



UPPSALA
UNIVERSITET

**A COMPARISON OF DIFFERENT METHODS FOR
BANKRUPTCY PREDICTION**

Submitted by
Peter John Jarvis

*A thesis submitted to the Department of Statistics
in partial fulfillment of the requirements
for a one-year Master of Science degree in Statistics
in the Faculty of Social Sciences*

Supervisor
Rauf Ahmad

Spring, 2024

ABSTRACT

Corporate bankruptcy prediction is an important topic in accounting and finance, having significant relevance for many stakeholders, including investors, creditors, suppliers, and other key entities within the financial ecosystem who have an interest in the financial health of a company. This paper adds to the literature on comparing newer methods with older traditional methods in bankruptcy classification. The study uses a large dataset of American public companies listed on the New York Stock Exchange and NASDAQ, consisting of accounting data from 8,262 different companies spanning the period from 1999 to 2018. Logistic regression, randomForest, XGBoost, and Altman's Z-Score models are compared with and without the application of SMOTE (Synthetic Minority Oversampling Technique). The results show that the application of SMOTE significantly improves classification accuracy with regard to sensitivity and balanced accuracy for all models. The best performing model is XGBoost with a balanced accuracy of 60.83% without using SMOTE and 70.33% when using SMOTE. The results provide evidence that newer machine learning-based approaches outperform traditional methods and that SMOTE is an effective way to improve model performance.

Keywords: Bankruptcy, Machine learning, SMOTE, Imbalanced data, Z-Score model

Contents

1	Introduction	3
1.1	Literature Review	4
2	Theory	6
2.1	The Z-score model	6
2.2	Logistic Regression using LASSO	7
2.3	RandomForest	8
2.4	XGBoost	9
2.5	SMOTE	10
3	Data	11
4	Method	14
4.1	Classification Procedure	14
4.2	Evaluation Measures	16
5	Results	18
6	Discussion and Conclusion	19
6.1	Future research	22
7	References	23
	Appendices	25

1 Introduction

Corporate bankruptcy prediction is an important topic in accounting and finance, holding significant relevance for many stakeholders, including investors, creditors, suppliers, and other key entities within the financial ecosystem who have an interest in a company's financial health. A major reason for interest in the topic is due to the high costs associated with bankruptcy, in a study by Altman (1984) indirect and direct costs of bankruptcy are estimated to average at 11% to 17% of firm value three years prior to bankruptcy. Consequently, corporate bankruptcy has been studied extensively in the literature.

This paper adds to the literature by comparing newer methods with older traditional methods in bankruptcy classification. Logistic regression, randomForest, XGBoost and Altman's Z-Score models are compared on a fairly large dataset of 8,262 public American companies listed on the New York Stock Exchange and NASDAQ. The fact that bankruptcy data often is imbalanced, due to non-bankrupt data points being much more common than bankrupt data points, can make it harder for models to accurately predict bankruptcies, and often leads to the model being biased to overpredicting the majority class. In the case of this study the percentage of bankrupt firms is only 6.6%, which is low. Therefore, an application of SMOTE could improve the ability of the models to separate the two classes. A key research question of this study is therefore whether SMOTE (Synthetic Minority Oversampling Technique) can enhance the accuracy and balanced accuracy of machine learning and traditional methods for predicting bankruptcy in public American companies. Although SMOTE has been used in the context of bankruptcy classification, it has not yet been extensively studied on American companies, especially not on large datasets of public companies. There is one recent study conducted on American companies by Garcia (2022) who finds that SMOTE improves classification accuracy especially with regard to sensitivity and AUC, however the study by Garcia examines 1824 firms whereas this study will look at a much larger dataset of 8262 firms.

The hypothesis is that applying SMOTE to create a more balanced dataset will improve the performance of the models, particularly in terms of sensitivity and balanced accuracy, see for example Garcia (2022) who finds that SMOTE improves sensitivity and AUC scores compared to non-SMOTE trained models. The models are compared using the following performance metrics: Accuracy, sensitivity, specificity, balanced accuracy and AUC.

The results are mixed depending on what performance metric is used. However, balanced accuracy scores take into account both sensitivity and specificity and are therefore particularly

appropriate for evaluating models trained on imbalanced datasets. When comparing the models by this metric the best performing models are the SMOTE XGBoost and the SMOTE random-Forest, which indicates that SMOTE improves model performance and that newer machine learning-based approaches outperform the older methods of the Z-Score model and logistic regression. This could possibly be due to the machine learning models' ability to capture non-linear relationships in the data, something the linear models are not able to do.

The paper has the following structure: Section one introduces the subject and summarizes the most important findings in previous literature. Section two describes the models used in the study. Section three gives an overview of the dataset, discusses the different types of bankruptcy, and addresses the potential issue of multicollinearity. Section four describes the classification procedure, which includes how the models are trained and evaluated. Section five discusses the results. Section six discusses the results and summarizes the most important conclusions from the study, as well as provides suggestions for future research.

1.1 Literature Review

The first attempt to predict corporate bankruptcy using a statistical method can be attributed to Beaver (1966), who used financial ratios in a univariate analysis. Later Altman (1968) used multiple discriminant analysis to predict bankruptcy, also using common financial ratios as explanatory variables. The model created by Altman is commonly called the "Z-score" model and consists of a weighted sum of profitability, leverage, liquidity, solvency, and activity ratios indicating a likelihood of a company going bankrupt within a specified time frame. Depending on the specific use case, one can use different cut-off points for predicting bankruptcy, which makes the model highly flexible and easy to use. The model assumes that the explanatory variables follow a multivariate normal distribution, which has been questioned in subsequent research. Nonetheless, Altman's model has been used by auditors, accountants, banks and judges to assess the probability of a company going bankrupt and has been found to be useful in the near-term (1-2 years).

