Final Project – Analyses of Lending Practices

I. Linear Classification

(Q2.2.1) Linear classifier – comparisons

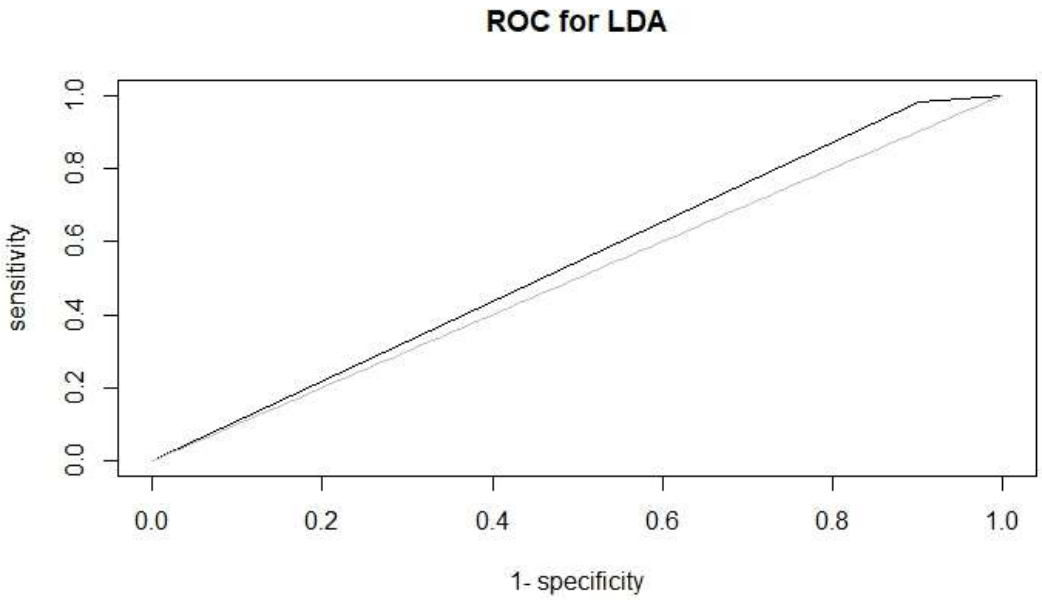
Scenario – 1

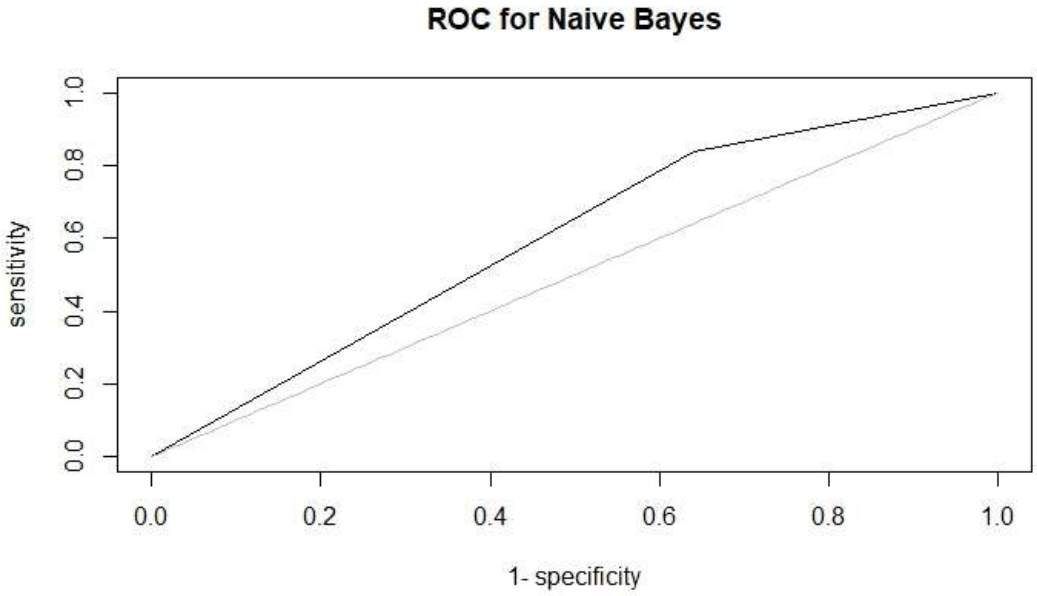
	ACCURACY	SPECIFICITY
LDA	**78.77%**	**12.60%**
LOGISTIC	**78.81%**	**9.70%**
NAÏVE BAYES	**73.15%**	**36%**

