Project: Creditworthiness

Step 1: Business and Data Understanding

The manager of a bank has requested a classification model to systematically evaluate the creditworthiness of new loan applicants. To build the model, data on previous applications and their creditworthiness is required. Specifically, twelve features (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", "Type-of-apartment", "No-of-Credits-at-this-Bank", "Age_years") will be required for both previous applicants and new applicants along with a target class ("Credit-Application-Result") for the previous applicants. A binary classification model will then be utilised to classify each loan application as 'Creditworthy' or 'Non-Creditworthy'.

Step 2: Building the Training Set

Features with a high proportion of missing values ("Duration in current address") and zero variance ("Concurrent Credits", "Occupation") were removed. The features "Foreign Worker", "Guarantors", "No of dependents" were also removed as despite these features having two unique values, they had low variability (Fig 1). Additionally, "Telephone" was removed due to its irrelevance to this business problem. Finally, missing values in the "Age" feature were imputed with the median age. The median was chosen as it is an average measure robust to outliers and the "Age" feature is positively skewed. No continuous variable had a correlation greater than 0.7 (Fig 2).

Fig (1) Histograms of the features.

	Duration.of.Credit.Month	Credit.Amount	Instalment.per.cent	Most.valuable.available.asset	Age.years	Type.of.apartment
Duration.of.Credit.Month	1.0000000	0.5704408	0.0795146	0.3047342	-0.0663189	0.1531405
Credit.Amount	0.5704408	1.0000000	-0.2856309	0.3277621	0.0686430	0.1686831
Instalment.per.cent	0.0795146	-0.2856309	1.0000000	0.0781104	0.0405397	0.0829360
Most.valuable.available.asset	0.3047342	0.3277621	0.0781104	1.0000000	0.0854367	0.3796504
Age.years	-0.0663189	0.0686430	0.0405397	0.0854367	1.0000000	0.3330748
Type.of.apartment	0.1531405	0.1686831	0.0829360	0.3796504	0.3330748	1.0000000
No.of.dependents	-0.0604413	0.0055003	-0.1164661	0.0507817	0.1177351	0.1707221
	No.of.dependents					
Duration.of.Credit.Month	-0.0604413					
Credit.Amount	0.0055003					
Instalment.per.cent	-0.1164661					
Most.valuable.available.asset	0.0507817					
Age.years	0.1177351					
Type.of.apartment	0.1707221					
No.of.dependents	1.0000000					

Fig (2) Correlation matrix for continuous variables

A training data set was generated with twelve features, a target class and five hundred observations.

Step 3: Train your Classification Models

Data was randomly split into *Estimation (70%) and Validation (30%) samples and four models were trained (Logistic Regression, Decision Tree, Forest Model, Boosted Model).*

Logistic Regression

Using an 'Enter' method, all 12 features were utilised in the logit logistic regression model. On the estimation dataset, the generated model had a MacFadden's R^2 of 0.21 and had eight significant features excluding the intercept (fig 3).

	Estimate	Std. Error	z value	Pr(>|z|)	
(Intercept)	-3.0136120	1.013e+00	-2.9760	0.00292	**
Account.BalanceSome Balance	-1.5433699	3.232e-01	-4.7752	1.79e-06	***
Duration.of.Credit.Month	0.0064973	1.371e-02	0.4738	0.63565	
Payment.Status.of.Previous.CreditPaid Up	0.4054309	3.841e-01	1.0554	0.29124	
Payment.Status.of.Previous.CreditSome Problems	1.2607175	5.335e-01	2.3632	0.01812	*
PurposeNew car	-1.7541034	6.276e-01	-2.7951	0.00519	**
PurposeOther	-0.3191177	8.342e-01	-0.3825	0.70206	
PurposeUsed car	-0.7839554	4.124e-01	-1.9008	0.05733	.
Credit.Amount	0.0001764	6.838e-05	2.5798	0.00989	**
Value.Savings.StocksNone	0.6074082	5.100e-01	1.1911	0.23361	
Value.Savings.Stocks£100-£1000	0.1694433	5.649e-01	0.3000	0.7642	
Length.of.current.employment4-7 yrs	0.5224158	4.930e-01	1.0596	0.28934	
Length.of.current.employment< 1yr	0.7779492	3.956e-01	1.9664	0.04925	*
Instalment.per.cent	0.3109833	1.399e-01	2.2232	0.0262	*
Most.valuable.available.asset	0.3258706	1.556e-01	2.0945	0.03621	*
Type.of.apartment	-0.2603038	2.956e-01	-0.8805	0.3786	
No.of.Credits.at.this.BankMore than 1	0.3619545	3.815e-01	0.9487	0.34275	
Age_years	-0.0141206	1.535e-02	-0.9202	0.35747	