Following on from Altman, Ohlson (1980) employed a logistic regression model to predict corporate bankruptcy, using similar financial ratios as explanatory variables as in previous research. One improvement from earlier efforts from Beaver and Altman is the ability to provide a probability of default, as Beaver's and Altman's models only give a score, which is then used to implement a decision boundary for classification purposes. Lennox (1999) also used logit

and probit models on a sample of data from the United Kingdom and found that these models outperformed the type of discriminant analysis that Beaver and Altman used. Lennox also argued that some of the commonly used explanatory variables in bankruptcy prediction, like leverage and cash flow ratios, have significant non-linear effects, and that taking into account these non-linearity's in explanatory variables is beneficial for model performance.

In more recent years more advanced methods have been used in bankruptcy prediction, many of which do not require any underlying assumptions of the distribution of the data and can therefore be useful if there are non-linear relationships in the data as evidenced by Lennox (1999). Barboza et al. (2017) compare different machine learning techniques including bagging, boosting, support vector machines, and random forest and compare these with traditional methods like discriminant analysis and logistic regression as well as neural networks. On a sample of 10,000 American firms from 1985 to 2013 they find that machine learning models achieve about a 10% higher accuracy on average than traditional methods like discriminant analysis and logistic regression.

Another study by Tsai, Hsu, and Yen (2014) also compared different machine learning methods including support vector machines, multilayer perceptrons, decision trees, and neural networks as well as used bagging and boosting. They find that a decision tree ensemble using 80-100 classifiers and boosting performs the best.

A study by Garcia (2022) compared different methods for oversampling and undersampling data, including SMOTE, to deal with the class imbalance issue that is common in bankruptcy classification. The study used various models, including XGBoost and randomForest, and evaluated them on a dataset of 1824 US firms. The results show that the application of SMOTE, or one of its extensions, significantly improves classification accuracy with regard to sensitivity and AUC scores, thus providing evidence that SMOTE can improve machine learning models when applied to bankruptcy classification on American companies. Especially noteworthy were the large improvements in sensitivity when using SMOTE. The results from the study also show that machine learning methods outperform traditional models like logistic regression, providing further evidence that more complex models perform better in the context of bankruptcy classification.

Additional evidence that SMOTE can improve the classification accuracy of models for imbalanced data is provided in a study by Veganzones and SÁ©verin (2018). They find that when using a model trained on a dataset that has 20% or less of the minority class represented

it significantly impacts the performance of the model with regard to sensitivity. The greater the class imbalance, the worse the model performs. The study also finds that SMOTE was the resampling technique that improved the performance the most out of a total of four compared methods.

2 Theory

2.1 The Z-score model

The Z-score model was first described by Altman in his seminal paper "*Financial ratios, discriminant analysis and the prediction of corporate bankruptcy*" in 1968.

The model uses a form of multivariate discriminant analysis and combines five financial ratios to determine bankruptcy. Below the model is described mathematically:

$$Z = 1.2X_1 + 1.4X_2 + 3.3X_3 + 0.6X_4 + 1.0X_5$$

Where:

- X_1 is the Working Capital/Total Assets ratio,
- X_2 is the Retained Earnings/Total Assets ratio,
- X_3 is the EBIT/Total Assets ratio,
- X_4 is the Market Value of Equity/Book Value of Total Liabilities ratio, and
- X_5 is the Sales/Total Assets ratio.

The model aims to find the best linear combination of the five financial ratios to separate bankrupt and non-bankrupt firms. The resulting Z-score is a numerical value. The higher the Z-score the lower the probability of bankruptcy and vice versa. One decision that has to be made is the threshold for the Z-score value to determine where one wants to separate the classes, this decision has implications for the trade-off in specificity and sensitivity in the model. A higher threshold will increase the specificity at the cost of reducing sensitivity, and the lower the threshold will increase sensitivity and reduce specificity. The threshold can be adjusted accordingly, depending on whether reducing false negatives or false positives is more important.

2.2 Logistic Regression using LASSO

Logistic regression is a commonly used model when modeling binary outcomes. As this study deals with a binary outcome, logistic regression is a natural choice of model to implement. The model calculates the probability of the outcome variable Y taking on one of the two possible outcomes as a function of a set of linear predictors X .

The model is described below:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k \quad (1)$$

where p is the probability that $Y = 1$, $\beta_0, \beta_1, \dots, \beta_k$ are the parameters of the model and X_1, X_2, \dots, X_k are the predictor variables. The probability that $Y = 1$ is then given by the logistic function:

$$p = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k)}} \quad (2)$$

Logistic regression is an adaptable model that offers a high degree of interpretability, allowing a clear understanding of how predictor variables impact the outcome. The log odds of the outcome variable is a function of the predictor variables, where a one-unit increase in a predictor variable leads to a corresponding change in the log odds of the outcome, see (Hosmer Jr, Lemeshow, and Sturdivant, 2013) for a detailed overview of the method.

The parameters in a logistic regression model can be estimated using various methods. Maximum Likelihood Estimation (MLE) is a common choice; however, in datasets with a high correlation between predictor variables, alternative methods such as Ridge and LASSO regression can be more effective. Given the high correlation among the predictor variables in this dataset, as indicated by the correlation plot in the "Data" section of the study, LASSO will be employed for estimating the parameters of the logistic regression model. LASSO regression was first described by Tibshirani (1996). This method minimizes a loss function that includes a penalty term proportional to the absolute value of the coefficients, effectively reducing overfitting and aiding in feature selection by shrinking some coefficients to zero, which is helpful when there is a lot of multicollinearity between the explanatory variables.

