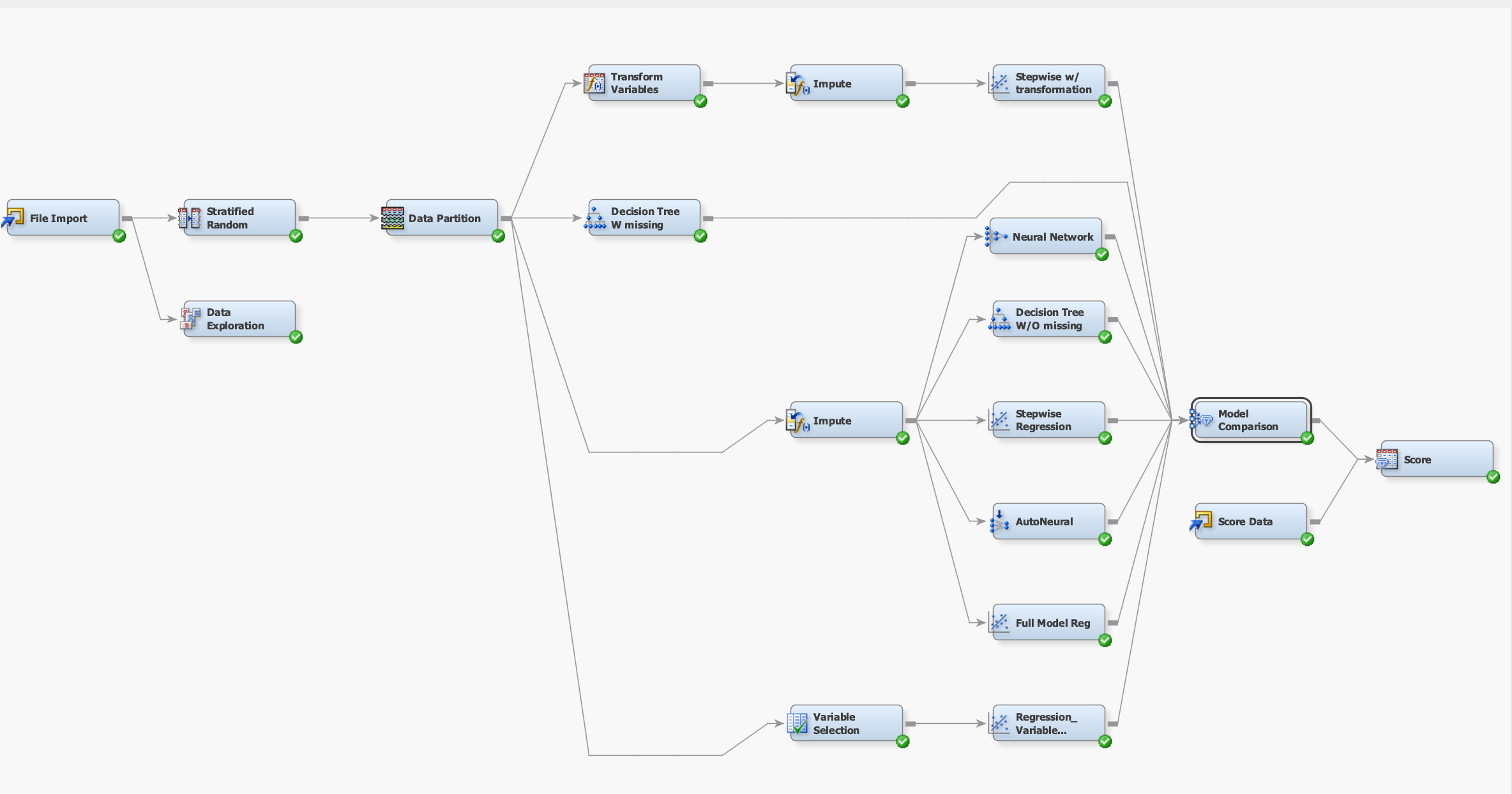
**Data partition**

We split 80% of the data for Training and 20% for Validation.

**Modelling**

We used the following models:

1. Decision Tree with missing values
2. Decision Tree with missing values imputed
3. Regression Full Model
4. Stepwise Regression
5. Neural Network
6. Auto Neural Network



To Briefly explain the flow of our project, initially we had to deal with unbalanced data. So we used stratified random sampling technique to overcome it. Then we partitioned data into 80-20. We noticed that some of the significant variables are skewed, so we did logarithmic transformation to reduce the skewness of the variables. Most importantly, we had to deal with missing data. We had around 40 variables with over 50 % of missing values. We imputed the variables with up to 65% of missing values with mean for interval variables and with mode for categorical variables. After we cleaned, transformed and imputed our data, the data became ready for use to build the models. We tried different models in different ways as stated above to understand how the models are performing. After numerous analysis, we identified stepwise regression with transformed variables as the best model to predict whether an applicant will be able to repay the loan or not. The results of our analysis are discussed in the topics that follow.

