Project 3: Predict Loan Default Risk

Business and Data Understanding

1. What decisions needs to be made?

As a loan officer at a young and small bank, I need to come up with an efficient solution to classify new customers on whether they can be approved for a loan or not.

Due to a financial scandal that hit a competitive bank last week, a sudden influx of 500 new people migrated to my bank, I need to use a series of classification models to figure out the best predictive model and provide a list of creditworthy customers in the next two days, in order to don't miss this huge opportunity for the bank that I work for.

2. What data is needed to inform those decisions?

- Data on all past applications:
  Credit Application Result, Account Balance, Duration of Credit Month, Payment Status of Previous Credit, Purpose, Credit Amount, Value Savings Stocks, Length of current employment, Instalment per cent, Most valuable available asset, Age years, Type of apartment, No of Credits at this Bank

- The list of customers that need to be processed in the next few days:
  Account Balance, Duration of Credit Month, Payment Status of Previous Credit, Purpose, Credit Amount, Value Savings Stocks, Length of current employment, Instalment per cent, Most valuable available asset, Age years, Type of apartment, No of Credits at this Bank

3. What kind of model (Continuous, Binary, Non-Binary, Time-Series) do we need to use to help make these decisions?

As the answer to this problem is Binary (creditworthy / non- creditworthy), we need to build a model that best fit with it. In order to achieve it, I'll compare the following binary classification models and choose the one that performs best:

- Logistic Regression;
- Decision Tree;
- Random Forest;
- Boosted Model

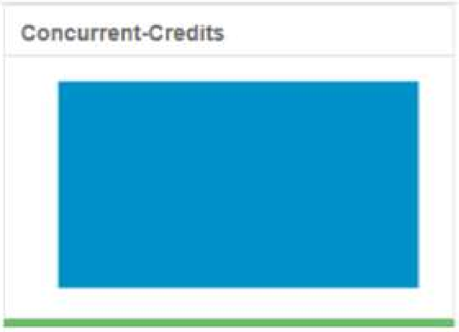
Building the Training Set

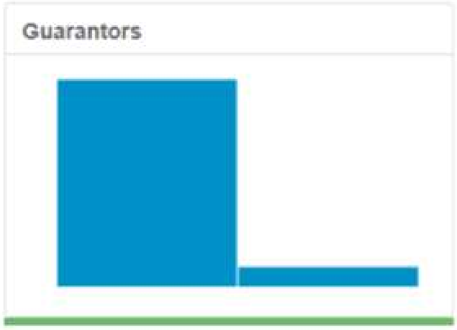
1. In your cleanup process, which fields did you remove or impute?

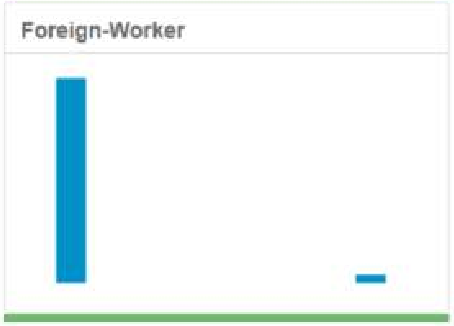
1.1. Data Exploration

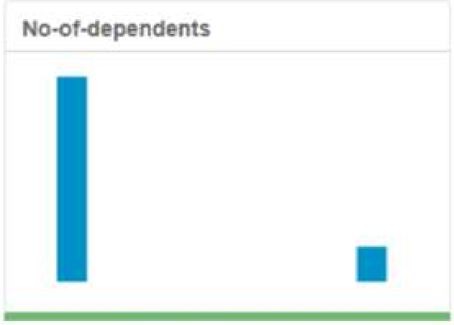
To the cleanup process, I started with Data Exploration, visualizing the data distribution and identifying which fields could be removed because of its "Low Variability". All the following graphs were classified as "Low Variability" and was removed:

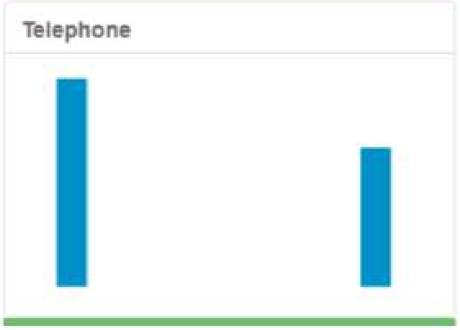
Because the "Duration-in-Current-address" field has a lot of missing data (69%) I also removed this field.

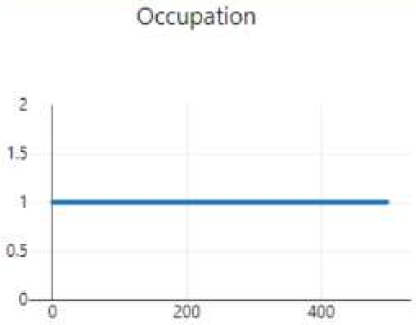


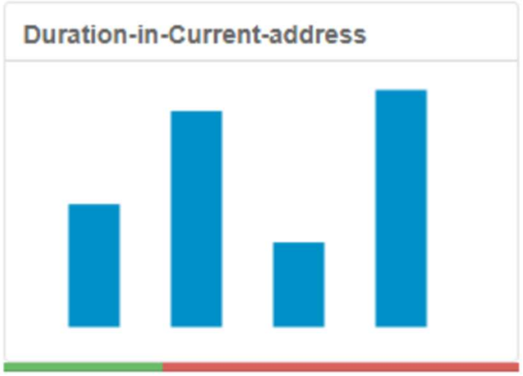










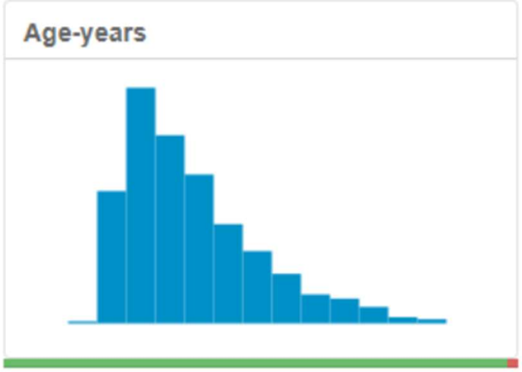


Now looking for the remaining fields can be identified some missing data in the “Age-years” field. As the missing data is only a few parts of the total (2%) and this field seems to be a high candidate to be an important variable to our prediction, I imputed the missing values with the median of the Age-years.

Although there are better techniques of value-imputation of the missing values, I used an imputation of median-values, which inputs less bias in the dataset than the average-values instead.