The LASSO method for logistic regression is shown below:

$$\min_{\beta} \left\{ -\frac{1}{N} \sum_{i=1}^N \left[y_i(\beta_0 + \beta^T x_i) - \log(1 + e^{\beta_0 + \beta^T x_i}) \right] + \lambda \sum_{j=1}^p |\beta_j| \right\} \quad (3)$$

where N is the number of observations, y_i represents the binary outcome for the i -th observation, β_0 is the intercept and β is a vector of the predictor variables.

The term $\beta^T x_i$ represents the dot product of the coefficient vector and the predictor variable vector for the i -th observation. The regularization parameter λ controls the degree of the LASSO penalty, larger values lead to greater shrinkage of the coefficients towards zero. The objective function comprises the log-likelihood of the logistic regression model, measuring the fit of the model to the data, see (Tibshirani, R. 1996) for a comprehensive overview of LASSO logistic regression.

2.3 RandomForest

The randomForest model was first described by Breiman (2001) and uses an ensemble of decision trees for classification and regression purposes. By combining many different trees, one can reduce bias and improve overall accuracy according to Breiman (2001).

The way the model works is by independently training decision trees on random subsamples of the data and only considering a random set of predictor variables for each split in the decision tree. The trees are then combined in an ensemble, and for classification the majority vote of the trees is the output of the model. Below is a mathematical description of the model:

For classification tasks, the prediction \hat{y} of the Random Forest is obtained by:

$$\hat{y} = \operatorname{argmax}_y \sum_{t=1}^T \mathbf{1}\{f_t(\mathbf{x}) = y\},$$

where $\mathbf{1}\{\cdot\}$ is the indicator function and y ranges over all possible class labels.

Below is a schematic description of how a randomForest works:

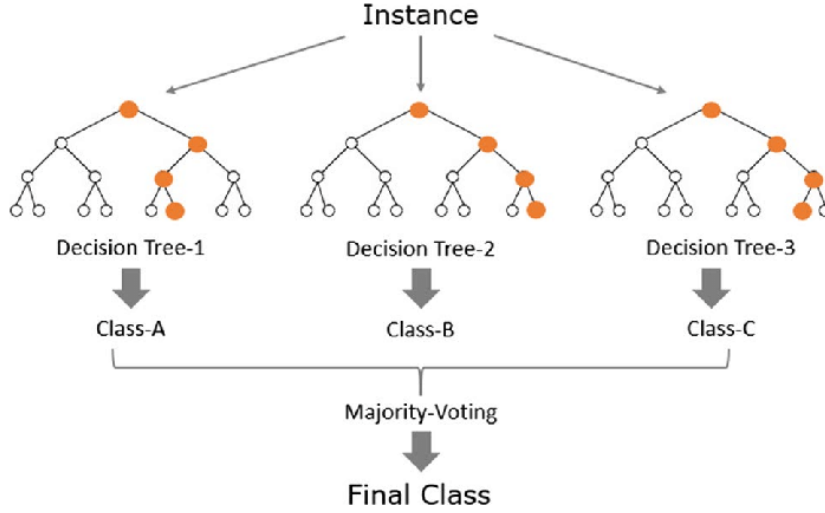


Figure 1: Schematic depiction of the randomForest algorithm (Golze, Zourlidou & Sester, 2020).

2.4 XGBoost

XGBoost stands for eXtreme Gradient Boosting and was first described by Chen and Guestrin (2016). The model is an ensemble decision tree that sequentially combines many individual weak decision trees.

Let the dataset be $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$, where \mathbf{x}_i is the feature vector for the i -th occurrence and y_i is the target variable.

Then let $F_k(\mathbf{x})$ denote the model in the k -th iteration.

At every iteration, XGBoost optimizes a function by selecting the splits at each iteration that improve the model the most.

$$\mathcal{L}^{(k)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(k)}) + \sum_{i=1}^k \Omega(F_i),$$

where l is the loss function that measures the difference between predicted $\hat{y}_i^{(k)}$ and true y_i , and $\Omega(F_i)$ is a regularization term that penalizes the complexity of the model F_i . This helps to reduce potential overfitting problems.

The model $F_k(\mathbf{x})$ in the k -th iteration is then found by minimizing the function $\mathcal{L}^{(k)}$:

$$F_k(\mathbf{x}) = F_{k-1}(\mathbf{x}) + \arg \min_{h \in \mathcal{H}} \sum_{i=1}^n [l(y_i, \hat{y}_i^{(k-1)} + h(\mathbf{x}_i))] + \Omega(h),$$

where \mathcal{H} incorporates all potential weak learners.

The final prediction \hat{y} is equal to the majority vote of all individual weak learners.

$$\hat{y} = \sum_{k=1}^K F_k(\mathbf{x}),$$

where K is the total number of iterations that have been executed.

2.5 SMOTE

SMOTE stands for synthetic minority oversampling technique and can be applied in cases where the dataset has a class imbalance. SMOTE was first described in a paper by Chawla et al. (2002) highlighting that in real world datasets often the abnormal cases are the interesting cases, in this paper the bankrupt firms are what we are most interested to detect, which is difficult if they only represent a small fraction of the entire sample. SMOTE involves synthetically creating new samples by oversampling the minority class and therefore creating a more balanced dataset that hopefully improves prediction accuracy.