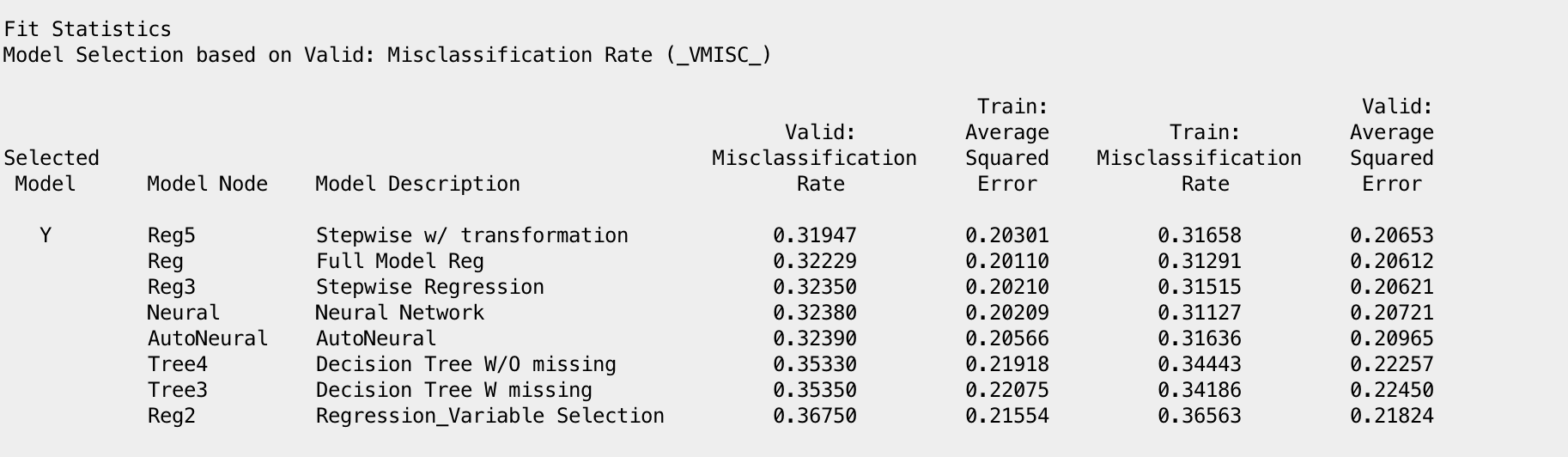
Node Names with description:

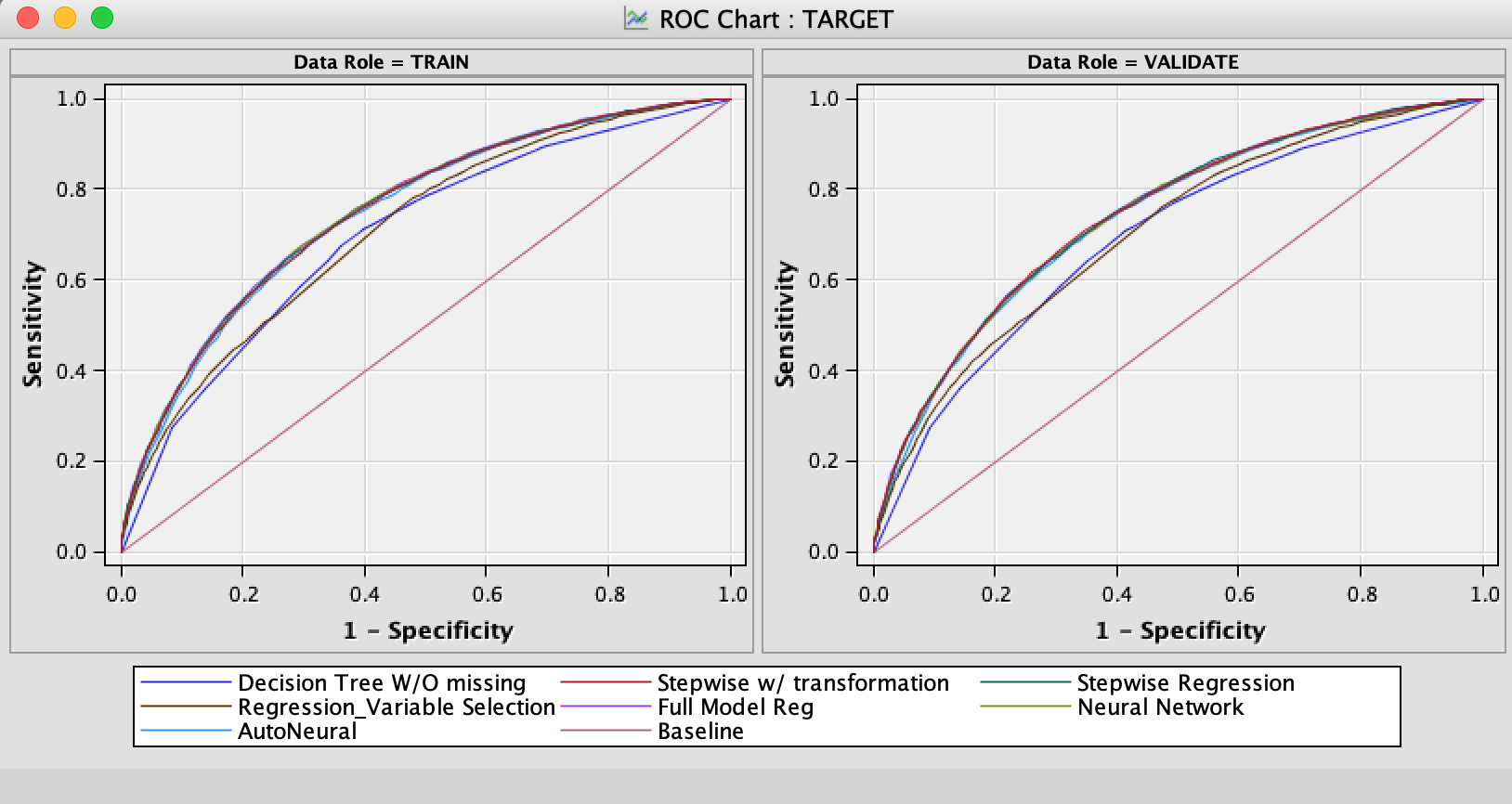
1. Reg5: Stepwise Regression with transformation
2. Tree4: Decision trees without missing values
3. Reg: Full Regression Model
4. Reg3: Stepwise Regression without transformation
5. Neural Network
6. Auto Neural
7. Tree3: Decision trees with missing values
8. Reg2: Regression with Variable Selection

We also scored the data using the winning model.