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

Fig (3) Logistic Regression Coefficients and p values.

On the validation data set, the Logistic Regression method achieved a classification accuracy of 78%. Examination of the Confusion matrix for this model illustrates that the model was no better than guessing class for those who were non-creditworthy (fig 4) and so was biased to predicting creditworthy applicants.

Confusion matrix of LogReg

	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	95	23
Predicted_Non-Creditworthy	10	22

Fig (4) Confusion Matrix of the Logistic Regression

Decision Tree

The decision tree contained eight features. Using the Gini impurity values, the three most important features were “Account Balance”, “Value.Savings.Stocks” and “Duration of Credit”(fig 5).

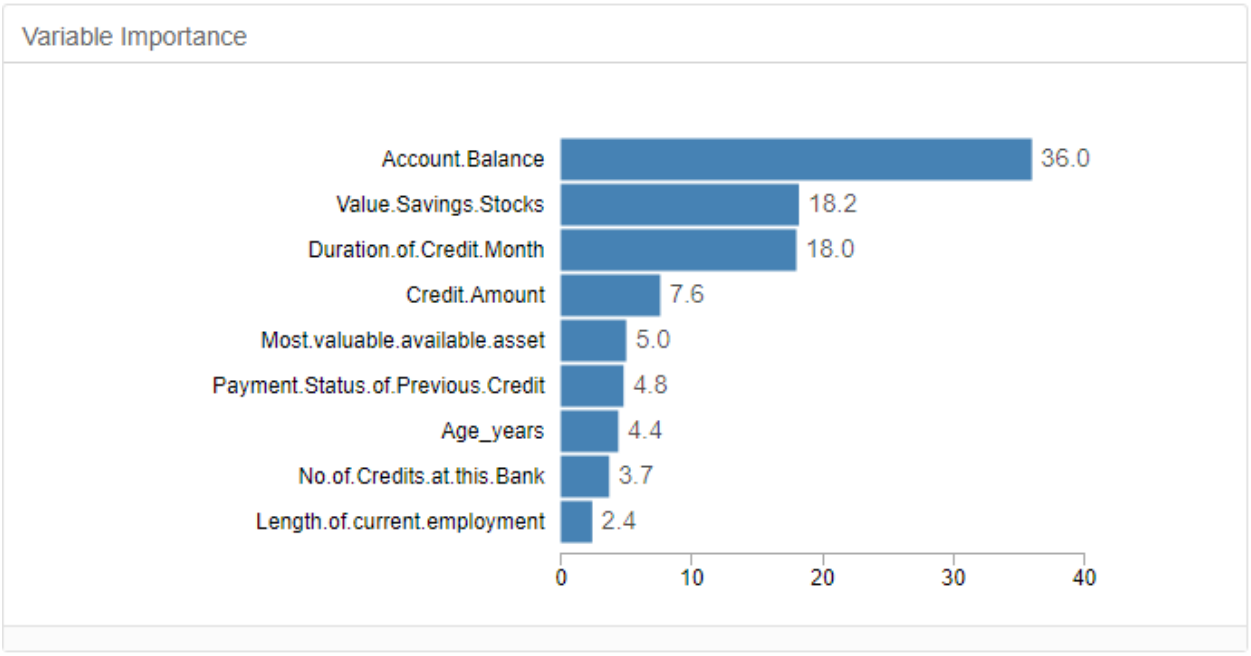

Fig (5) Variable Importance for the Decision Tree based on Gini impurity values.