SMOTE works by following a procedure of four steps:

1. **Identify the Minority Class:** First the minority class in the dataset is identified, in this study that is the case where a firm is bankrupt.
2. **Select the K-Nearest Neighbors:** Secondly, one selects the k nearest neighbours in the feature space that are closest to the minority class.
3. **Generate Synthetic Samples:** Then one randomly generate new samples by joining the line segments of the k minority class nearest neighbors.
4. **Add Synthetic Samples:** Finally the synthetically created samples are added back to the original dataset, thus effectively oversampling the minority class.

Below is a mathematical description of the oversampling process:

Let x_i be a minority class sample and k its nearest neighbors in the feature space, which are denoted as x_{nn} , a synthetic sample x_{new} is then created along the line segment joining x_i and x_{nn} in the feature space by selecting a random value in $\delta \in [0, 1]$:

$$x_{new} = x_i + \delta \times (x_{nn} - x_i)$$

where x_i and x_{nn} are vectors representing feature vectors in the dataset.

The process above is repeated until you achieve the desired balance in the dataset. One can choose how many k nearest neighbours to generate new data from. For example, as described by Chawla et al. (2002), if you want to oversample 200% of the original amount in the minority class, you can select the two nearest neighbors from which to generate new samples. Therefore, the method is flexible and it is easy to control the amount of over-sampling needed.

3 Data

The dataset used in the study consists of American public companies listed on the New York Stock Exchange (NYSE) and NASDAQ, consisting of accounting data from 8,262 different companies spanning the period from 1999 to 2018. There are no missing values in the dataset. The dataset is public and has been compiled and provided in a study by Lombardo et al. (2022).

The Securities and Exchange Commission (SEC) provides two distinct scenarios of bankruptcy in America. The first scenario occurs when a company files for Chapter 11 bankruptcy, indicating its intention to restructure its balance sheet and continue to function. This typically involves a comprehensive review of the company's financial structure, with a focus on restructuring debt and equity instruments to better align with its current circumstances. The objective is to keep operating the business as a going concern.

Under Chapter 11 bankruptcy, the company's management retains control over day-to-day operations. However, some decisions may require the approval of the bankruptcy court to ensure that the interests of all stakeholders are protected.

The other scenario is called a Chapter 7 which means that the Company will cease operating as a going concern and all assets are liquidated. The proceeds are distributed among creditors and equity holders according to their priority.

In the dataset bankruptcy from either Chapter 11 or Chapter 7 is labelled the same (Failed) and conversely if the Company does not experience any Chapter 7 or Chapter 11 filing the label is (Alive), where this variable is coded as a factor variable. The fiscal year before the chapter filing is labelled as bankrupt for the next year. This ensures that the dataset makes it possible for the models to predict bankruptcy at least one year before it happens (Lombardo et al. 2022). The potential bankruptcies in 2018 are also possible to be treated as other years, since the dataset was compiled using filings including 2019 as well, ensuring the look-ahead period of one year to be the same for 2018 as for all other years in the data. One potential issue

with the data occurs if for example there is a bankruptcy in January and one in December in 2015, as these would both be labelled the same as bankrupt in 2014 with no way of telling that there is actually almost an entire year in difference from the two filings in the data. Therefore it would be better if the data was more granular so that one could use maybe quarterly or monthly data to deal with this potential problem, however unfortunately this dataset does not enable this.

Bris et al. (2006) found in a study on American firms from 1995 to 2001 that Chapter 7 proceedings offer few advantages over Chapter 11 proceedings, as they tend to take almost as long to resolve, offer lower recovery rates for creditors, and cost about the same.

Table 1: Description of Variables

Variable	Description
Current_Assets	Current assets of the company
Cost_of_Goods_Sold	Cost of goods sold
Depreciation_and_Amortization	Depreciation and amortization
EBITDA	Earnings before interest, taxes, depreciation, and amortization
Inventory	Inventory value
Net_Income	Net income of the company
Total_Receivables	Total receivables
Market_Value	Market value of the company
Net_Sales	Net sales of the company
Total_Assets	Total assets of the company
Total_Long_term_Debt	Total long-term debt
EBIT	Earnings before interest and taxes
Gross_Profit	Gross profit
Total_Current_Liabilities	Total current liabilities
Retained_Earnings	Retained earnings
Total_Revenue	Total revenue
Total_Liabilities	Total liabilities of the company
Total_Operating_Expenses	Total operating expenses
Status_label	Bankruptcy status Alive/Failed

Table 1 gives an overview of the exploratory variables and the outcome variable in the data. All variables come from accounting data and can be derived from the balance sheet,

income statement, and cash flow statement of the companies. The outcome variable is called *Status_label* and is simply a binary outcome variable that takes the value 0 if the company is not bankrupt and the value 1 if the company is bankrupt in the current time period. Chapter 7 and Chapter 11 bankruptcies are not differentiated in the data.