Scenario – 2

	ACCURACY	SPECIFICITY
LDA	**78.79%**	**12.66%**
LOGISTIC	**78.87%**	**9.83%**
NAÏVE BAYES	**73.15%**	**36%**

From the table above, in both the scenarios LDA and Logistic give similar accuracy rate but LDA performs better on the specificity criteria, i.e. the classification of denied applications. We consider specificity criterion because the dataset is skewed and there are a lot more applications accepted than denied (around 20 %). Across both scenarios Naïve Bayes performed well on the specificity criterion but suffered from poor accuracy. To compare between better performing linear classifier among LDA and Naïve Bayes we plot ROC curves and compare their AUC. Overall, the scenario 2, which included all variables, performed slightly better than scenario 1 that excluded two variables – Applicant Income and Loan Amount.





LDA AUC - 0.55

Naïve Bayes AUC – 0.60

From the ROC curves above and the AUC values we conclude that Naïve Bayes performs best among three linear classifiers.

II. Non-Linear Classification

(Q2.2.7) Non-Linear classifiers – comparisons

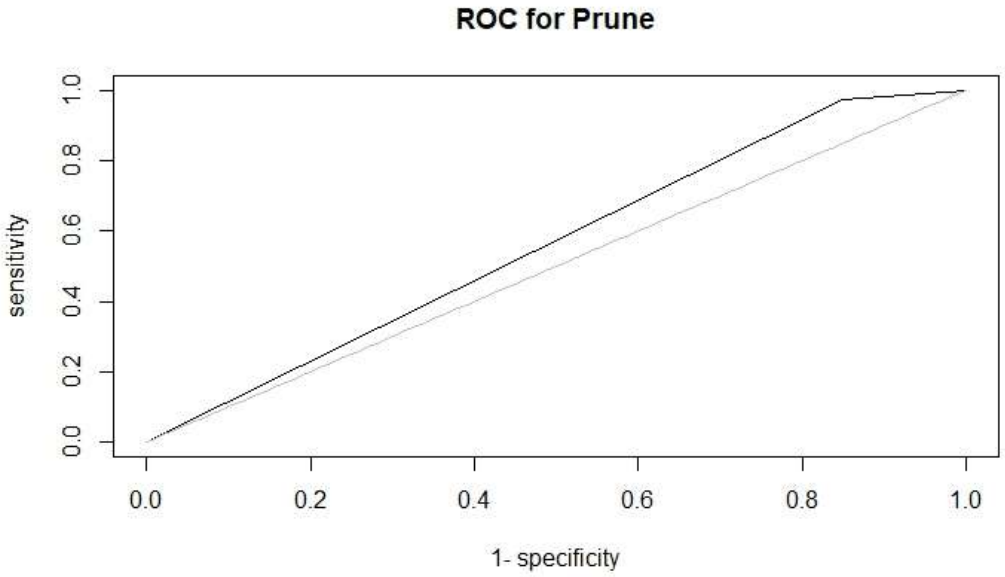
Scenario 1

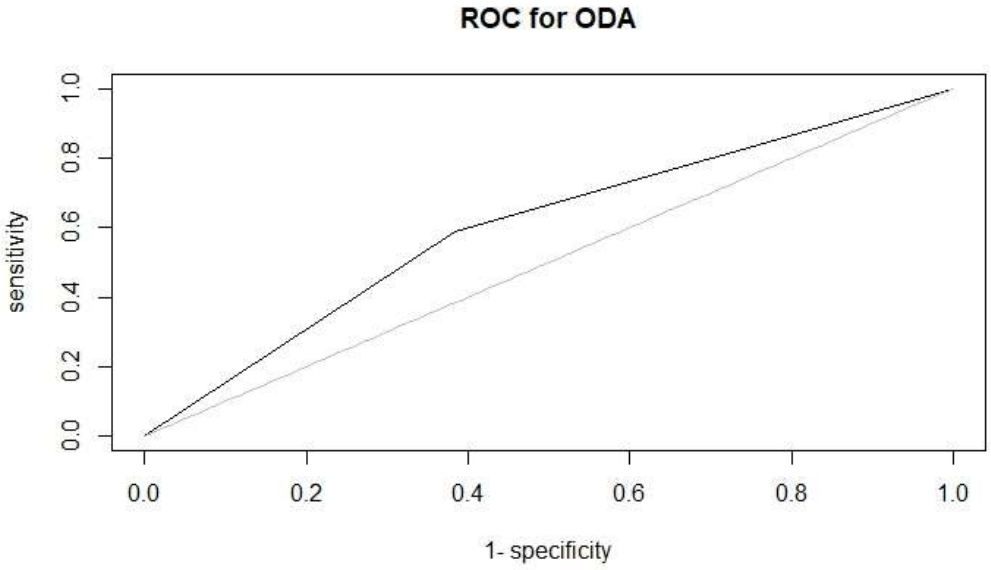
	ACCURACY	SPECIFICITY
QDA	59.35%	61.67%
DECISION TREE (PRUNED)	79.41%	15%
KNN	75.12%	11.07%