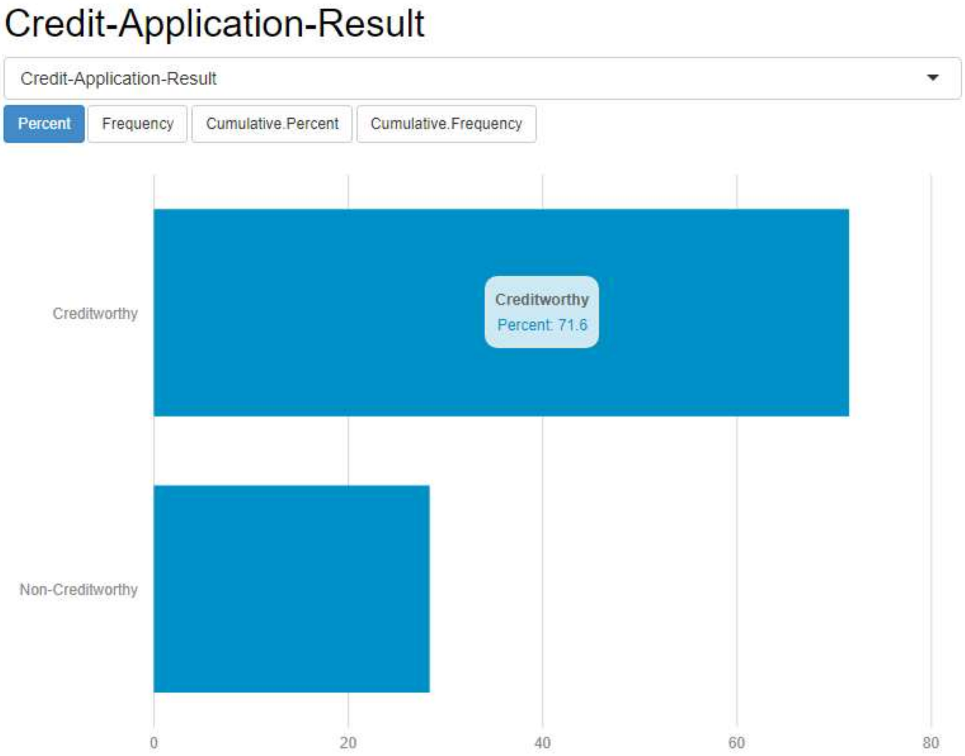
Using the "average values" could insert a skewed graph in our model based on the age of the current clients, in this case we would be inputting an error/bias in our model and it could return a biased analysis, for example: if we train/teach our model that people with 50+ years are the most reliable/creditworthy, most of the new customers that have less than 50 years probably wouldn't be accepted in our bank, but this concept of reliability was based in our current customers and this couldn't be (and probably isn't) an indisputable truth, especially when compared with a large amount of data.

When we're using the median of age-years, we're inputting the median value between the oldest and youngest, representing the age group that we attend in our bank. Even though this method input some errors in our dataset, at least our dataset isn't biased, and these errors tend to be lower than the average method when applying our model with a large age group.



1.2. Frequency of target-variable

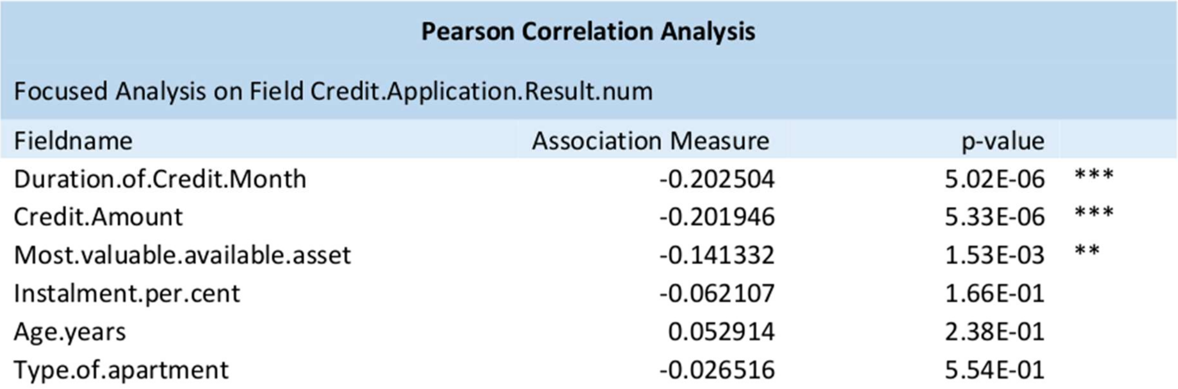
In order to analyze the frequency of the creditworthy and non-creditworthy in our dataset, we could see that our bank grants a lot of credit, probably because it's a young and small bank.



1.3. Predictors variables - Correlation

After applying logic to check if our list of potential variables has duplicates variables, now we will check the correlation between remaining variables.

I checked for correlations between my predictor variables to see if there is any possibility of multicollinearity in my dataset. Below is a table that shows the correlations between the different predictor variables:



Pearson Correlation Analysis

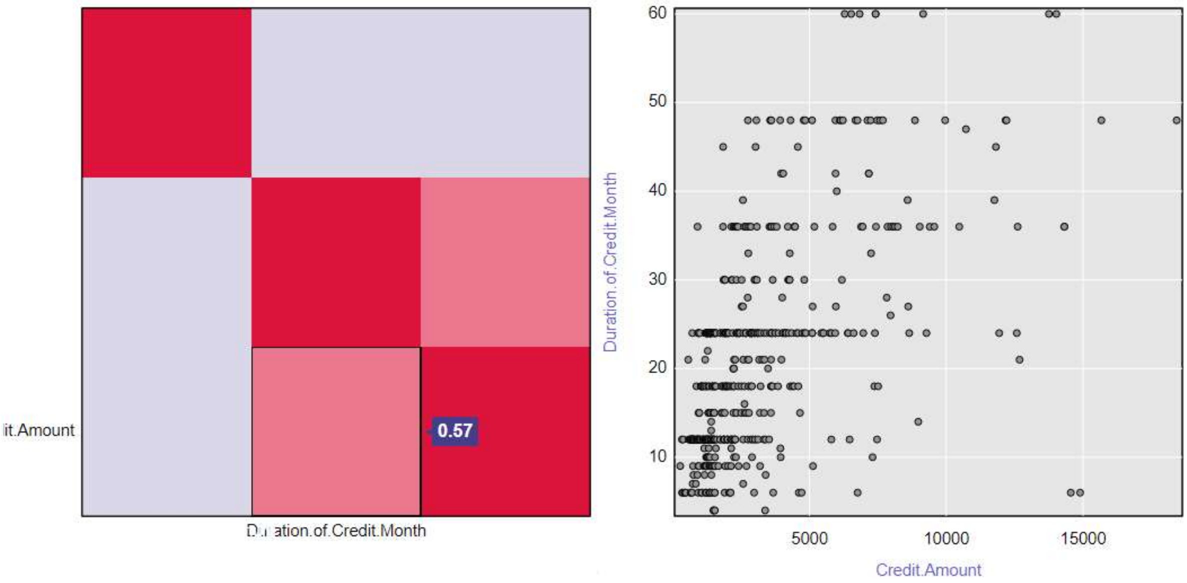
Focused Analysis on Field Credit.Application.Result.num