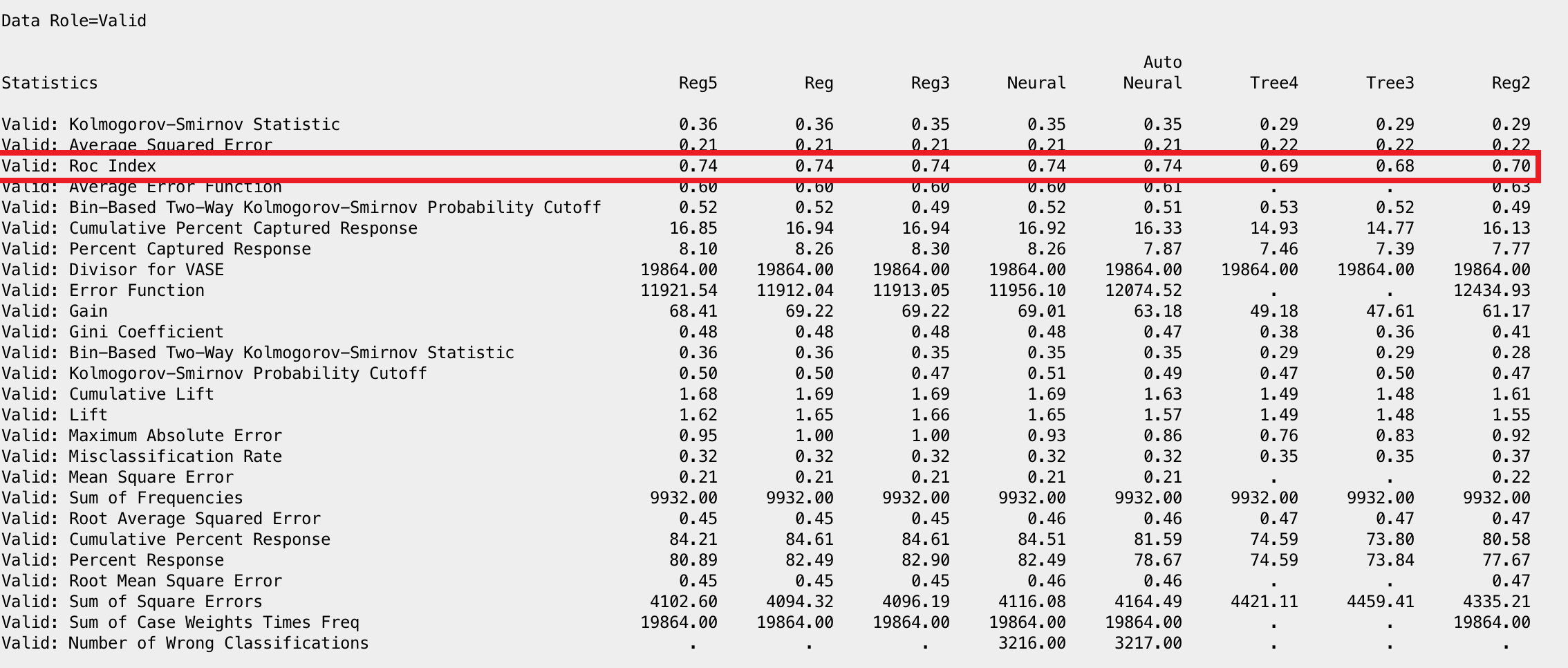
**Model evaluation**

We used Misclassification Rate, Area under ROC curve and Lift chart to assess our models. Based on these measures, we could see from the below fit statistics and graphs that stepwise regression with transformation has the lowest **misclassification rate of 0.31947, Highest ROC index of 0.74.**

