Step 1: Business and Data Understanding

1. What decisions needs to be made?

The decision that needs to be made is to determine whether each of 500 new loan applicants is creditworthy or not. The threshold for classification is that the probability of a client being creditworthy is higher than that of his/her being non-creditworthy. This will later on helps the manager make a decision on whom to give a loan to.

2. What data is needed to inform those decisions?

We need to study past credit history of past customers with target variables and predictor variables. This will be used as input to the training classification models. After that, we need a list of new customers to put into production.

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

I based on the methodology map to determine the suitable types of model.

First, we are trying to predict whether each of new clients is creditworthy or not. Therefore, we want to predict an outcome instead of describing data.

Second, this is a data rich problem since we have both data from past clients and new client list with.

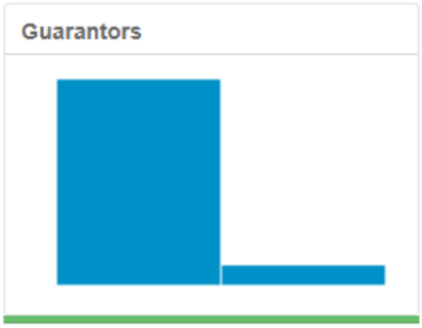
Third, since we want to group into two groups, it is going to be a binary classification model for which Logistic Regression, Decision Tree, Forest Model, or Boosted Model can be called for. Therefore, I will test all these four models, compare their performance, and decide which one to put into action.

Step 2: Building the Training Set

1. Data manipulation:

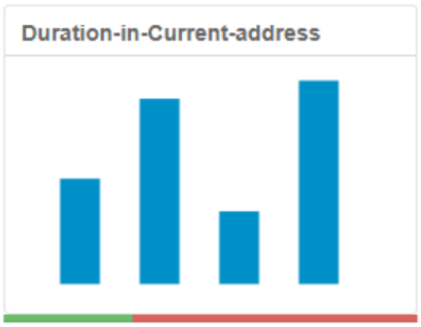
To prepare for the training model, I did transformation to the following fields:

Field #1: Guarantors:



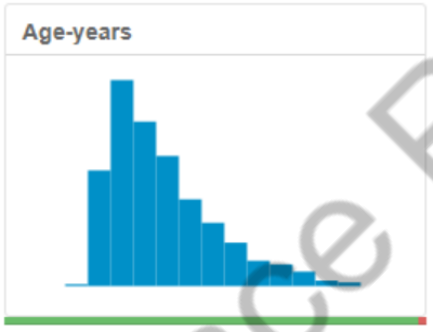
This variable has low variability as the data is heavily skewed towards one value. Therefore, I will remove this variable from the model.

Field #2: Duration-in-Current-Address



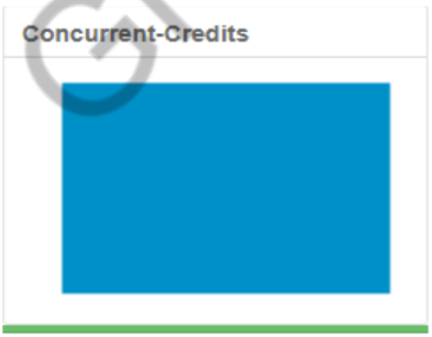
This variable has 68.8% of missing values (represented by the large red portion of the bar). Treating this variable with methods such as imputation or removal of null values might cause biases. Therefore, I will remove this variable.

Field #3: Age years



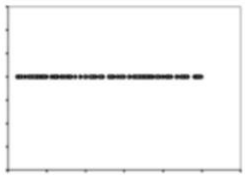
This variable has a low percentage of missing valuables (at 2.4%). Therefore, it is treatable with imputation without causing major impact to the model. Therefore, I will impute missing values with the median value. I choose the median over the mean as the **age-years** variable is right-skewed.