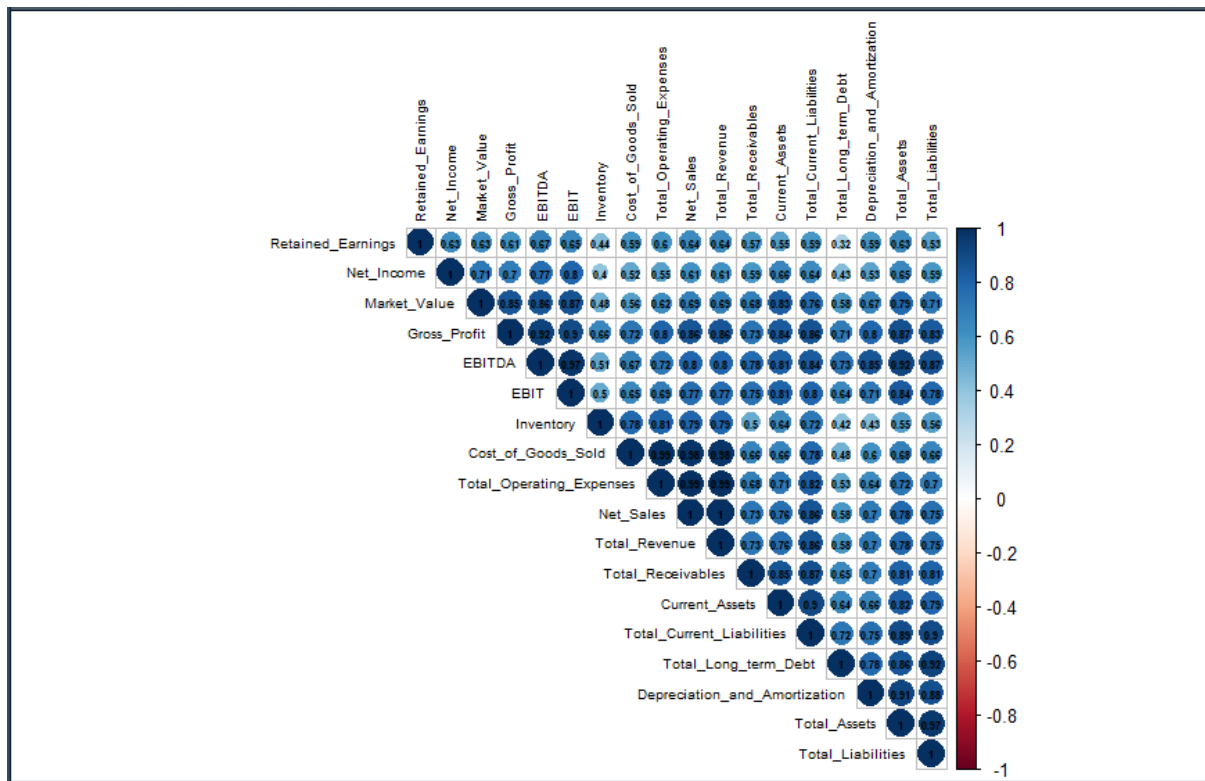


Figure 2: Correlations between the explanatory variables in the dataset

Figure 2 gives a visual representation of the correlation between the explanatory variables in the dataset. The figure gives an indication of the presence of multicollinearity in the data, as there are a lot of correlations above 0.8 and even 0.9 in the dataset. This does make sense, since many of the ratios in the data are derived from similar accounting measurements like revenue, cost of goods sold, expenses, etc. The presence of multicollinearity could be an issue for the logistic regression model, however, the use of LASSO as a regularization technique helps to deal with this problem, as LASSO shrinks parameters towards zero, as explained in a study by Chan et al. (2022), effectively performing feature selection.

The XGBoost and randomForest models do not rely on any assumption of independence between explanatory variables and are able to capture complex non-linear relationships between variables in the data. As the randomForest and XGBoost models only consider a subset of all explanatory variables at each split, and the XGBoost also includes a penalization term,

these models should therefore be more robust to multicollinearity than the logistic regression model. That machine learning models are more robust to multicollinearity is corroborated by Chan et al. (2022) who conclude that machine learning methods in general fit data with multicollinearity better than OLS regression.

Since the Z-score model only selects five of the explanatory variables by design it does not have the same need as the other models to address any potential multicollinearity issues.

4 Method

4.1 Classification Procedure

The data from 8,262 different companies listed on the NYSE and NASDAQ spanning the period from 1999 to 2018 was first loaded into R, which is the program used for all analysis in the study. Next the *year* and *company name* variables were excluded from the data as they were assumed to not have any explanatory power for the study and the outcome variable *status_label* was re-coded to a factor variable. The dataset was then split into two sets, with 80% intended for training and 20% for testing of the models, resulting in 62945 observations for training and 15737 observations for testing. The split was done using random sampling. This is a common approach to ensure the models are trained on a sufficient amount of data and then tested on out-of-sample data to ensure that the models are tested on unseen data. A table showing the distribution of the bankrupt and non-bankrupt data points for the two datasets is presented below:

Table 2: Distribution of Bankruptcy Status in Training and Test Sets

Dataset	Non-Bankrupt (0)	Bankrupt (1)
Training Set	58777 (93.41%)	4168 (6.59%)
Test Set	14685 (93.32%)	1052 (6.68%)

However, the training and test sets described in Table 2 are only used for the models where SMOTE was not applied to the training data. For the models that were trained using SMOTE the training and test sets are instead described below:

As shown in Table 3 the training set for the SMOTE models is balanced with 50% bankrupt companies and 50% non-bankrupt companies, however the testing set is left exactly the same

Table 3: Distribution of Bankruptcy Status Post-SMOTE in Training Set and Original Test Set

Dataset	Non-Bankrupt (0)	Bankrupt (1)
Training Set (SMOTE)	58777 (50.00%)	58777 (50.00%)
Test Set (Original)	14685 (93.32%)	1052 (6.68%)

to ensure all models are tested on the same data. See Section 2.6 in the study for a detailed description of the SMOTE oversampling process. The SMOTE algorithm was implemented using the *recipe* function from the *recipes* package.

For logistic regression analysis, LASSO (Least Absolute Shrinkage and Selection Operator) was implemented using the *cv.glmnet* function from the *glmnet* package to find the optimal value for the regularization parameter *lambda*. This was done using 10-fold cross-validation on the training data. Cross-validation also helps prevent overfitting. Then the model was tested and evaluated on the test set, 0.5 was used as the probability threshold for predicting observations as bankrupt or non-bankrupt. All performance metrics were then calculated using the *pROC* and *caret* packages. The exact same process was implemented for the SMOTE LASSO logistic regression model, the only difference being that the SMOTE LASSO logistic regression model was trained on the balanced dataset shown above in Table 3.