The overall accuracy of this model was 74.67%. Examining the confusion matrix for this model demonstrated the decision tree was again poor at classifying those who are not creditworthy [it is biased to classifying creditworthy applicants] (fig 6).

Confusion matrix of Tree

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

Fig (6) Confusion Matrix for the Decision Tree

Forest Model

A forest model containing 500 trees was generated with three variables tested at each split. This achieved an out of bag error rate of 24%. Examining the mean decrease Gini value illustrates that the tree most important features were Credit amount, Age and Duration of Credit (Fig 7).

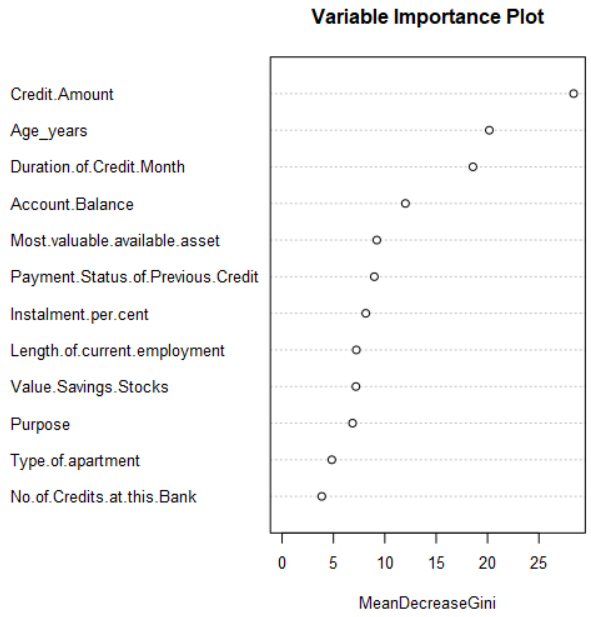

Fig (7) Variable Importance for the forest model based on the mean Gini decrease.

The forest model achieved a classification accuracy of 80% on the valuation dataset. The confusion matrix for this model illustrates that the improved accuracy is due to a

high creditworthy classification bias with 96.2 % correctly classified as creditworthy and only 42% correctly classified as not creditworthy (Fig 8). This model is biased to predicting creditworthy applicants.

Confusion matrix of Tree_Forest

	Actual_Creditworthy	Actual_Non-Creditworthy
Predicted_Creditworthy	101	26
Predicted_Non-Creditworthy	4	19

Fig (8) Confusion Matrix for the Forest Model.

Boosted Model

A gradient boosted model was generated using an ensemble of 4000 trees and the Bernoulli loss function. Examining the variance impotence plot suggests that two variables, namely account balance and credit amount were the most important to this model (Fig 9).