Scenario 2

	ACCURACY	SPECIFICITY
QDA	55.68%	67.91%
DECISION TREE (PRUNED)	79.41%	15.18%
KNN	74.92%	12.33%

From the table above, Decision Trees and KNN have better accuracy rate than QDA, but QDA manages to predict the denied applications much better than any model considered so far. The Pruned Decision Trees are preferred overall because they have high accuracy and AUC. For non-linear models, scenario 1 seems to be marginally better. This may be due to overfitting because of addition of two variables in scenario 2.





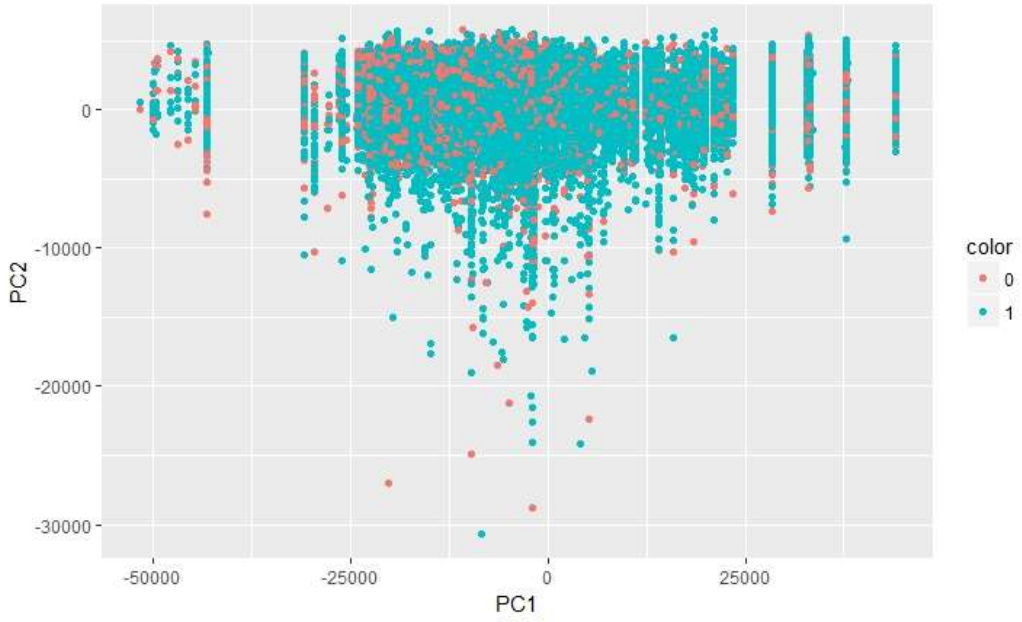
The ROC curve for Pruned Decision Trees and QDA is shown in the graphs above.

QDA AUC – 0.60

Decision AUC – 0.55

The AUC values and accuracy values give conflicting measures in this case. Though the AUC for QDA is better than decision trees, because of the extremely low accuracy rate we prefer Decision Trees over QDA.

From our experiments, the data doesn't seem to be linearly separable as the linear models don't have high accuracy rates and very low specificity which suggests the models are predicting the denied applications poorly. Contrast this with QDA, though the model has low accuracy, is better at predicting the denied applications. To check this we perform PCA on the explanatory variables with two principle components.