Field #4: Concurrent-Credits



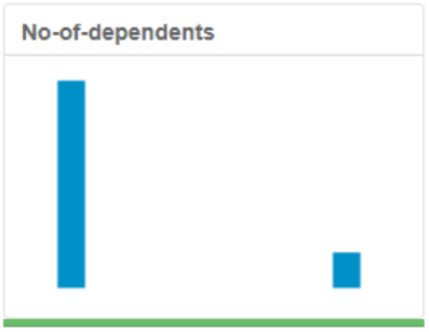
This variable has only one value. Therefore, I will remove it from the model.

Field #5: Occupation

Occupation		0.0%	1	1.000	1.000	1.000	1.000	0.000	This field has a small number of unique values, and appears to be a categorical field. Consider changing the field data type to "string".

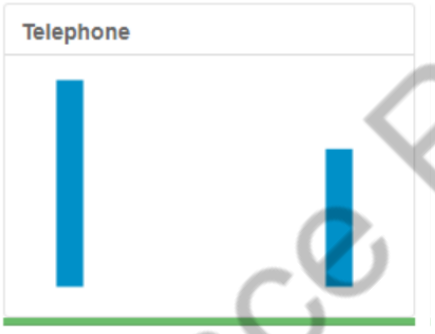
Like Concurrent-Credits, Occupation will also be removed for the same reason.

Field #6: No-of-Dependents



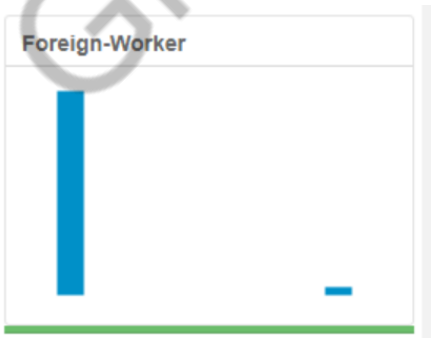
This variable has low variability as it is is heavily skewed towards one value over the other. Therefore, I will remove this variable.

Field #7: Telephone



It is intuitive that there is no connection between **telephone** and the **credit-application-status**. Therefore, I will remove this variable.

Field #8: Foreign-Worker



This variable has low variability as it is is heavily skewed towards one value over the other. Therefore, I will remove this variable.

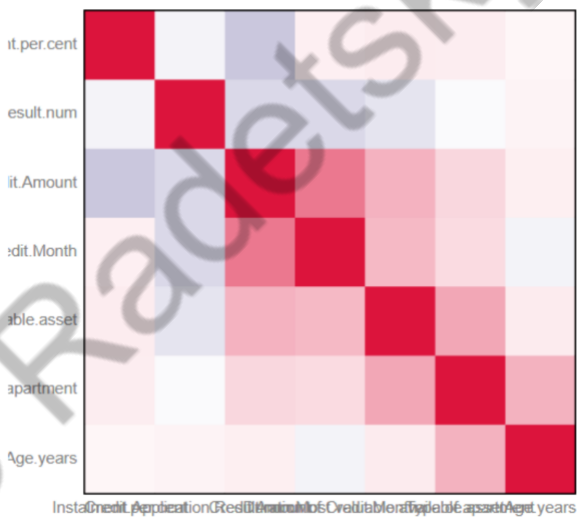
2. Correlation of variables

Next, I will check if there are any correlation between predictor variables or any duplicate predictor variables:

Full Correlation Matrix

	Credit.Application.Result.num	Duration.of.Credit.Month	Credit.Amount	Instalment.per.cent	Most.valuable.available.asset	Age.years
Credit.Application.Result.num	1.000000	-0.202504	-0.201946	-0.062107	-0.141332	0.052914
Duration.of.Credit.Month	-0.202504	1.000000	0.573980	0.068106	0.299855	-0.064197
Credit.Amount	-0.201946	0.573980	1.000000	-0.288852	0.325545	0.069316
Instalment.per.cent	-0.062107	0.068106	-0.288852	1.000000	0.081493	0.039270
Most.valuable.available.asset	-0.141332	0.299855	0.325545	0.081493	1.000000	0.086233
Age.years	0.052914	-0.064197	0.069316	0.039270	0.086233	1.000000
Type.of.apartment	-0.026516	0.152516	0.170071	0.074533	0.373101	0.329350