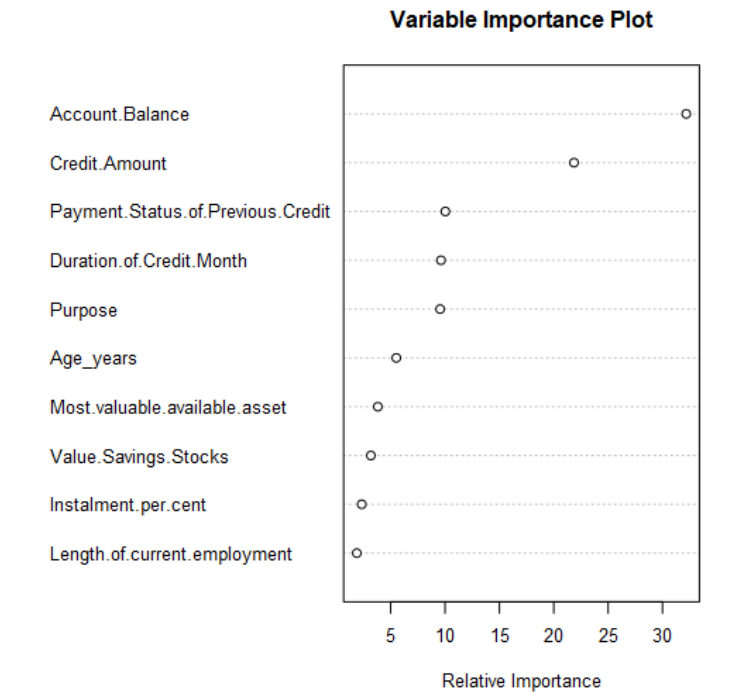

Fig (9) Variable importance for the boosted model.

On the validation dataset, the boosted model achieved an accuracy of 78.7 %. The confusion matrix for this model demonstrated that the boosted model performed the worst of the models at correctly identifying non-credit worthy customers (37.3%) while achieved high prediction accuracy for those who were creditworthy (96.2%). As such this model was again biased to predicting creditworthy applicants. (Fig 10)

Confusion matrix of Tree_Boost

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

Fig (10) Confusion Matrix for the Boosted Model

Step 4: Writeup

The purpose of this report was to create a classification model to systematically evaluate the creditworthiness of new loan applicants. To do so, four standard models were compared in Alteryx, namely; Logistic Regression, Decision Tree, Forest Model and Boosted Model.

All the examined models demonstrated high classification accuracy (74.7 – 80%). However, they also demonstrated a high bias to correctly predicting applicants who were creditworthy. Indeed, for all models, the accuracy of correctly classifying application who were not creditworthy was worse than guessing at random (37.7 – 48.9 %) while correctly classifying applications as creditworthy was high (86.7 -96.2%). This

is likely due to high class imbalance observed in the dataset and future reports should look to oversampling/under sampling methods.

It is unknow if either classification specificity or sensitivity are more important for this business problem. As such this report primarily focused on overall classification accuracy. While the receiver operator curve suggests that the boosted model is the best performer, the forest model also performed well (Fig 11). Furthermore, this report utilises a standard 0.5 probability threshold for classification. As such the Forest model was chosen given its high overall classification accuracy (80%) and higher specificity in comparison to the boosted model (42.2 vs 37.8%). The chosen forest model classified 408 of the new applications as creditworthy. Full Alteryx workflow is illustrated in Fig 12.

