John Ryan Kivela, MA
UMass Boston
College of Management
MSIS - 672 Final Project
5/5/2024

Introduction

We are tasked with using potential customer information to predict Credit Mix. Let's suppose we work for a local philanthropic investment group that is trying to decide which new project to invest in. We want to use information we have collected about a company to predict the quality of their credit, and by extension the risk of investing in their product. By using this information to inform investment decisions, the company makes sound decisions about which project to support given financial viability and increases its likelihood of success overall. Ultimately, we found that the Random Forest Classifier yielded the best predictive performance, providing valid predictions 98% of the time.

To do this we considered 4 machine learning models:

- Classification Trees
  - Decision Tree Classifier
  - Random Forest Classifier
  - Gradient Boosted Classifier
- Neural Networks
  - Multi-layer Perceptron Classifier

The Target Variable is Credit Mix, which has 3 classes: Good, Bad, and Standard.

Data Preprocessing

We manipulated the data in many ways. First, we converted data types and identified variables that would not be needed for the assessment. Next, we addressed missing data by inputing sensible values. Finally, we used one hot encoding to encode categorical data so that it can be assessed by decision trees and neural networks.

Here is a summary:

1. The output_file data set contains 40,195 observations of 25 variables.
2. Several variables were excluded from the test set:

Feature	Reason for exclusion
'Credit_History_Age'	Assignment parameters
'Type_of_Loan'	Assignment parameters
'ID'	Not useful in this context
'Customer_ID'	Not useful in this context
'Month'	Not useful in this context

3. Accounted for missing or null values through conversion and imputation.

Feature with Null Values	Data Preparation Activity
'Occupation'	Replace NaN with "Unknown"
'Monthly_Inhand_Salary'	Impute
'Type_of_Loan'	Drop variable.
'Num_of_Delayed_Payment'	Impute
'Changed_Credit_Limit'	Impute

'Num_Credit_Inquiries'	Impute
'Credit_History_Age'	Convert and impute
'Amount_invested_monthly'	Impute
'Payment_Behaviour'	Drop na
'Monthly_Balance'	Convert and impute

4. One hot encoded Payment of Minimum Amount.

Analysis

Four separate models were designed to find the best predictor of credit mixes so that we know which people or companies to lend money to:

- Decision Tree Classifier
- Random Forest Classifier
- Gradient Boosted Classifier
- Multi-layer Perceptron

The following sections outline the results of each model.

Decision Tree Classifier

A decision tree creates logical rules to separate a population into subgroups based on the relative importance of a set of features. In this case, customer information predicts Credit Mix.

Results

Decision Tree Classifier Assessment		
Exhaustive Grid Search	**Setting / Result**	**Comments**
Hyperparameters:		
Criterion	Entropy	Quality of split
Max Depth	8	Maximum number of splits
Minimum Impurity Decrease	.00001	
Minimum Sample Split	200	Cases needed to make a split
This set of parameters yields a best score of 0.917. Overfitting is mitigated by having a higher min sample split and reducing the number of splits.		
Cross Validation Results		
Accuracy Range	0.900 to 0.912	
Mean Accuracy	0.907	
Assessment	Accuracy is stable	
Classification Report		
Accuracy	0.923 (Moderately high)	
Precision (True Positive)	0.94	Highest for Bad and Good (0.94)
Recall (Sensitivity to True Positive)	0.93	Highest for Standard (0.93)

Support	Highest = Standard	Lowest = Bad
Importance (Top 3 Features)		
Non Payment of Min Amount		
Outstanding Debt		
Interest Rates		

Summary

The accuracy is moderately high at 92% across the validation dataset! The most important features in this model were Payment of Minimum Amount, and Outstanding debt (Figure 1). The model was strongest at predicting for Good or Bad, and Standard had higher sensitivity to changes in the data set. The model also produces balanced metrics (F1-Score = .93). Finaly, Efforts were taken to optimize depth, impurity, and sample split to counteract overfitting, reducing the overall complexity of the model.