	Type.of.apartment
Credit.Application.Result.num	-0.026516
Duration.of.Credit.Month	0.152516
Credit.Amount	0.170071
Instalment.per.cent	0.074533
Most.valuable.available.asset	0.373101
Age.years	0.329350
Type.of.apartment	1.000000

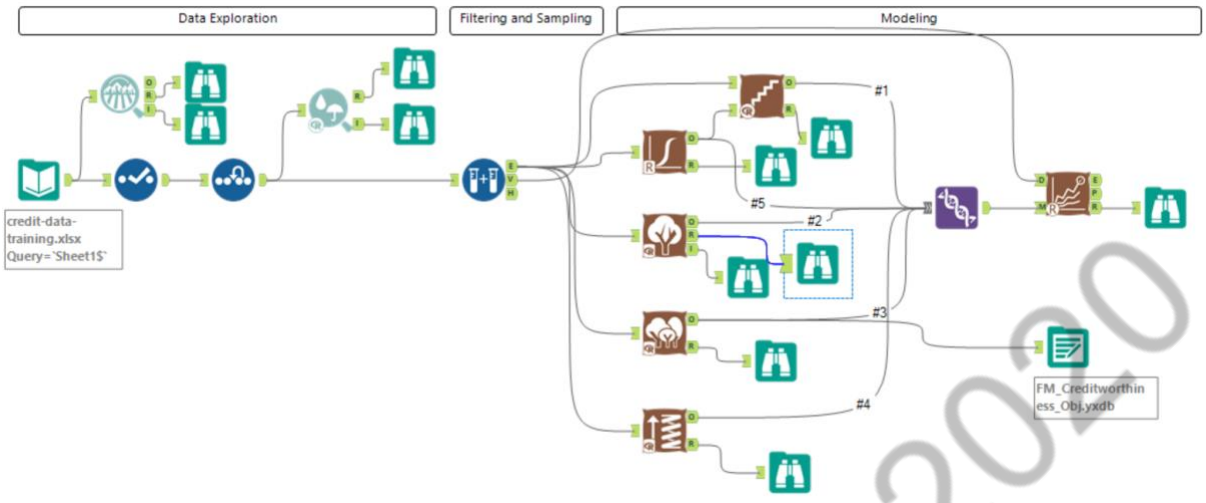


No pair of distinct variables has a correlation higher than 0.7. Therefore, there are no highly correlated variables in this dataset. Therefore, I won't remove any variable in this step.

Step 3: Train your Classification Models

First, I create your Estimation and Validation samples where 70% of your dataset goes to the Estimation set and 30% of the dataset goes to the Validation set.

The following workflow illustrates the building training models process:



1. Logistic Regression_stepwise:

The following table shows coefficients and P-values of predictor variables:

Coefficients:

	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	.

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

(Dispersion parameter for binomial taken to be 1)

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

Number of Fisher Scoring iterations: 5

The most significant variables are **Account Balance**, **Purpose**, **Credit Amount**, **Installment percent**, **Length of current employment.** Those variables have P-value lower than 0.05, which means the corresponding coefficients are statistically significant.

Confusion matrix (from the validation set):

Confusion matrix of LogisticReg_Stepwise

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

Key indicators:

Total Accuracy	76%

True Positive Rate	87.62%
False Positive Rate	51.11%
True Negative Rate	48.89%
False Negative Rate	12.38%

Model is biased towards predicting creditworthy rates because it has significantly higher accuracy in classifying creditworthy 87.6% than predicting non-creditworthy 48.89%. However, this model has a high Type I- error rate (at 51.11%).