From the figure above, the data doesn't have a linear separation with respect to the two principal components which seems to validate our previous conjecture.

III. Evaluation of more complex methods

Classifier performance on train

Below we state the hyper-parameter values that resulted in the lowest train error for 4 of complex models trained.

1. Random Forest – mtry = 6
2. SVM – Cost = 10, Gamma = 4
3. Lasso – Lambda = 0.0005
4. Ridge—Lambda = 0.001

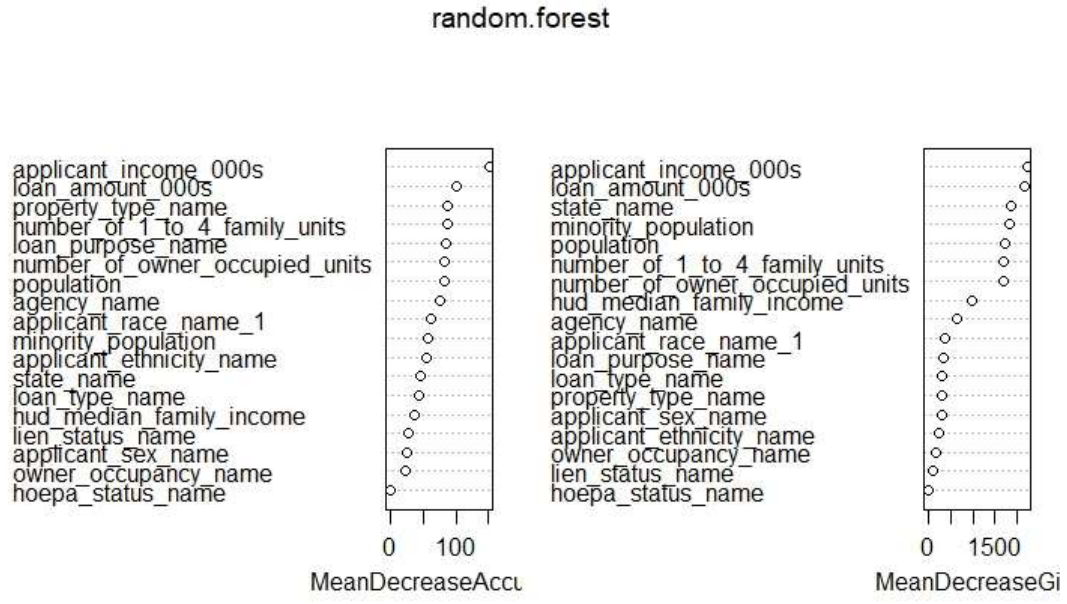
Classifier performance on test

	ACCURACY	SPECIFICITY
RANDOM FOREST	**79.54%**	**16.17%**
BOOSTING	**78.43%**	**6.8%**
SVM	**78.87%**	**1.7%**
LASSO	**78.81%**	**9.30%**
RIDGE	**78.82%**	**9.63%**

All the five models had similar accuracy rates, but Random Forest seemed to perform the best not only with regards to accuracy but also considering the specificity of predictions.

Below we tabulate the features based on their importance in measuring the variance of responses as observed during Random Forest.

Important Features



Overall Summary

4.1. Tree based methods like Decision Trees and Random Forest performed better than all other models considered. They not only resulted in a high accuracy rate but also better predicted the rejections of applications. Random Forest is the overall best performing model for this problem.

4.2. Hyper-parameter tuned for Random Forest was the number of features in each tree (mtry). The value which gave best classification rate was 6. Random Forest is used to rate the importance of features in explaining the response variable. This helps us conclude that applicant income and loan amount are the most important determinant of loan application approval.

4.3. The tree-based methods don't have a non-parametric approach and don't make any assumptions regarding shape of the decision boundary. They make their decisions based only on values of the feature under consideration.

4.4. There is a trade-off while considering any model in a Machine Learning setting. One model is not the best for all problems. In the problem investigated in this project, there response variable distribution was highly skewed (~80% accepted) and therefore model selection was largely based on how good the model was at predicting the denied application. The better the model predictions in case of applications denied the worse the model performed in terms of accuracy (consider QDA). Therefore, there was a trade-off to consider between accuracy and specificity. Also, more complex models generally gave better results but due to their complexity and hyper-parameter tuning they were slower to implement and the increase in accuracy came at the cost of slower implementation time.