Fieldname	Association Measure	p-value	
Duration.of.Credit.Month	-0.202504	5.02E-06	***
Credit.Amount	-0.201946	5.33E-06	***
Most.valuable.available.asset	-0.141332	1.53E-03	**
Instalment.per.cent	-0.062107	1.66E-01	
Age.years	0.052914	2.38E-01	
Type.of.apartment	-0.026516	5.54E-01	

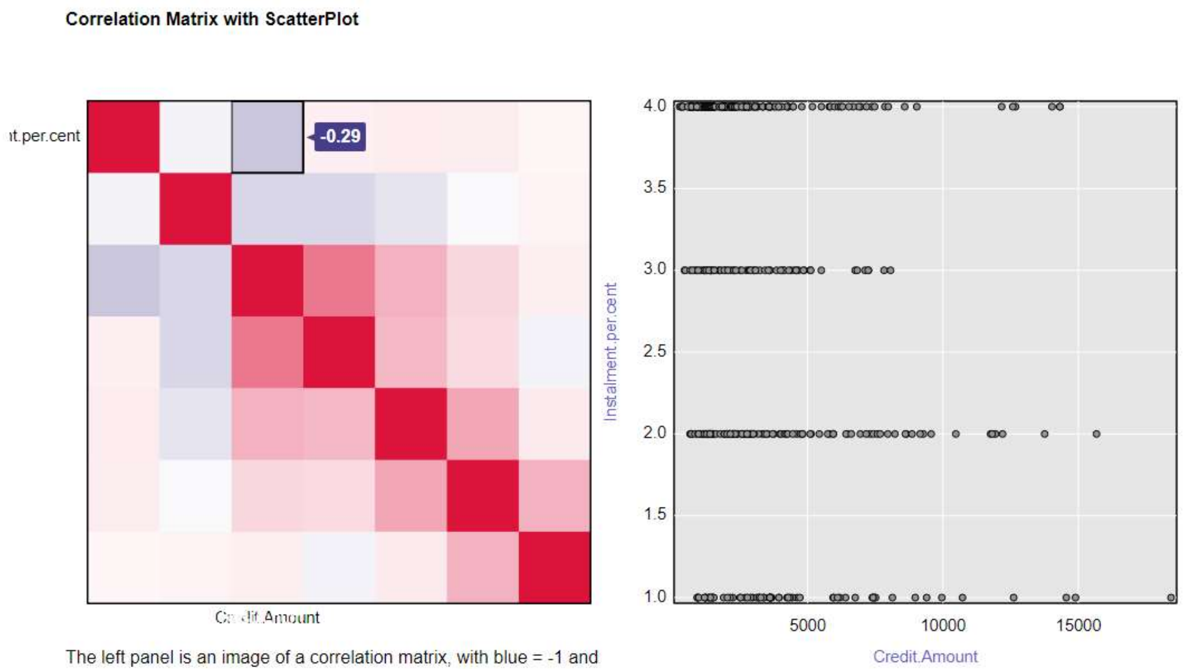
After analyzing the above table, now I wonder if there is an inner correlation between our first and second predictor variables (Duration of credit month and Credit amount) because the association measures between them are very close.

As we could see below, there's a significant correlation, but these variables haven't shown as a duplicate variable (correlation below 0.7), so we'll use them in our model.

Correlation Matrix with ScatterPlot



So, after analyzing the following last graph, I could conclude that all these variables are singular and can be used in our predict models



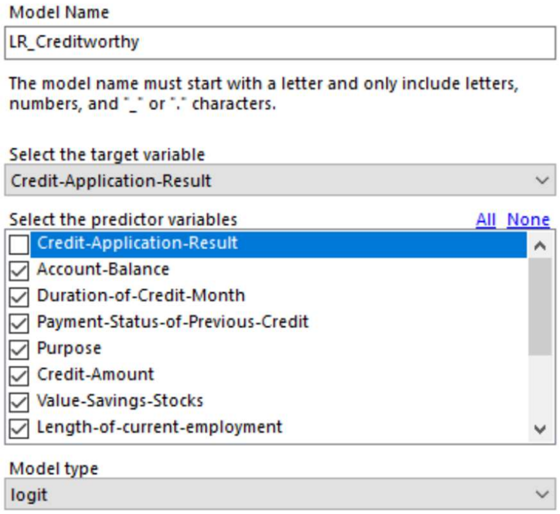
The left panel is an image of a correlation matrix, with blue = -1 and

Train your Classification Models

In order to obtain a reliable model which could predict/classify our customers, we need to train our model. First, I created an Estimation and Validation samples where 70% of the dataset were to Estimation and 30% of the entire dataset were reserved for Validation.

1. Logistic Regression

The first model we'll test is Logistic Regression. To let the Stepwise tool decide which predictor variables are significant, we chose all the variables in the Logistic Regression tool, other than the target variable.



Model Name

LR_Creditworthy

The model name must start with a letter and only include letters, numbers, and "_" or "." characters.

Select the target variable

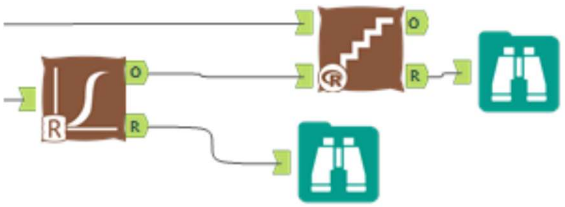
Credit-Application-Result

Select the predictor variables — All None

- ☐ Credit-Application-Result
- ☑ Account-Balance
- ☑ Duration-of-Credit-Month
- ☑ Payment-Status-of-Previous-Credit
- ☑ Purpose
- ☑ Credit-Amount
- ☑ Value-Savings-Stocks
- ☑ Length-of-current-employment

Model type

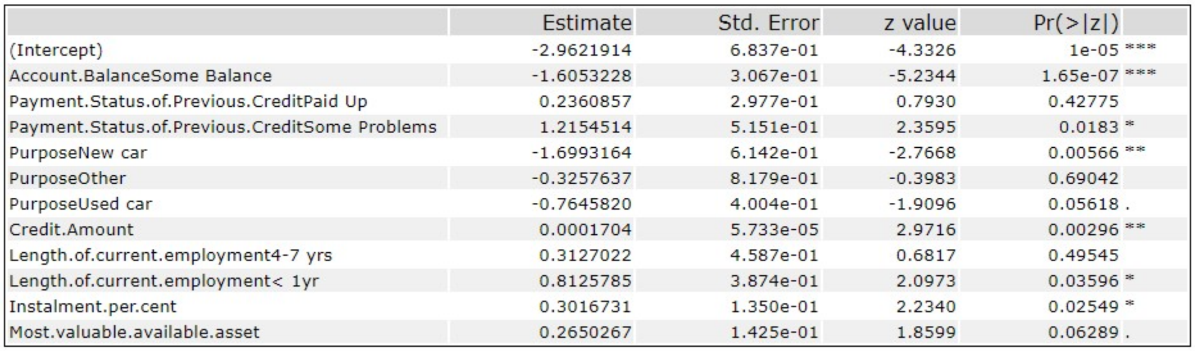
logit



According to the Logistic Regression model, the Predictor Variables that are significant are:

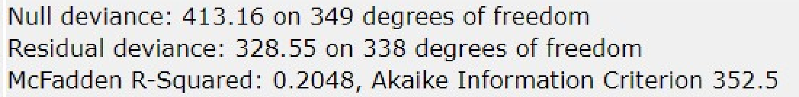
- Account-Balance
- Payment-Status-of-Previous-Credit
- Purpose

- Credit-Amount
- Length-of-current-employment
- Instalment-per-cent



	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-2.9621914	6.837e-01	-4.3326	1e-05 ***
Account.BalanceSome Balance	-1.6053228	3.067e-01	-5.2344	1.65e-07 ***
Payment.Status.of.Previous.CreditPaid Up	0.2360857	2.977e-01	0.7930	0.42775
Payment.Status.of.Previous.CreditSome Problems	1.2154514	5.151e-01	2.3595	0.0183 *
PurposeNew car	-1.6993164	6.142e-01	-2.7668	0.00566 **
PurposeOther	-0.3257637	8.179e-01	-0.3983	0.69042
PurposeUsed car	-0.7645820	4.004e-01	-1.9096	0.05618 .
Credit.Amount	0.0001704	5.733e-05	2.9716	0.00296 **
Length.of.current.employment4-7 yrs	0.3127022	4.587e-01	0.6817	0.49545
Length.of.current.employment< 1yr	0.8125785	3.874e-01	2.0973	0.03596 *
Instalment.per.cent	0.3016731	1.350e-01	2.2340	0.02549 *
Most.valuable.available.asset	0.2650267	1.425e-01	1.8599	0.06289 .

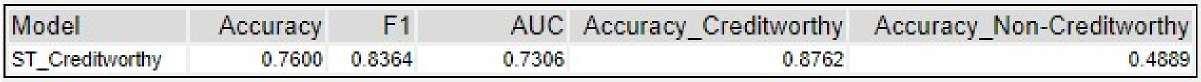
The R-Squared value sounds not good at all, with a value of 0.2048



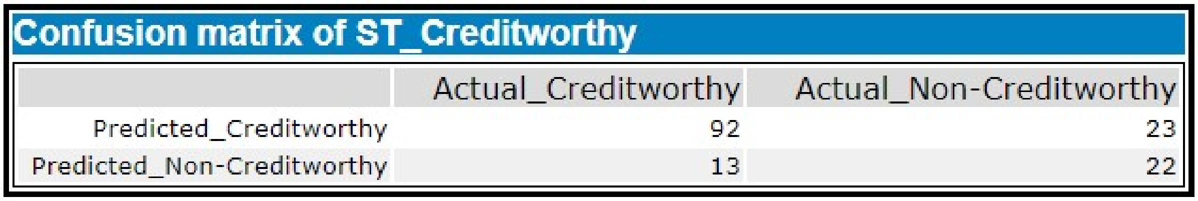
```
Null deviance: 413.16 on 349 degrees of freedom
Residual deviance: 328.55 on 338 degrees of freedom
McFadden R-Squared: 0.2048, Akaike Information Criterion 352.5
```

Now, if we look at how well this model would perform with the Validation data (30% of the dataset), we can see an overall percent of 0.7600, which isn't an excellent value, but it still a good prediction. If we look at confusion matrix, the accuracy in correctly predicting creditworthy individuals is 80% and the accuracy in correctly predicting non-creditworthy individuals 63%. This means that this particular model has bias towards correctly predicting creditworthy individuals because its accuracy in this segment is way higher than in the other.

Logistic Regression – Validation Data



Model	Accuracy	F1	AUC	Accuracy_Creditworthy	Accuracy_Non-Creditworthy
ST_Creditworthy	0.7600	0.8364	0.7306	0.8762	0.4889

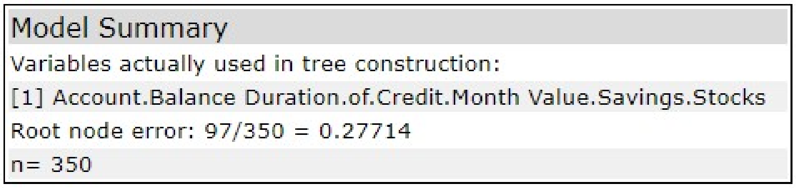


Confusion matrix of ST_Creditworthy

	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	92	23
Predicted_Non-Creditworthy	13	22

2. Decision Tree

Testing the Decision Tree model into our dataset we could see that even though the Root node error is quite high it still under 30%, which considered as an acceptable error.

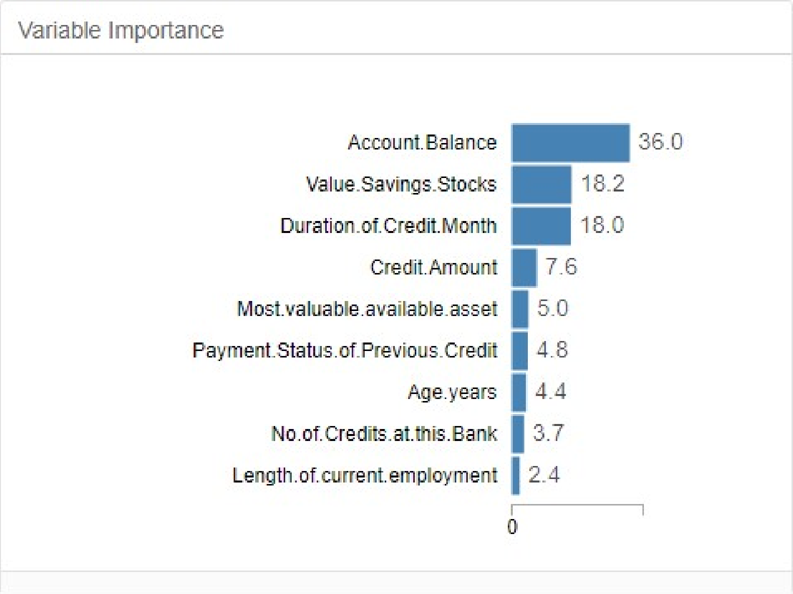


```
Model Summary
Variables actually used in tree construction:
[1] Account.Balance Duration.of.Credit.Month Value.Savings.Stocks
Root node error: 97/350 = 0.27714
n= 350
```