The randomForest models were trained using 5-fold cross-validation and the *train* function of the *caret* package. The "ranger" method for faster computation was implemented. Both models used 500 as the number of trees to grow during training, and the optimal number of variables to consider at each split *mtry* was found using grid search. A general rule of thumb when using the randomForest algorithm for classification tasks is to use the square root of the number of explanatory variables as the value for the *mtry* parameter. Therefore the grid search was centered around this value, since it should logically be a good starting point. The same process was used for both the SMOTE randomForest model and the baseline randomForest model, the only difference being the usage of the balanced dataset for the SMOTE randomForest, ensuring that the models were trained and evaluated exactly the same except for the difference in training data used.

The XGBoost models were trained using 5-fold cross-validation and the *train* function from the *caret* package. The "xgbTree" method was specified for training XGBoost models, and 500 boosting rounds were implemented. Optimal hyperparameter values for the learning rate and

depth of trees were found using grid search, similarly to what was done for the randomForest models. As for all other models the same process was used for the baseline XGBoost model and the SMOTE XGBoost model, only difference being the training data used.

Finally, the Z-Score models were calculated and evaluated. These models are not trained in the same way that the other models are since they rely on calculating a Z-Score value for all observations and applying a decision threshold for predicting whether a single datapoint is likely to be a bankrupt company or a non-bankrupt company. The Z-Score value was calculated for the three thresholds 1.1, 1.8 and 2.5, as described in section 2.2 in the study. The predictions were then made using the threshold as the decision rule, if the Z-Score for a specific company was lower than the threshold the company was classified as bankrupt and if it was higher than the threshold value it was classified as non-bankrupt. The rationale being the higher the Z-Score value the lower the likelihood of the company being bankrupt. The reason for choosing three different thresholds is to see how the model's classification decisions changes with a different threshold, these values were deemed to give a representative view of the models potential performance over different threshold values.

4.2 Evaluation Measures

There are many metrics that can be used to evaluate a binary classifier. The choice of evaluation measures is especially important when the dataset is imbalanced, as some evaluation measures can provide misleading results in such cases. An example of this is accuracy, which is commonly used to evaluate binary classifiers, however, Thölke et al. (2023) argues that as accuracy weights the per-class ratios with regard to class size proportionally, it can suffer from the issue of class imbalance and gives increasingly high performance as the class imbalance increases, which is misleading. Therefore it is important to carefully select evaluation metrics that give a good representation of the models' actual performance. In this study, the following five evaluation metrics are used: Accuracy, Sensitivity, Specificity, Balanced Accuracy and AUC. Balanced accuracy and AUC are better choices for imbalanced data according to Thölke et al. (2023) as they do not suffer the same issues as accuracy does. The metrics are described and discussed below.

Accuracy

Accuracy is defined as the ratio of correct predictions to the total number of cases. Accuracy

can however be misleading in the case of imbalanced data and should be interpreted with care as previously mentioned. The mathematical calculation for accuracy is shown below:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Number of Cases}}$$

Sensitivity

Sensitivity measures the proportion of actual positives that are correctly identified. In the context of this study it measures the model's ability to correctly predict bankrupt firms as bankrupt. The mathematical calculation for sensitivity is shown below:

$$\text{Sensitivity} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

Specificity

Specificity measures the proportion of actual negatives that are correctly identified. In the context of this study it measures the models ability to correctly classify a non-bankrupt firm as non-bankrupt. The mathematical calculation for specificity is shown below:

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}$$

Balanced Accuracy

Balanced accuracy is the arithmetic mean of sensitivity and specificity. This metric is especially useful for imbalanced datasets and gives a good representation of the models overall ability to correctly predict both classes and therefore Thölke et al. (2023) recommend its use for imbalanced data. The mathematical calculation for balanced accuracy is shown below:

$$\text{Balanced Accuracy} = \frac{\text{Sensitivity} + \text{Specificity}}{2}$$

Area Under the ROC Curve (AUC)

The AUC measures the ability of a classifier to distinguish between classes and is used as a summary of the ROC curve. The ROC plots the True Positive Rate (TPR) against the False Positive Rate (FPR) over all different threshold settings, showing the trade-offs between sensitivity and specificity. The AUC is a measure of this separability; a higher AUC indicates better model performance. The mathematical calculation for AUC is shown below:

$$\text{AUC} \approx \sum_{i=1}^{n-1} \frac{(\text{FPR}_{i+1} - \text{FPR}_i) \times (\text{TPR}_i + \text{TPR}_{i+1})}{2} \quad (4)$$

5 Results

Table 4: Comparison of performance metrics for all models. Baseline RF = randomForest model without using SMOTE, SMOTE RF = randomForest model using SMOTE, Baseline XGB = XGBoost model without using SMOTE, SMOTE XGB = XGBoost model using SMOTE, LASSO LR = Logistic regression using LASSO for regularization, SMOTE LASSO LR = Logistic regression using both LASSO and SMOTE, Z-Score (1.1 th.) = Z-Score model using 1.1 as the threshold, Z-Score (1.8 th.) = Z-Score model using 1.8 as the the threshold, Z-Score (2.5 th.) = Z-Score model using 2.5 as the threshold.