2. Decision tree

Model Summary

Variables actually used in tree construction:

[1] Account.Balance Duration.of.Credit.Month Purpose

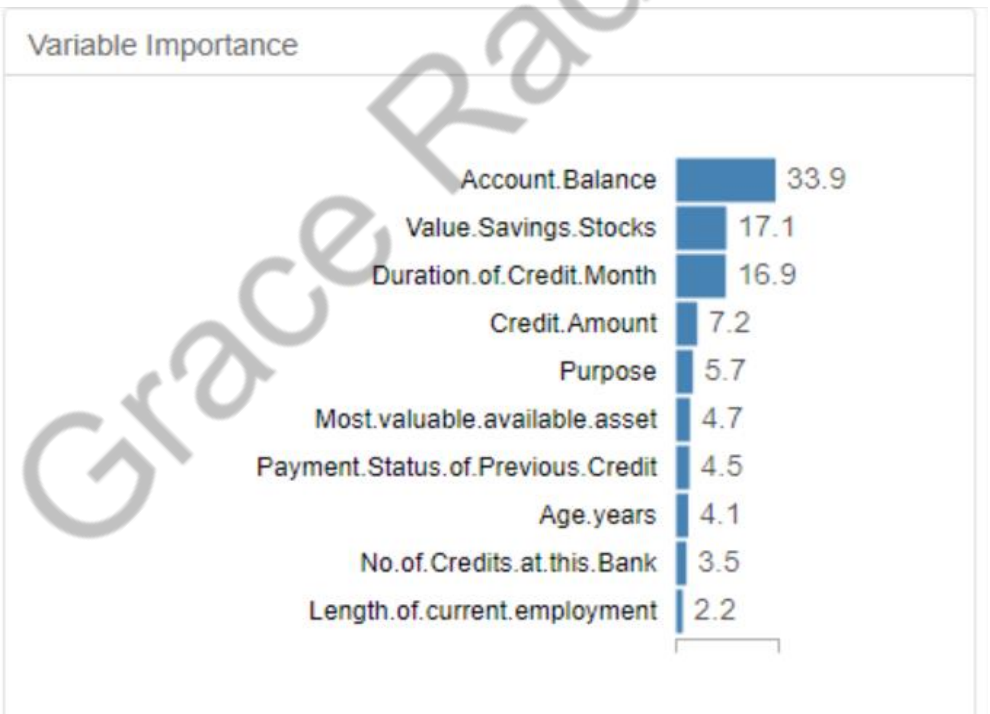
[4] Value.Savings.Stocks

Root node error: 97/350 = 0.27714

n= 350

Pruning Table

Level	CP	Num Splits	Rel Error	X Error	X Std Dev
1	0.068729	0	1.00000	1.00000	0.086326
2	0.041237	3	0.79381	0.94845	0.084898
3	0.025773	4	0.75258	0.88660	0.083032



The most significant variables are **Account Balance**, **Value. Savings. Stocks**, **Duration of Credit Month, and Credit Amount,** having 33.9, 17.1, 16.9, and 7.2 importance point, respectively.

Confusion matrix (from the validation set):

Confusion matrix of DecisionTree

	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	93	26
Predicted_Non-Creditworthy	12	19

Key indicators:

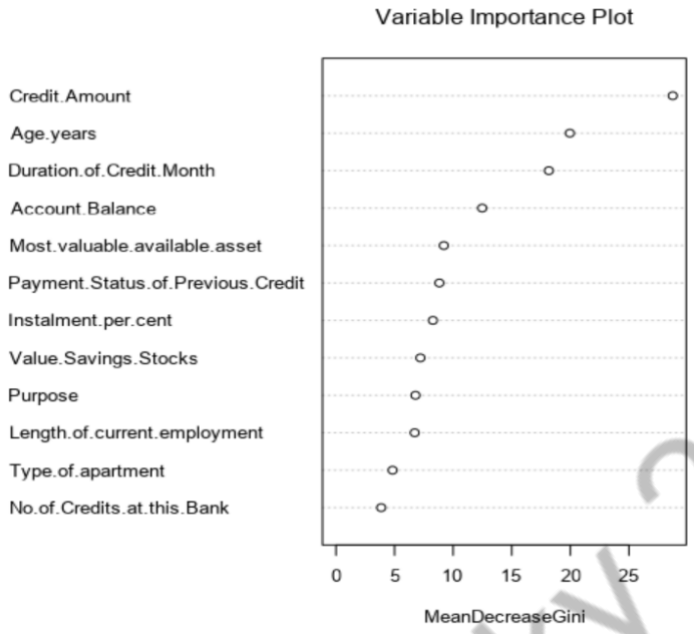
Total accuracy	74.67%
True Positive Rate	88.57%
False Positive Rate	57.78%
True Negative Rate	42.22%
False Negative Rate	11.43%