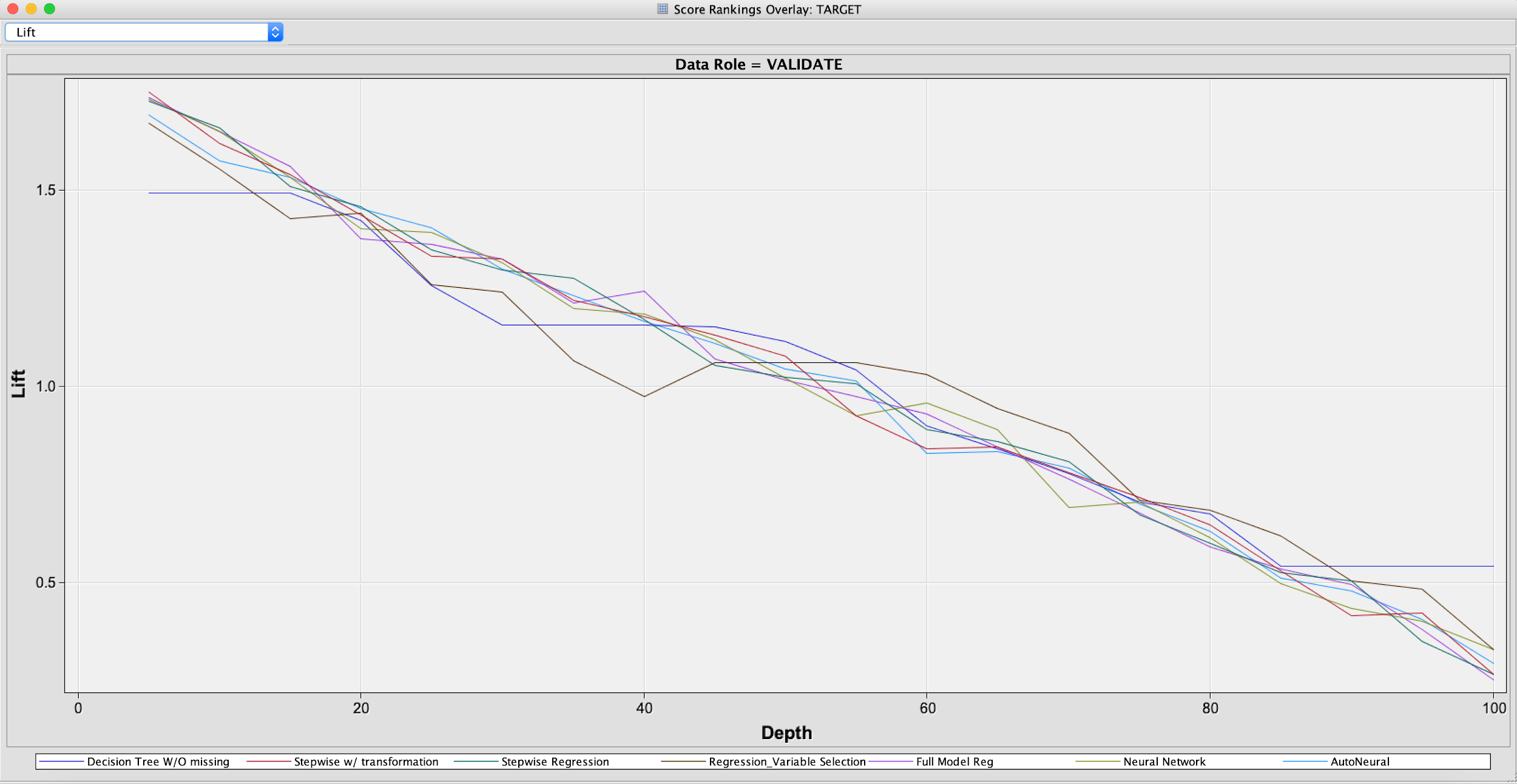
Misclassification Rate for all Models

ROC Chart for all Models:

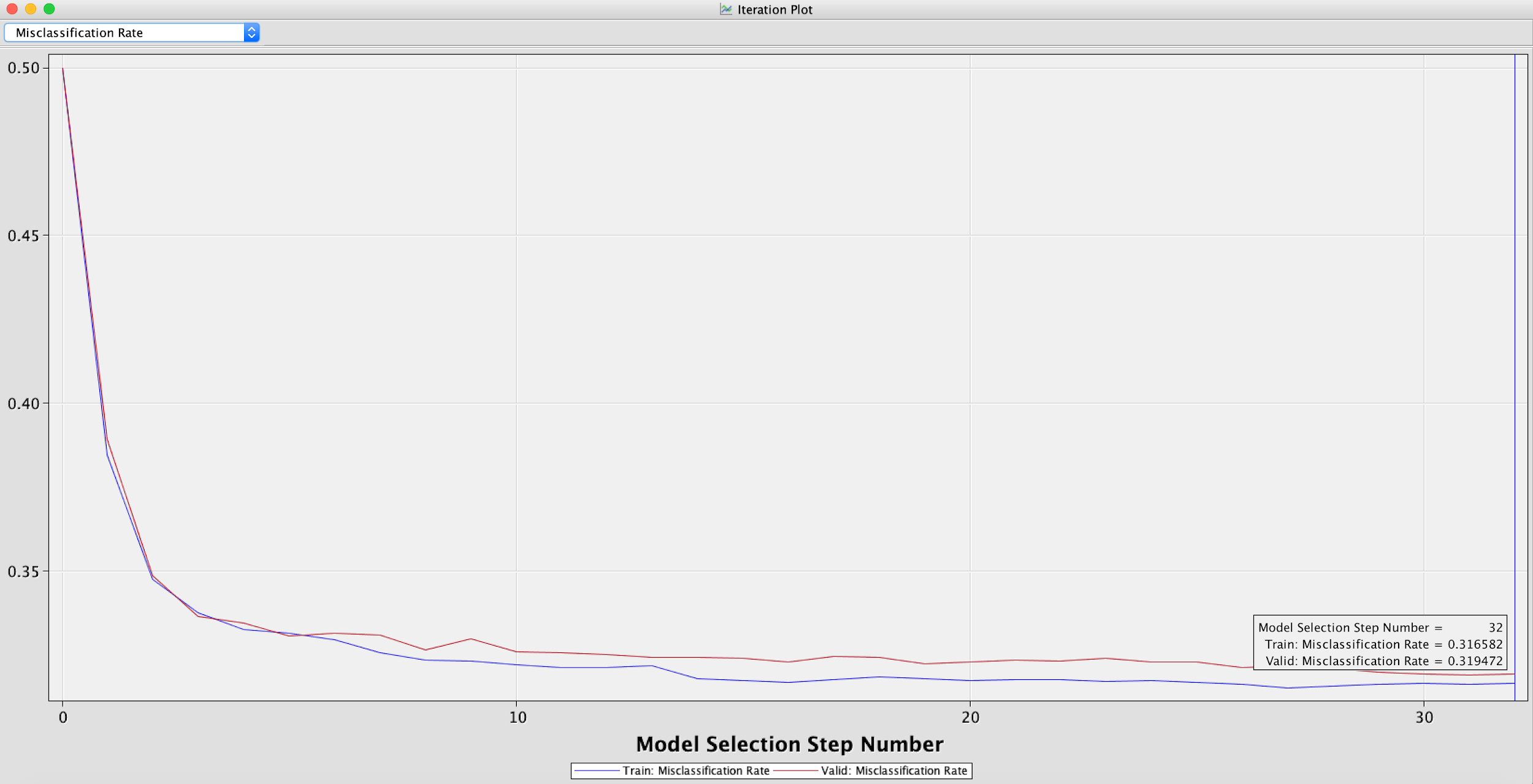
Area under the curve (ROC) for all Models