Random Forest Classifier

A random forest classifier is an ensemble of decision trees, wherein each tree assesses a different random selection of data. The results of the ensemble are then considered together to generate a prediction to the investor, in this case, Credit Mix.

Results

Random Forest Classifier Assessment		
Exhaustive Grid Search	**Setting / Result**	**Comments**
Hyperparameters:		
N Estimators	300	Number of trees
Criterion	Entropy	Quality of Split
Max Depth	15	Number of Splits
Minimum Impurity Decrease	.00001	
Minimum Sample Split	2	Number needed to Split
* This set of parameters yields a best score of 0.94. * *Overfitting is mitigated by using the ensemble random forest.		
Cross Validation Results		
Accuracy Range	0.930 to 0.939	
Mean Accuracy	0.934	
Assessment	Accuracy is stable	
Classification Report		
Accuracy	0.98 (Very high)	
Precision (True Positive)	0.99	Highest for Standard (0.99)
Recall (Sensitivity to True Positive)	1.00	Highest for Good (1.00)
Support	Highest = Standard	Lowest = Bad
Importance (Top 3 Features)		
Interest Rate		
Outstanding Debt		

Number of Delayed Payments		

Summary

The accuracy is very high at .98 across the validation dataset! The most important features in this model were Interest Rate, Outstanding Debt, Number of Delayed Payments, and Number of Bank Accounts (Figure 3). The model was marginally stronger at predicting standard, but by only .01. The model also produces balanced metrics (F1-Score = .98). Efforts were taken to optimize depth, impurity, and sample split to counteract overfitting.

Gradient Boosting Classification

A boosted tree classification also creates an ensemble of trees; however, a gradient boosting classifier, instead of creating many random trees, creates a sequence of trees that work to reduce the error of the previous tree in its group.

Results

Gradient Boosting Classifier Assessment		
Exhaustive Grid Search	**Setting / Result**	**Comments**
Hyperparameters:		
N Estimators	75	Number of trees
Max Depth	4	Number of splits
Minimum Impurity Decrease	.1	
Minimum Sample Split	100	Number required to split
* This set of parameters yields a best score of 0.937. This is an increase from the default model (accuracy = .914). * Overfitting is mitigated by using the gradient boosting.		
Cross Validation Results		
Accuracy Range	0.95	
Mean Accuracy	0.95	
Assessment	Accuracy is relatively stable	
Classification Report		
Accuracy	0.95 (Very high)	
Precision (True Positive)	0.96	Highest for Bad (0.96)
Recall (Sensitivity to True Positive)	0.97	Highest for Bad (0.97)
Support	Highest = Standard	Lowest = Bad
Importance (Top 3 Features)		
Outstanding Debt		
Non Payment of Min Amount		
Interest Rate		

Summary

The accuracy is high at .937 across the validation dataset! The most important features in this model were Outstanding Debt, Non-payment of Minimum Amount, Interest Rate, And Number of Delayed Payments. The model was marginally stronger at predicting Bad, but by only .01 (Figure

5). The model also produces balanced metrics (F1-Score = .95). In addition, efforts were taken to optimize depth, impurity, and sample split to counteract overfitting. Considerations also needed to be made regarding computing power. With more sophisticated hardware, better results may be possible.

Multilayered Perceptron Classifier

A Multilayered Perceptron Classifier is a feedforward neural network. In this case, our dataset of customer information is fed to the model. The perceptron (or neuron) takes in these inputs, applies weights and biases, and iterates through backpropagation to predict Credit Mix.