This model has bias towards predicting creditworthy because it has significantly higher accuracy rate (88.57% vs 42.22%). This also tends to make more type I error than type II as it has higher False Positive Rate. This is risky as the bank may lose money when loaning to customers who are actually not creditworthy.

3. Random forest model

Record	Report
1	*Basic Summary*
2	Call: randomForest(formula = 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, data = the.data, ntree = 500, replace = TRUE)
3	Type of forest: classification Number of trees: 500 Number of variables tried at each split: 3
4	OOB estimate of the error rate: 23.1%
5	Confusion Matrix:

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



The most significant variables are **Credit Amount**, **Age Years**, **Duration of Credit Month, and Account Balance** based on the Mean Decrease Gini measure.

Confusion matrix (from the validation set):

Confusion matrix of ForestModel

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

Key indicators:

Total accuracy	79.33%
True Positive Rate	97.14%
False Positive Rate	62.22%
True Negative Rate	37.78%
False Negative Rate	2.86%

This model is biased towards predicting creditworthy customers more accurately as the true positive rate is almost absolute (at 97.14%). However, this model has the highest false positive rate, at 62.22%.

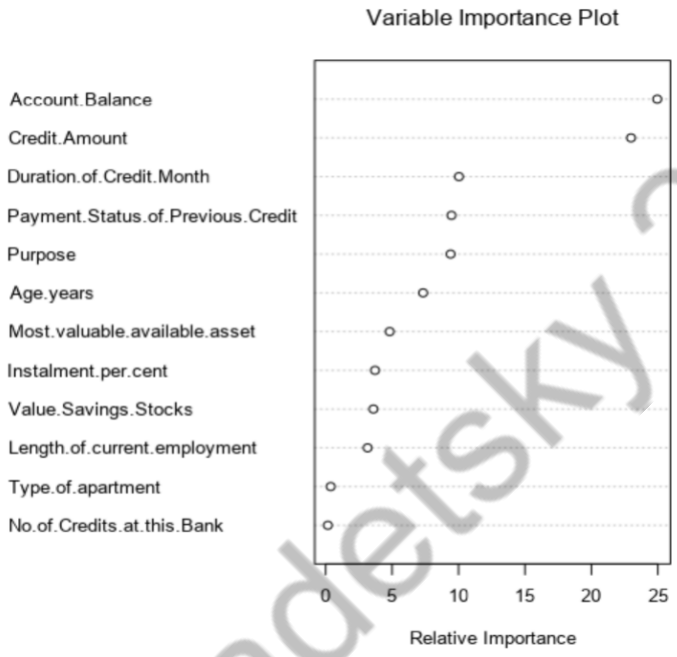
4. Boosted model

Record	Report
1	

Report for Boosted Model BoostedModel

Basic Summary:

Loss function distribution: Bernoulli
Total number of trees used: 4000
Best number of trees based on 5-fold cross validation: 1808



The Variable Importance Plot provides information about the relative importance of each predictor field. The measures are normalized to sum to 100, and the value for each field gives the relative percentage importance of that field to the overall model.

The most significant variables are **Account Balance**, Credit Amount, **Duration of Credit Month, Payment Status of Previous Credit** based on relative importance scores.