Lift Chart for all Models:

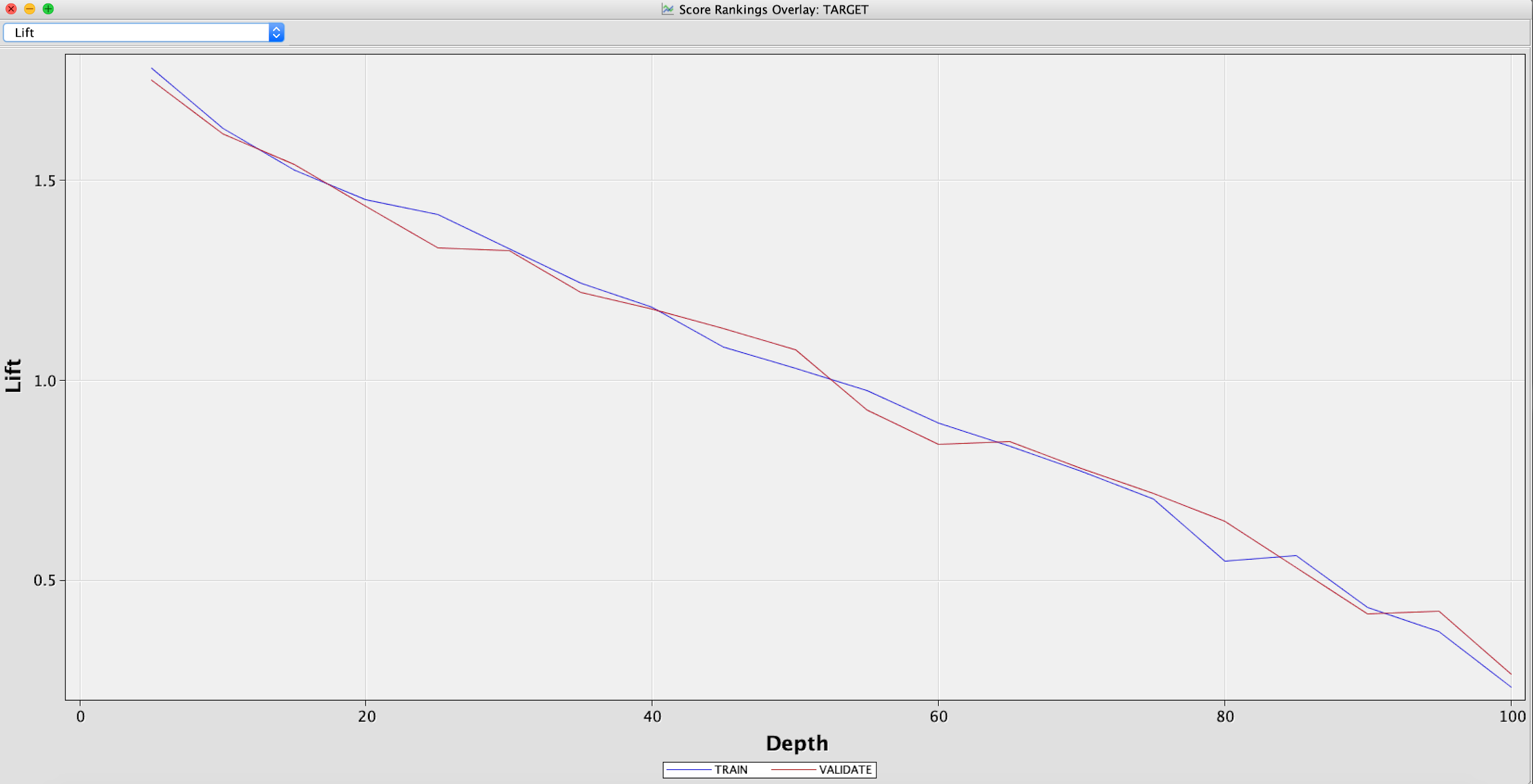


Misclassification Rate for our winning model :Stepwise Regression

From the below iteration plot, it is clear that there is no overfitting or underfitting in the selected model, since the model performs good with both the training and validation data set. We have the optimized model at step 32 with the lowest misclassification rate of **0.316582** for training and **0.319472** for validation dataset.