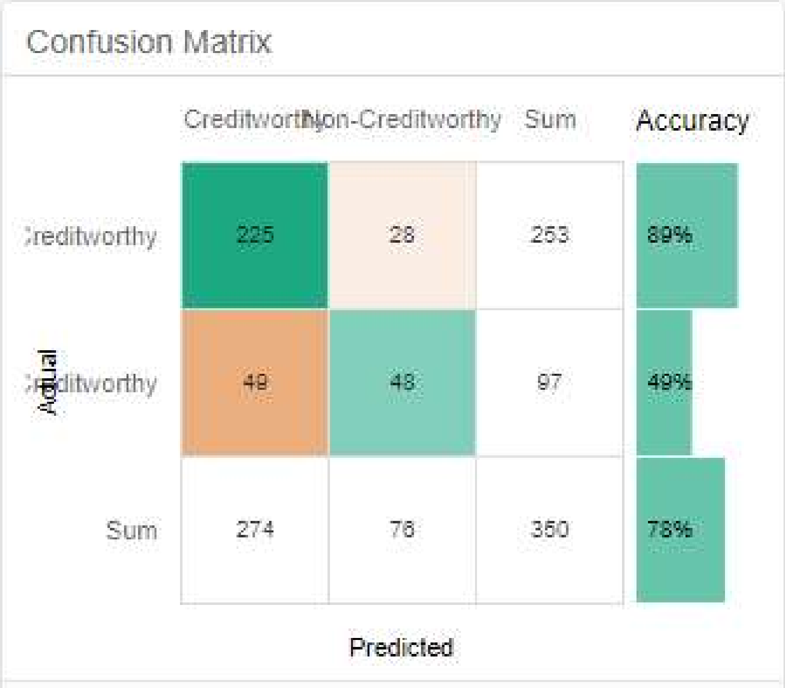
The Variable Importance Plot indicates that the most important predictor variables into this model are:

- Account-Balance
- Value-Saving-Stocks

- Duration-of-Credit-Month



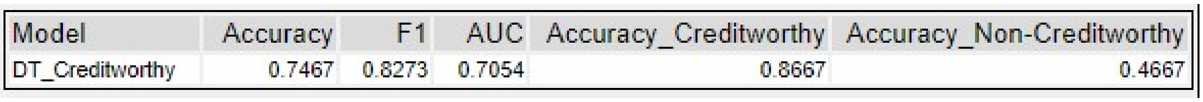
When we are validating our model against itself with the Confusion Matrix, we can see that the Sum of Accuracy is 78%, classifying it as a reliable model.


Confusion Matrix

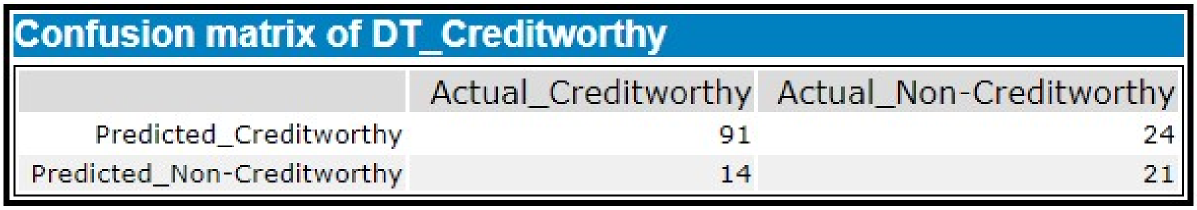
	Creditworth	Non-Creditworthy	Sum	Accuracy
:reditworthy	225	28	253	89%
:ditworthy	49	48	97	49%
Sum	274	76	350	78%

Looking at how well this model would perform with the Validation data, we can see that the overall percent accuracy is lower than the Logistic Regression Accuracy – 0.7467. If we look at confusion matrix, the analysis is similar to the Logistic Regression model, the accuracy in predicting creditworthy individuals is 79% and the accuracy in correctly predicting non-creditworthy individuals are 60%. Meaning that this particular model has bias towards correctly predicting creditworthy individuals because its accuracy in this segment is way higher than in the other.

Decision Tree – Validation Data



Model	Accuracy	F1	AUC	Accuracy_Creditworthy	Accuracy_Non-Creditworthy
DT_Creditworthy	0.7467	0.8273	0.7054	0.8667	0.4667



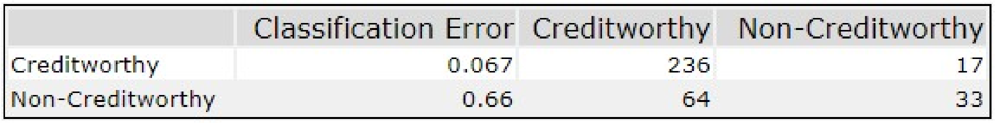
Confusion matrix of DT_Creditworthy

	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	91	24
Predicted_Non-Creditworthy	14	21

3. Forest Model

Looking at the Confusion Matrix of the Forest Model, we can see that the accuracy of this model (trained with only Estimation Data) is the best until now, with an overall value of 86%!

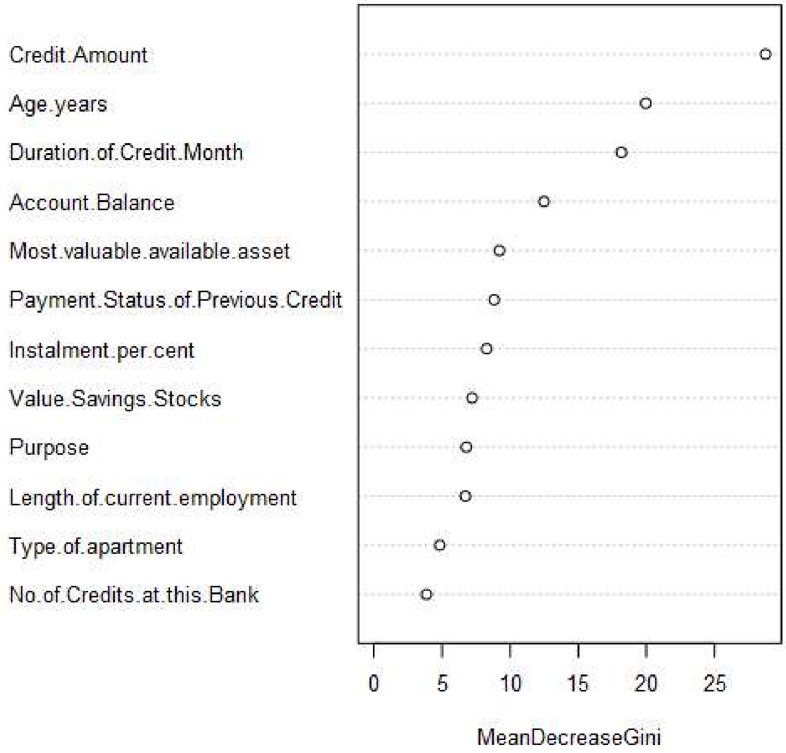
Confusion matrix – Forest Model – Estimation Data



	Classification Error	Creditworthy	Non-Creditworthy
Creditworthy	0.067	236	17
Non-Creditworthy	0.66	64	33

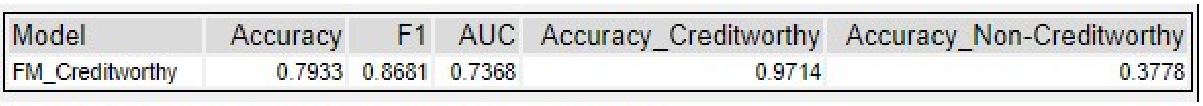
This model selected the Credit-Amount, Age-years and Duration-of-Credit-Month as the most important variables into this system.