Confusion matrix (from the validation set):

Confusion matrix of BoostedModel

	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	101	28
Predicted_Non-Creditworthy	4	17

Key Indicators:

Total accuracy	78.67%
True Positive Rate	96.19%
False Positive Rate	62.22%

True Negative Rate	37.78%
False Negative Rate	3.81%%

This model is also biased towards Creditworthy as it tends to predict the creditworthy outcome more accurately than non-creditworthy (96.19% vs 37.78%). This model has the highest chance of making type I error as the False positive rate is 62.22%.

Step 4: Writeup

Model Comparison Report

Fit and error measures

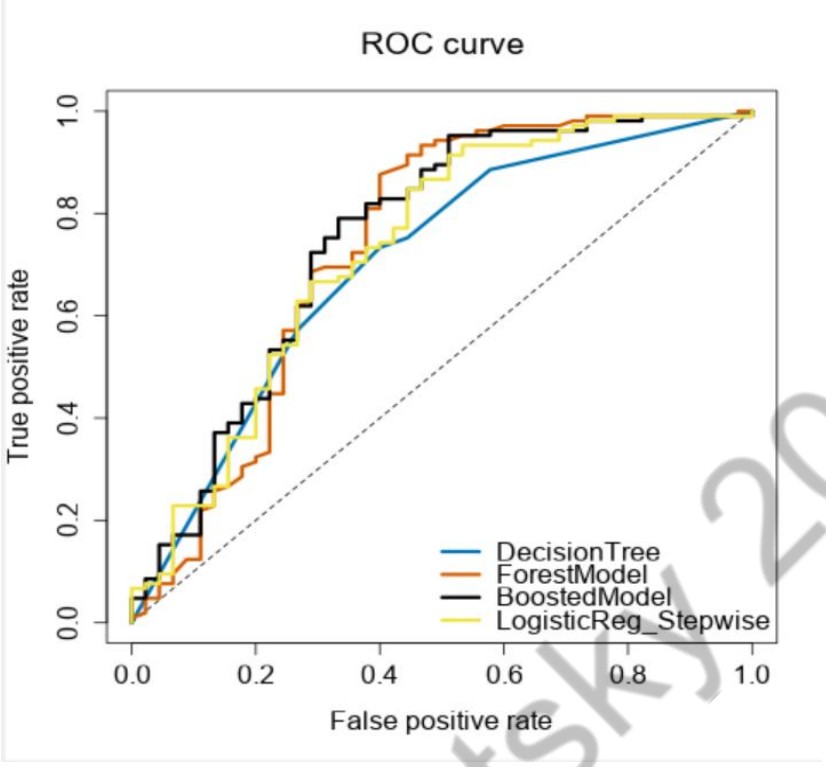
Model	Accuracy	F1	AUC	Accuracy_Creditworthy	Accuracy_Non-Creditworthy
DecisionTree	0.7467	0.8304	0.7035	0.8857	0.4222
ForestModel	0.7933	0.8681	0.7368	0.9714	0.3778
BoostedModel	0.7867	0.8632	0.7515	0.9619	0.3778
LogisticReg_Stepwise	0.7600	0.8364	0.7306	0.8762	0.4889

While accepting that all four models have their biases, I will choose overall accuracy rate and F1 score as the key deciding factors since the manager cares about general accurate classification than avoiding certain errors such as risking default when loaning to unqualified applicants.

From the above report, the **Forest Model** has the best performance with the highest overall accuracy rate when being compared against the validation data set (at 79.33%). This model also has the highest F1 score (at 86.81%), which means this model has the best balance between precision (exactness) and recall (completeness).

The **Forest Model** has highest accuracy within the creditworthy class (at 97.14%). The accuracy in predicting a true noncreditworthy applicant is subpar (at 37.78%) but not too far off from other models.

The ROC Curve shows that the Forest Model has the second highest Area Under the Curve (AUC) (at 73.68) and the curve is pulled more towards the top left corner. This means it has the second most optimal tradeoff between true positive rate and false positive rate (type I error). However, AUC is not deciding factor in this case because the data set is imbalanced (having more creditworthy values) and overall classification accuracy is more prioritized.



After applying the **Forest Model** to the list of 500 applicants, I identified **408** creditworthy applicants