	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUC
Baseline RF	0.9369	0.0570	0.9999	0.5285	0.8719
SMOTE RF	0.9225	0.4496	0.9564	0.7030	0.8686
Baseline XGB	0.9448	0.2196	0.9967	0.6082	0.8949
SMOTE XGB	0.9192	0.5182	0.9477	0.7330	0.8665
LASSO LR	0.9330	0.0057	0.9994	0.5025	0.6652
SMOTE LASSO LR	0.3743	0.3397	0.8565	0.5981	0.6791
Z-Score (1.1 th.)	0.2560	0.6098	0.2309	0.4203	0.6188
Z-Score (1.8 th.)	0.3246	0.4906	0.3129	0.4017	0.6188
Z-Score (2.5 th.)	0.4068	0.3898	0.4080	0.3989	0.6188

Table 2 summarizes the results of the study and compares the performance of the different models using accuracy, sensitivity, specificity, balanced accuracy, and the AUC score. Looking at the accuracy metric, the baseline XGBoost model has the best score with an accuracy of 94.48%, followed by the baseline randomForest model with an accuracy of 93.69% and the third best is the LASSO logistic regression with an accuracy of 93.30%.

If we instead consider sensitivity, the performance of the models is quite different. The Z-score model at a threshold value of 1.1 has the best score of 60.98%. Here sensitivity measures the ability of the model to correctly classify bankruptcies. The SMOTE XGBoost model performs second best with 51.82% and the third best model is the Z-score model using 1.8 as the threshold, which has a sensitivity of 49.06%. Specificity measures the ability of the model to correctly identify non-bankrupt cases in the data, here the baseline randomForest model performs the best with 99.99% in specificity, followed by the LASSO logistic regression and the baseline XGBoost model with 99.94% and 99.67% respectively.

To be noted is the poor ability of all Z-score models to identify non-bankrupt companies

having specificities of 40.80%, 31.29% and 23.09% for the 2.5, 1.8 and 1.1 thresholds, respectively.

For the balanced accuracy metric, the best performing model is the SMOTE XGBoost model, with a balanced accuracy of 73.30%, followed by the SMOTE randomForest model and the SMOTE LASSO logistic regression models with 70.30% and 59.81% in balanced accuracy, respectively.

Finally looking at the AUC metric the baseline XGBoost model has the highest score of 0.8949, followed by 0.8719 and 0.8686 for the baseline randomForest and the SMOTE randomForest respectively. The three Z-score models perform the worst with an AUC of 0.6188.

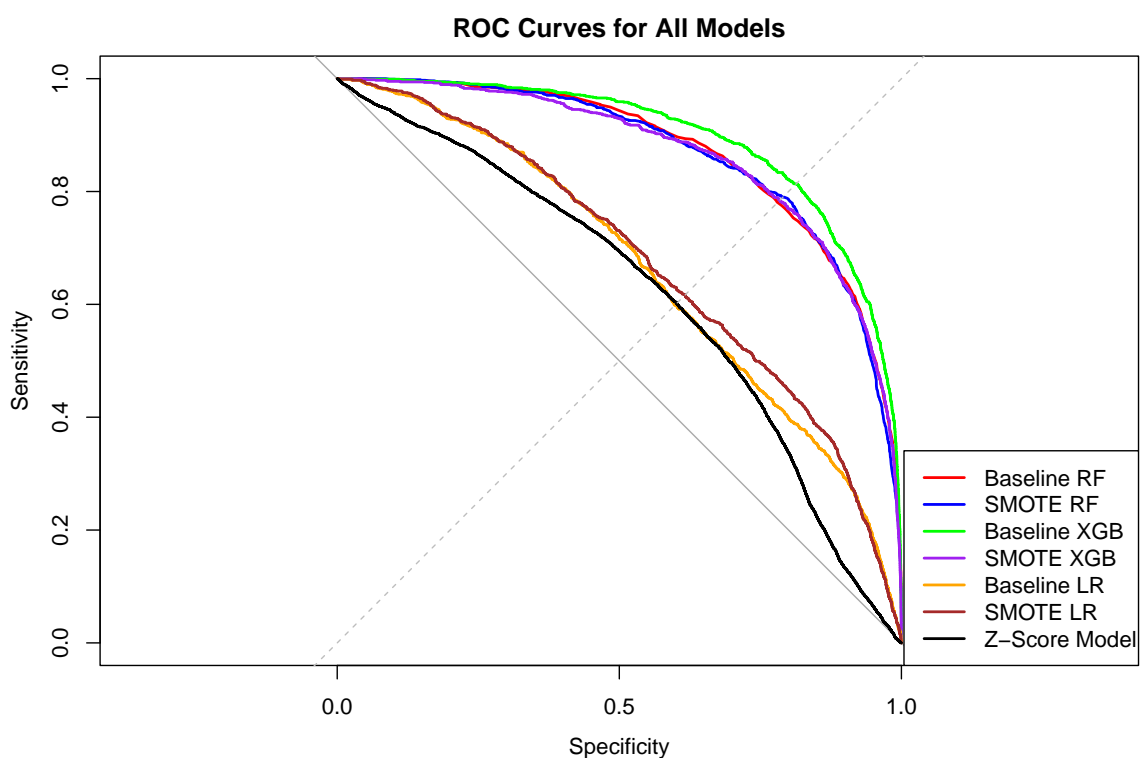


Figure 3: ROC curve for all models

Figure 3 shows the ROC curves for all models. The baseline XGBoost model has the highest AUC (area under the curve).