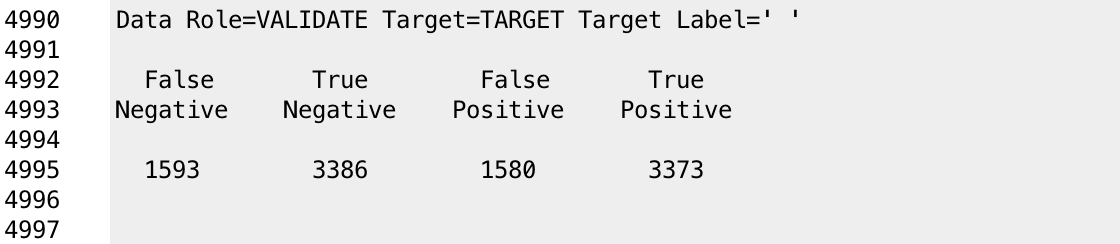
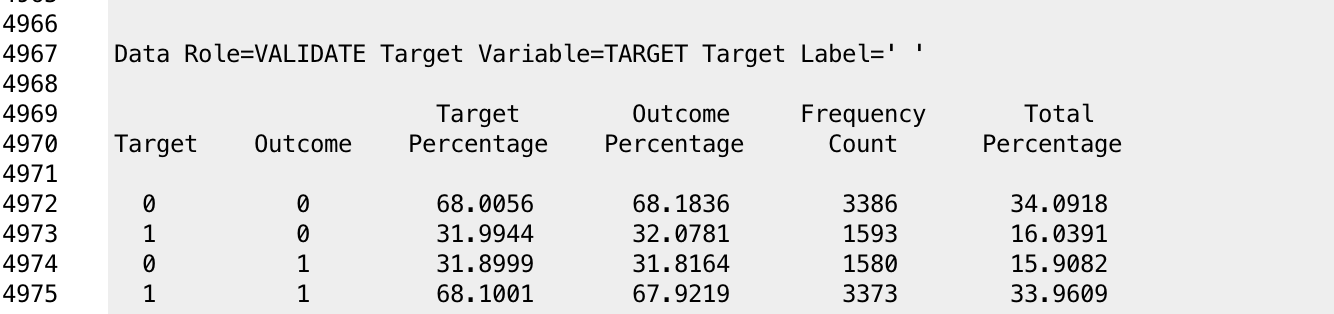
Lift Char**t** for our winning model :Stepwise Regression

From the Lift chart, we can see that by contacting 20% of the applicants using the predictive model, we can identify around 1.45 times as many loan defaulters as compared to using no model.