Results

Multi-layer Perceptron Classifier Assessment		
Exhaustive Grid Search	**Setting / Result**	**Comments**
Hyperparameters:		
Hidden Layer Size	10,	Number of nodes in a neuron
Activation Function	ReLu	
Solver	Adam	
Learning Rate	Constant	
This set of parameters yields a best score of 0.643. This is an increase from the default model (accuracy = 0.575). Overfitting is mitigated by using the gradient boosting.		
Cross Validation Results		
Accuracy Range	0.556 to 0.693	
Mean Accuracy	0.652	Loss to 0.592 on valid data indicates overfitting
Assessment	Accuracy is moderately low,	
Classification Report		
Accuracy	0.642	
Precision (True Positive)	0.727	Highest for Bad (0.96)
Recall (Sensitivity to True Positive)	0.642	Highest for Bad (0.97)
F-1	0.587	Lowest = Bad

Summary

The overall accuracy of the multilayer perceptron model is 0.664. Through optimization of hyperparameters, the model was improved from the default (0.575). The results for precision and accuracy also indicate model stability around 0.67. While the accuracy for this model is not overly high, it is stable. It should be noted that the computational expense for going deeper with this model was too great for the hardware I have at hand. Greater performance might be attainable with more sophisticated equipment

Conclusions and Recommendations

Our highest performing model was the Random Forest Classifier. It had an predictive accuracy level of 98%. The model had excellent precision and recall. Other performance metrics, like F-1, also indicate a high level of stability for this model. Interest Rate, Outstanding Debt, Number of Delayed Payments, and Number of Bank Accounts were the most important features out of the original 18 variables.

Given these results, it recommended that the investor adopt the Random Forrest Classification Model. In addition, when considering investment into a new company or individual, the customer should pay particular attention to the important features listed above.

References

Galit Shmueli, Peter C. Bruce, Peter Gedeck and Nitin R. Patel. Data Mining for Business Analytics: Concepts, Techniques and Applications in Python, 1st edition Wiley, 2019, ISBN: 978-1-119-54986-4