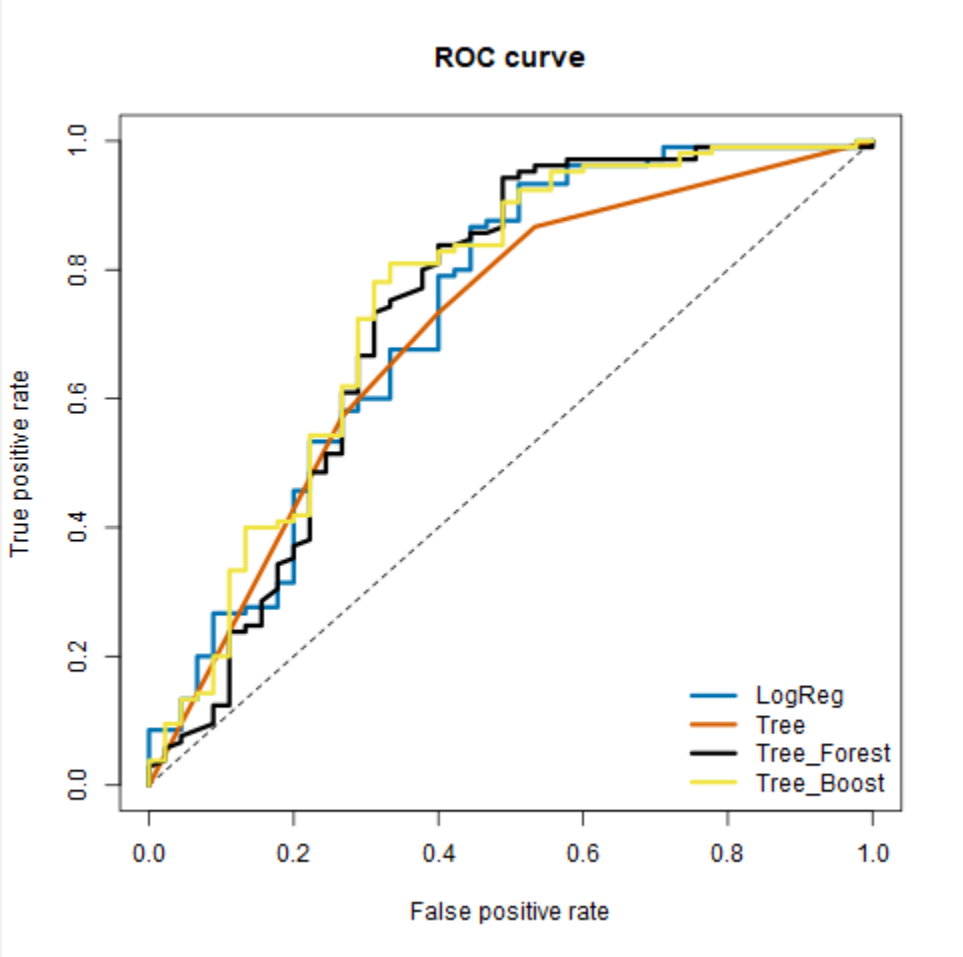

Fig (11) Receiver Operator Curve for the four models compared.

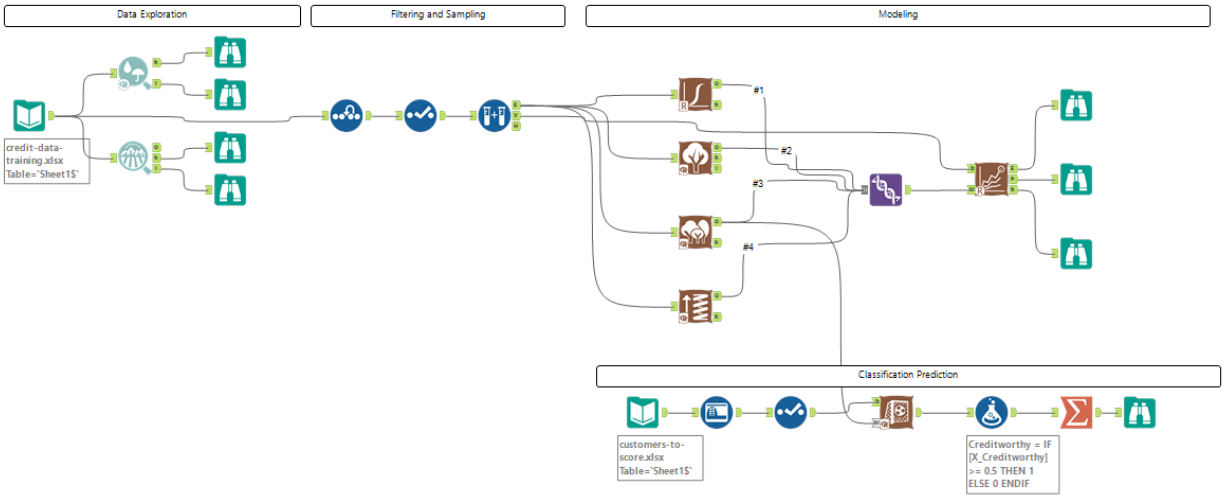

Fig (12). Alteryx Workflow