Variable Importance Plot – Forest Model

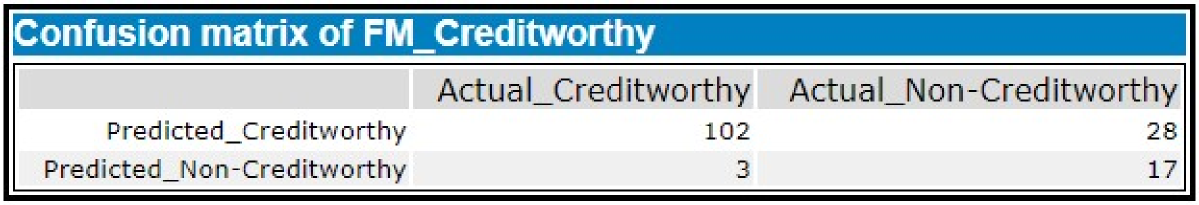


Comparing our model trained with Validation Data we can see the following results:

Forest Model – Validation Data



Model	Accuracy	F1	AUC	Accuracy_Creditworthy	Accuracy_Non-Creditworthy
FM_Creditworthy	0.7933	0.8681	0.7368	0.9714	0.3778



Confusion matrix of FM_Creditworthy

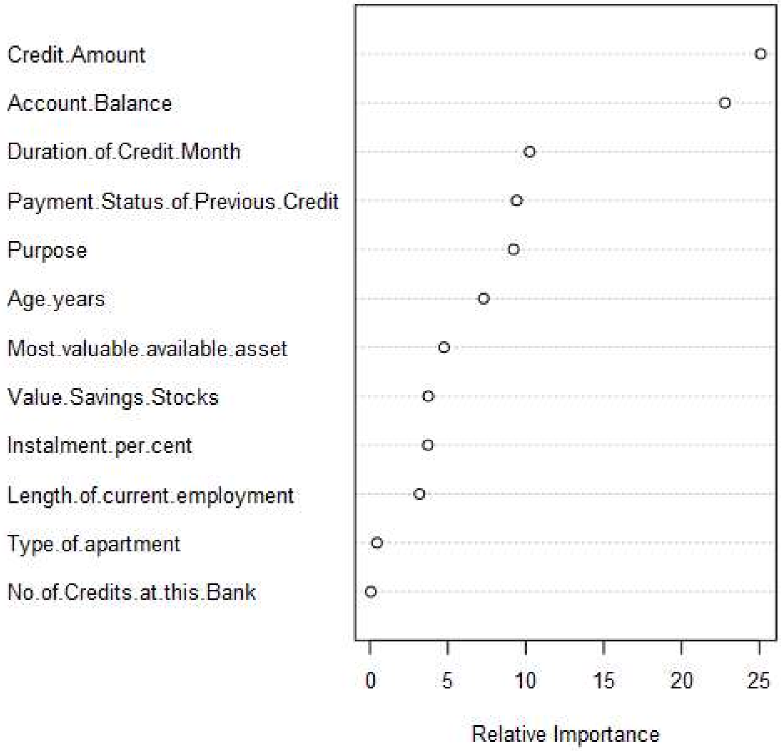
	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	102	28
Predicted_Non-Creditworthy	3	17

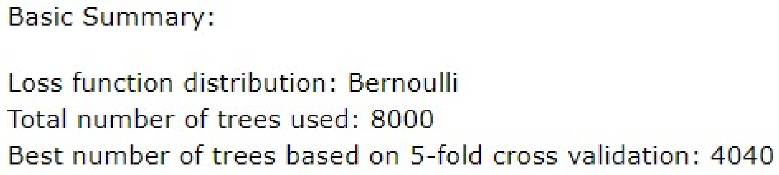
The accuracy of predicting creditworthiness in this model is about 78% and the accuracy of predicting non-creditworthiness is 85%. In this case, we can say that this model is almost not biased at all, because the difference between those accuracies is very small.

3. Boosted Model

When trained with Validation Data, the Boosted Model identify only Credit-Amount and Account-Balance as important variables to build this model.

Variable Importance Plot – Boosted Model

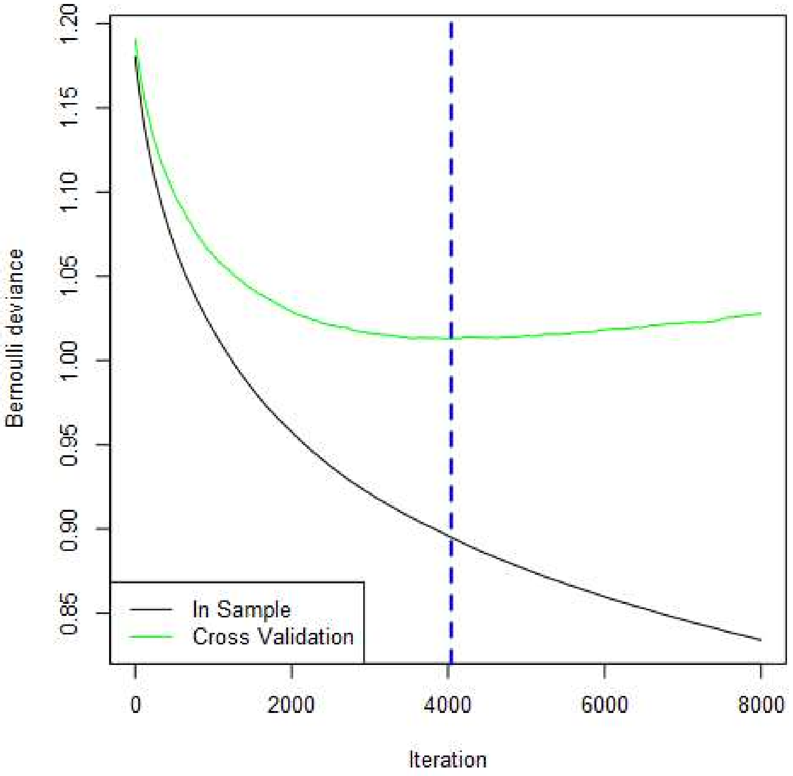




Basic Summary:

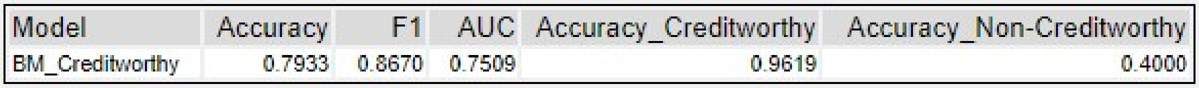
Loss function distribution: Bernoulli
Total number of trees used: 8000
Best number of trees based on 5-fold cross validation: 4040