Appendix

Figure 1: Decision Tree Classifier Importance Plot

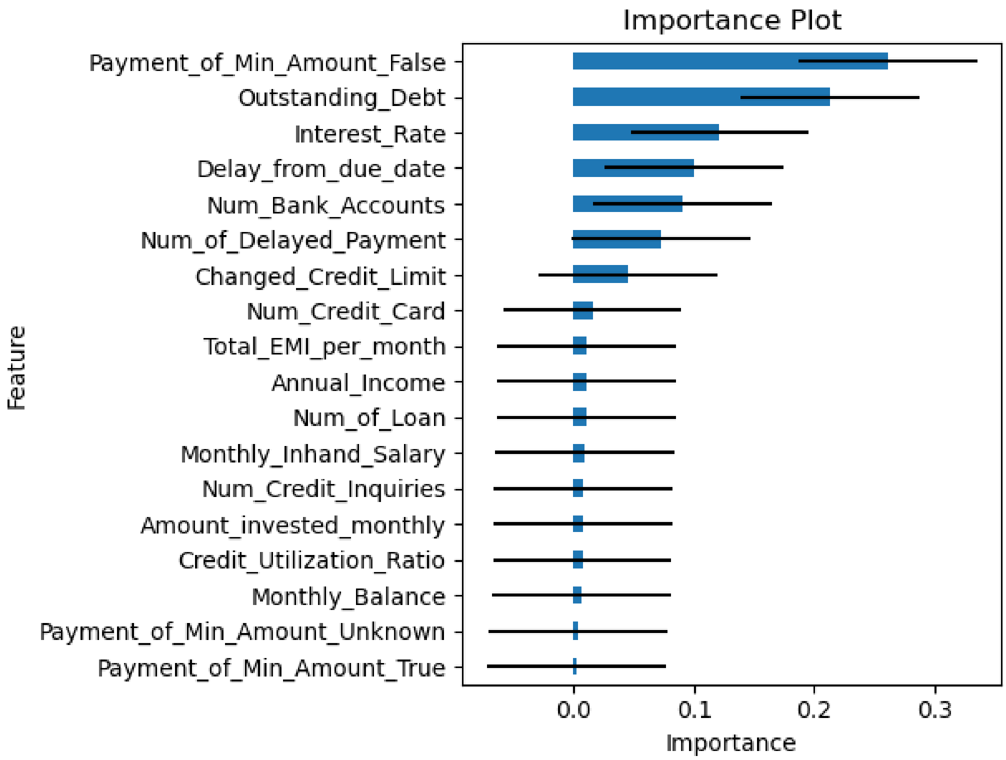


Figure 2: Decision Tree Classifier: Tree diagram

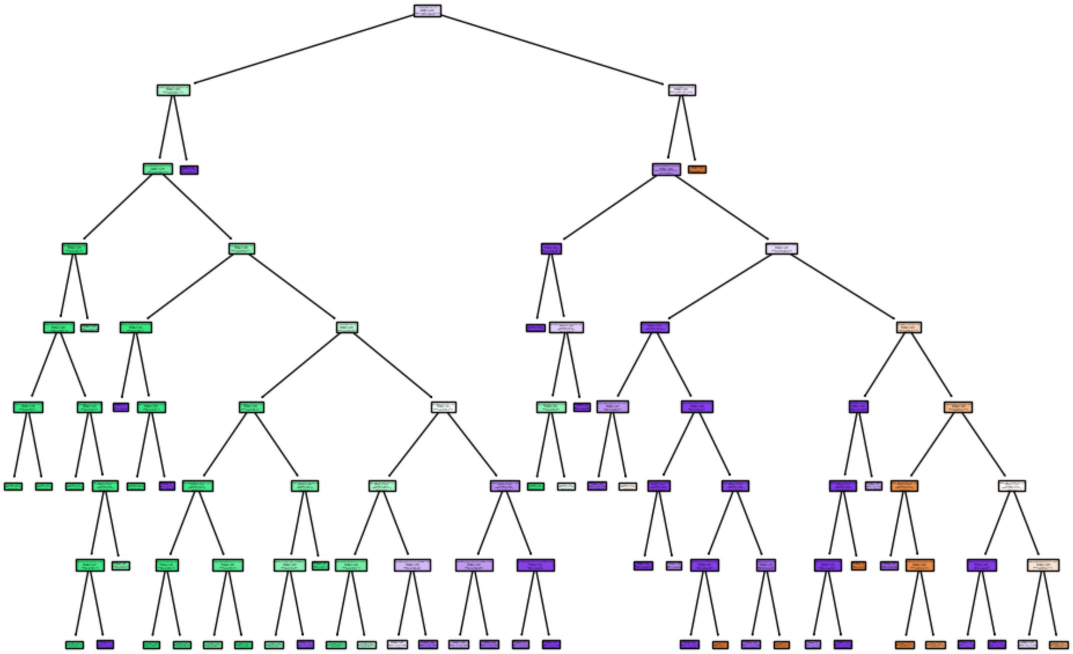


Figure 3: Random Forest Classifier: Importance Plot