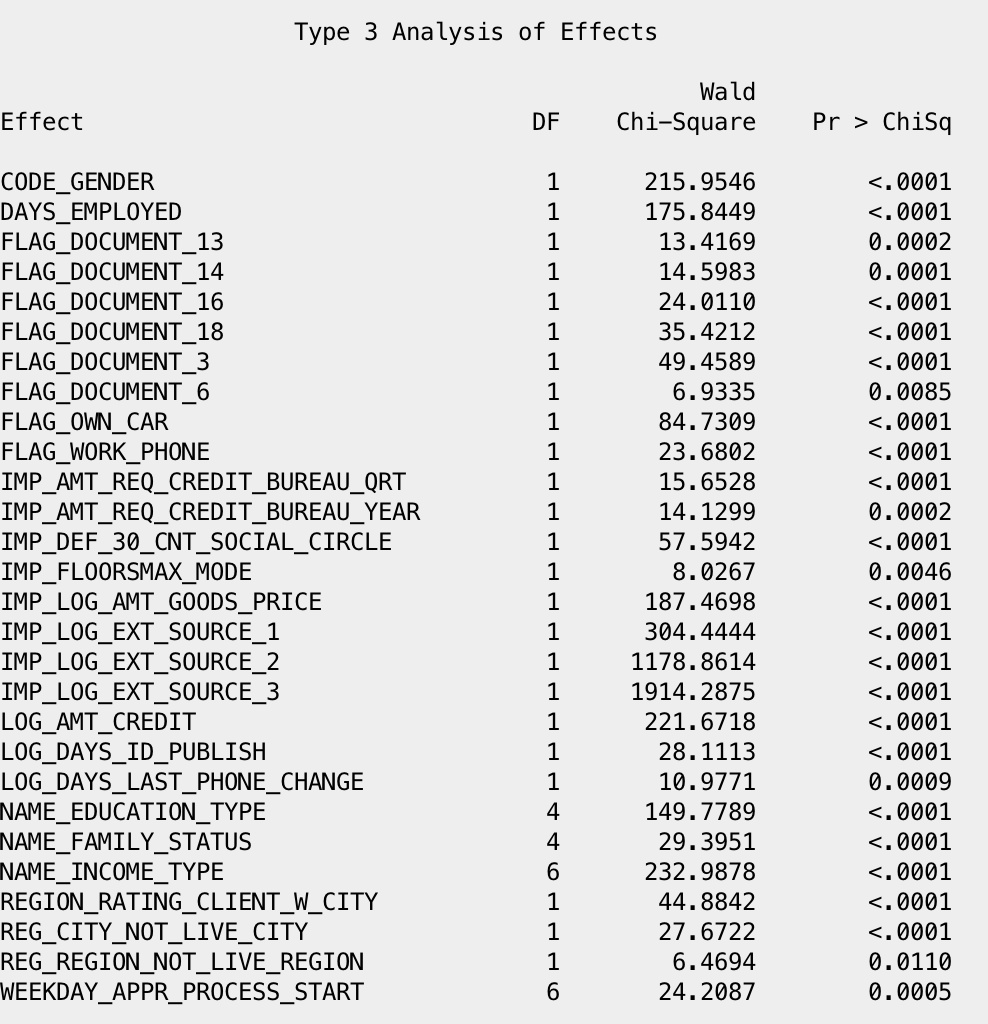
****

**Conclusion**

From our analysis, stepwise regression with transformation is the winning model with overall accuracy of 68%. This model is able to predict defaulter as defaulter, 67.92% of the time. Also, it is performing well for predicting non defaulters too. 69.25% of the time, this model is correctly predicting not a defaulter. As stated earlier, other important metrics such as lower misclassification rate (**0.31947)** and higher ROC index (**0.74**) substantiate our conclusion.

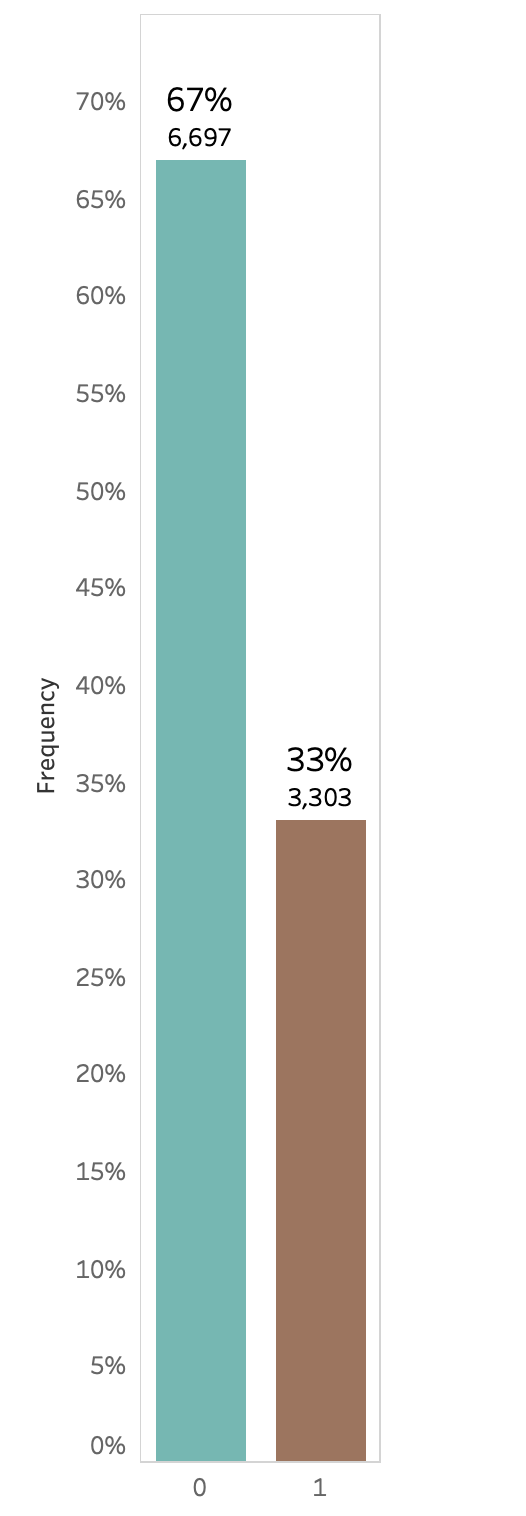


List of important variables:



**Model Deployment**

We scored the new dataset using the using the built model. Score dataset had 10,000 rows. Below is the distribution of the Predicted 0s(No for Default) and 1s (Yes for Default):



**Suggestions/Areas to explore further**

* Try classifying the applications based on risk (Probabilities near 1 have higher risk)
* Try Principal Components analysis: checking the correlation between variables
* Try regrouping/making a different set of dummy variables for Independent Categorical Variables which had too many levels
* Request Home Credit to provide detailed attribute information