Number of Iterations Assessment Plot

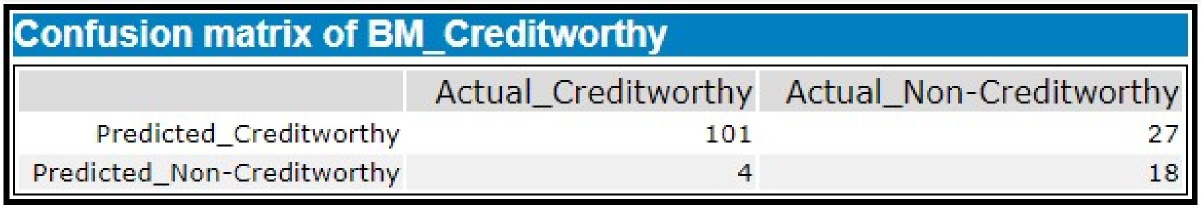


Comparing our model trained with Validation Data we can see the following results:

Boosted Model – Validation Data



Model	Accuracy	F1	AUC	Accuracy_Creditworthy	Accuracy_Non-Creditworthy
BM_Creditworthy	0.7933	0.8670	0.7509	0.9619	0.4000



Confusion matrix of BM_Creditworthy

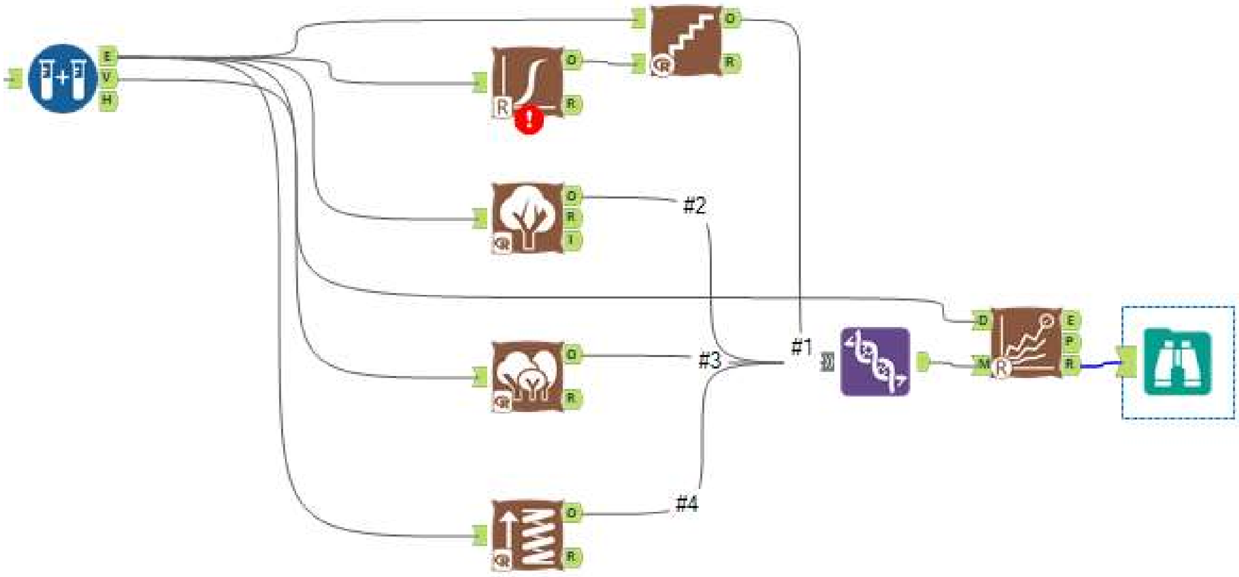
	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	101	27
Predicted_Non-Creditworthy	4	18

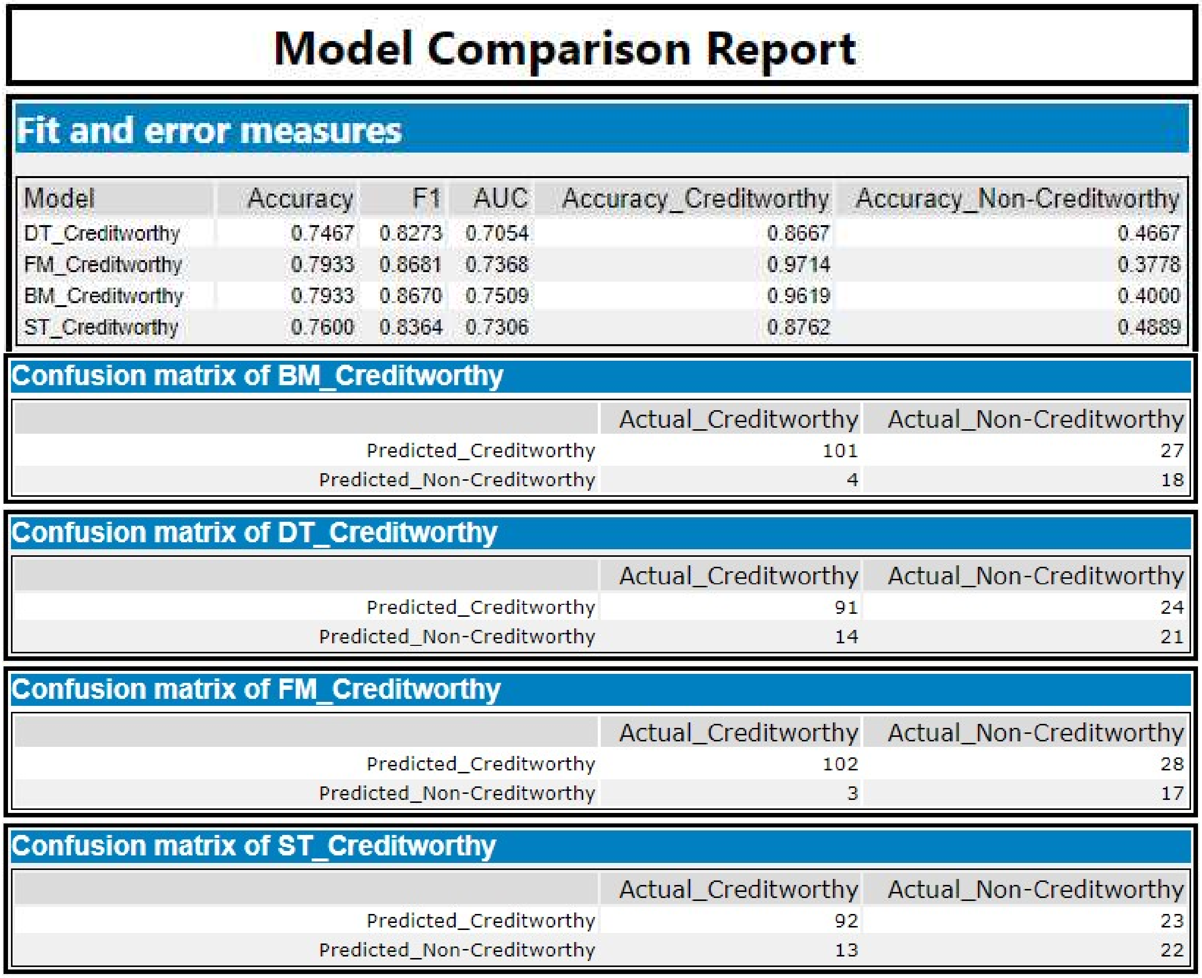
The accuracy of predicting creditworthiness in this model is 79% and 82% is the accuracy of predicting non-creditworthiness. In this case, we can say that this model also is almost not biased at all, because the difference between those accuracies is very small.