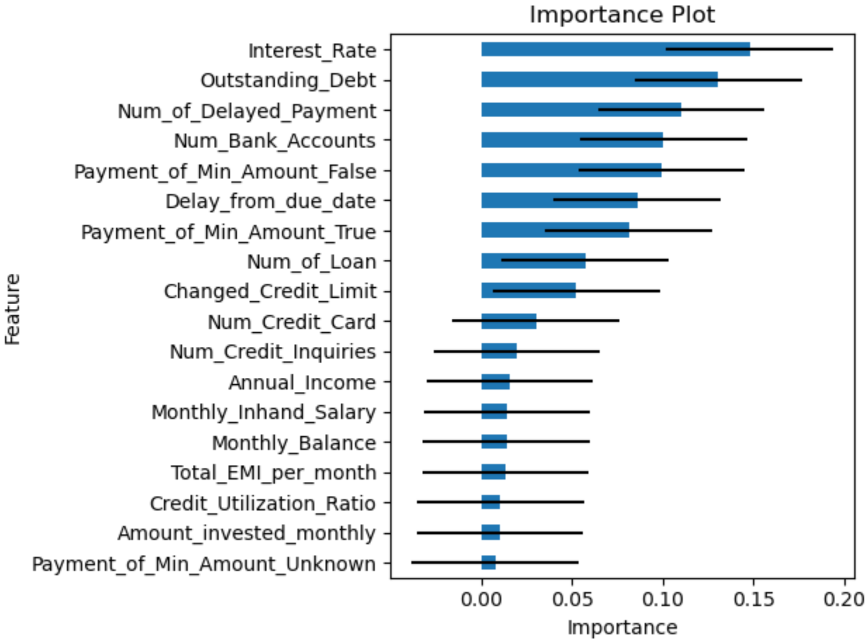


Figure 4: Random Forest Classifier; Confusion Matrix

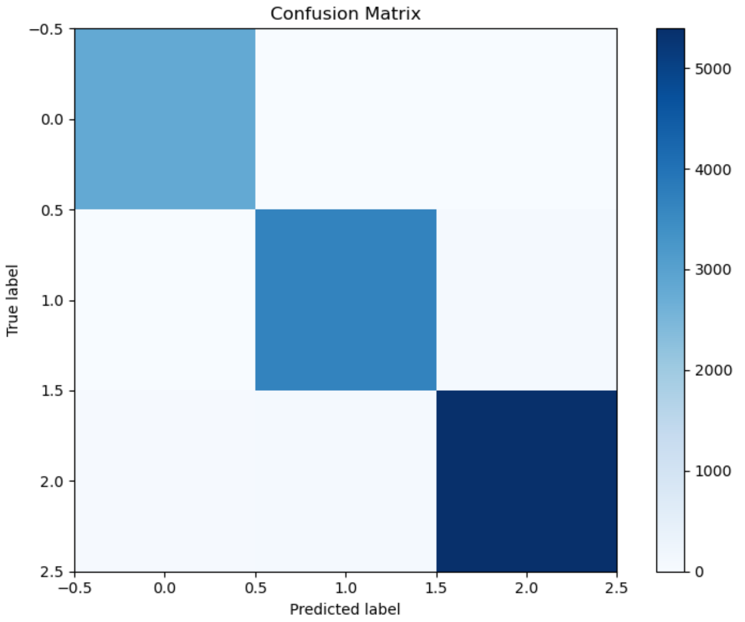
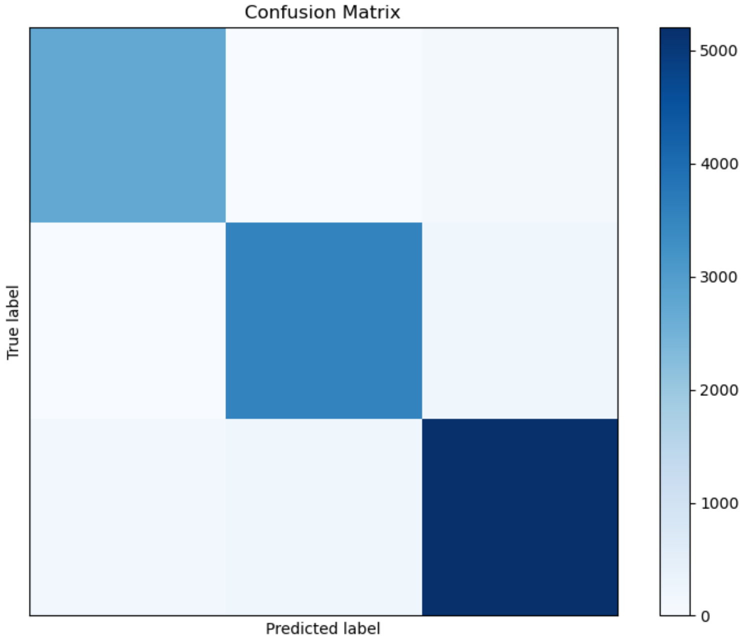


Figure 5: Gradient Boosting Classifier: Confusion Matrix


Confusion Matrix
True label
Predicted label