Writeup

1. Choosing the best Model

In order to look at the whole picture, we can compare all of these models side by side:





Model Comparison Report

Fit and error measures

Model	Accuracy	F1	AUC	Accuracy_Creditworthy	Accuracy_Non-Creditworthy
DT_Creditworthy	0.7467	0.8273	0.7054	0.8667	0.4667
FM_Creditworthy	0.7933	0.8681	0.7368	0.9714	0.3778
BM_Creditworthy	0.7933	0.8670	0.7509	0.9619	0.4000
ST_Creditworthy	0.7600	0.8364	0.7306	0.8762	0.4889

Confusion matrix of BM_Creditworthy

	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	101	27
Predicted_Non-Creditworthy	4	18

Confusion matrix of DT_Creditworthy

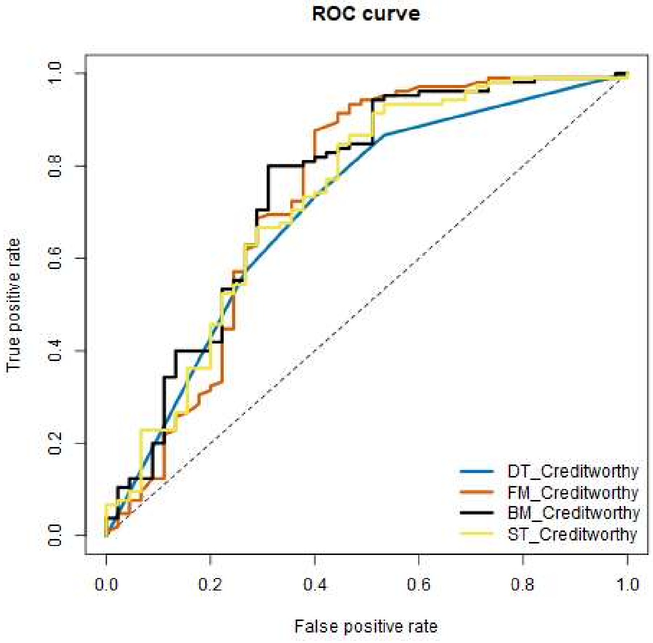
	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	91	24
Predicted_Non-Creditworthy	14	21

Confusion matrix of FM_Creditworthy

	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	102	28
Predicted_Non-Creditworthy	3	17

Confusion matrix of ST_Creditworthy

	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	92	23
Predicted_Non-Creditworthy	13	22

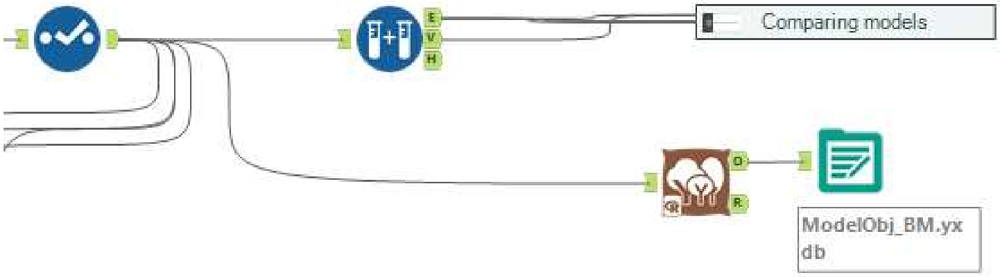


Taking over the overall accuracy to predict the best fit model, we came with the two bests models: Forest Model and Boosted Model, both with the highest overall accuracy of 79.33%, as well we can see that Forest model has the highest Accuracy Creditworthy at 97.14%. We're going further onto this analysis focusing only on these models because the most important on this analysis is how accurately we can identify people who qualify for loan.

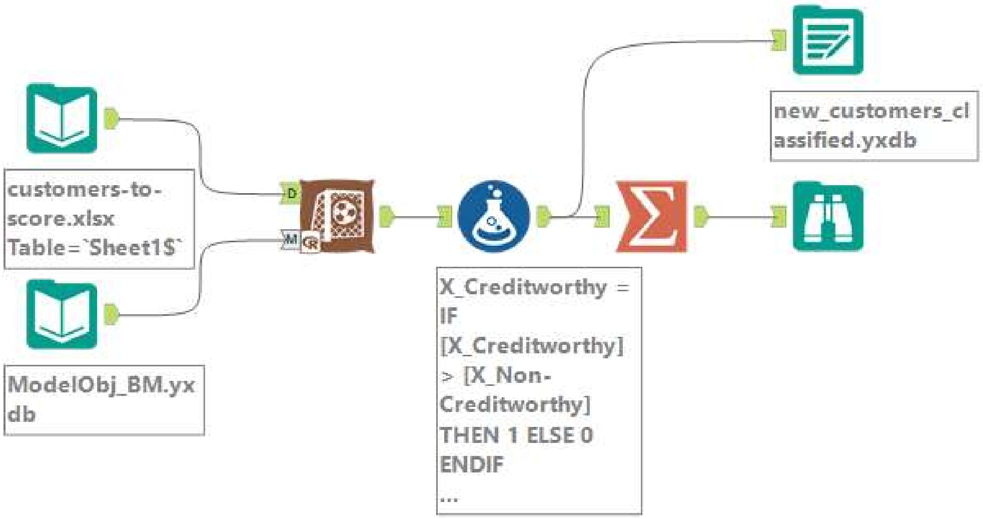
When we see at ROC graph, we can say that Forest model hugs the top most true positive side of the graph.
And, the Forest model also doesn't show any bias in predicting the models accuracy, hence, **we can say that Forest model is the best fit model for this project.**

2. How many individuals are creditworthy?

Now that we choose our trained model with Estimation and Validation data, we need to redo our model, but now with the whole dataset.



Scoring the missing data:


new_customers_classified.yxdb
customers-to-score.xlsx Table=`Sheet1$`
D
M
ModelObj_BM.yxdb

```
X_Creditworthy =
IF
[X_Creditworthy]
> [X_Non-
Creditworthy]
THEN 1 ELSE 0
ENDIF
...
```

With an overall accuracy of **79%** and **97%** of accuracy in predicting Creditworthy individuals, we can predict that **407** of the new customers can be classified as **Creditworthy**.

Creditworthy	Non-Creditworthy
407	**93**