6 Discussion and Conclusion

The results vary significantly depending on the model and which performance metric is used, illustrating the inherent trade-offs between different models and the application of SMOTE for

resampling. Notably, all models trained without SMOTE have low sensitivity, except the Z-Score models, which is probably due to the imbalanced nature of the original dataset, making it difficult for the models to distinguish between the classes. The results support this observation and show that without SMOTE, models tend to overpredict the majority class (non-bankrupt), achieving high specificity and accuracy, but at the cost of low sensitivity and much lower balanced accuracy. This could be a case of models that overfit the characteristics of the majority class. Looking at the performance for the baseline randomForest model this seems to be the case, with 99.99% in specificity, 5.70% in sensitivity and an accuracy of 93.69%. Similar results are observed with the LASSO logistic regression and the baseline XGBoost model, although the latter shows a somewhat improved sensitivity of 21.96%, indicating slight variations in how different models manage class imbalance. The LASSO logistic regression has a very low sensitivity of 0.057% which may indicate that linear models may have trouble separating the classes, and that non-linear models are better at discriminating between the classes. Between the baseline models, the XGBoost model achieves the highest balanced accuracy score of 60.82%, which indicates that if one does not use any resampling technique, the XGBoost model performs the best. In general, the linear models perform the worst in balanced accuracy, indicating that there may be some non-linear relationships in the data that these models are not able to capture, and therefore models that are able to capture non-linear relationships perform better.

The Z-Score models do not have the same issue of overpredicting the majority class, having sensitivities of 60.98%, 49.06%, and 38.98% for the threshold levels 1.1, 1.8, and 2.5, respectively. However, this is to be expected as the Z-Score model does not train on the data like the other models do, and it does not suffer from the imbalance issue the same. It simply gives a decision for every data point based on a linear combination of certain financial ratios. However, the specificity scores for the Z-Score models are low with 23.09%, 31.29% and 40.80% for the 1.1, 1.8 and 2.5 thresholds respectively which of course negatively impacts the models overall accuracy and balanced accuracy. This shows the trade-off in sensitivity and specificity depending on the choice of threshold value for the model with a higher threshold value giving a higher sensitivity but lower specificity. So, while the Z-Score models are able to achieve a high sensitivity, there is a clear trade-off in worse scores in specificity.

For the models trained using SMOTE the results are quite different from the non-SMOTE trained models. The sensitivity for these models is much higher than their non-SMOTE coun-

terparts at 44.96%, 51.82% and 33.97% for the randomForest, XGBoost and LASSO logistic regression models, respectively. This result indicates that the application of SMOTE has vastly improved the models ability to correctly identify bankruptcies, which would make sense and is in line with the hypothesis that balancing the classes in the training data would improve the models ability to learn the characteristic features of bankrupt data points. Interestingly, the models only perform slightly worse with regard to specificity, which leads to a higher balanced accuracy for all models. So, applying SMOTE vastly improves sensitivity at the cost of only slightly worsening specificity, leading to a higher balanced accuracy. Looking at accuracy, the SMOTE models have a slightly lower score compared with their non-SMOTE counterparts, showing the issue that accuracy can be misleading when evaluating models trained on imbalanced data. The model that performed best when evaluated on balanced accuracy and using SMOTE was the XGBoost model, scoring 73.30%, which is in line with previous research, for example Tsai, Hsu, Yen (2014) also found that ensemble methods using boosting is the best method for bankruptcy prediction.

The results support the hypothesis that SMOTE improves the sensitivity and balanced accuracy of machine learning classifiers for bankruptcy prediction by significantly enhancing sensitivity, with only a minor decrease in specificity. These findings align with previous research, although this study did not observe improvements in AUC scores for SMOTE-trained classifiers, as was reported by Garcia (2022).

In conclusion, the study finds that the logistic regression, randomForest, and the XGBoost models achieve high accuracy and specificity, but do so at the expense of sensitivity and balanced accuracy. These models tend to overpredict the majority class, leading to high specificity but low sensitivity. This tendency to overpredict the majority class likely comes from the imbalanced nature of the dataset used, leading to models that excel in identifying non-bankrupt cases while failing to detect actual bankruptcies. The Z-Score models do not have the same issue; instead they are capable of achieving high sensitivity, but instead at the cost of very low specificity and even lower balanced accuracy scores.

The application of SMOTE vastly improves the sensitivity and only slightly worsens specificity of the logistic regression, randomForest, and XGBoost models, leading to a higher balanced accuracy. The best performing models by balanced accuracy are the XGBoost model, followed by the randomForest model when using SMOTE for training. This shows that SMOTE is an effective way to improve model performance in the context of bankruptcy prediction.

6.1 Future research

For future research, one can explore different methods of balancing the data, including oversampling, undersampling, and combinations of various resampling techniques. Another potential area of research is the impact of feature engineering, as well as the combination of different models and resampling methods, on model performance in the context of bankruptcy prediction using imbalanced data. Additionally, examining how the inclusion of other explanatory variables, such as macroeconomic indicators like GDP, inflation, and unemployment rates, affects model performance is another potential area for future studies. Lastly, it could be interesting to investigate if there are differences between Chapter 7 and Chapter 11 bankruptcy proceedings with regards to model performance, as the two scenarios may involve different financial situations and therefore one may achieve different results by modelling each scenario separately.

7 References

- Altman, E. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *Journal of Finance*, 23, 589â609.
- Altman, E. (1984). A further empirical investigation of the bankruptcy cost question. *The Journal of Finance*, 39, 1067-1089. <https://doi.org/10.1111/j.1540-6261.1984.tb03893.x>
- Altman, E., Haldeman, R., & Narayanan, P. (1977). ZETA analysis: A new model to identify bankruptcy risk of corporations. *Journal of Banking and Finance*, 1, 29â54.
- Barboza, F., Kimura, H., & Altman, E. (2017). Machine learning models and bankruptcy prediction. *Expert Systems with Applications*, 83, 405â417.
- Beaver, W. (1966). Financial ratios as predictors of failure. *Journal of Accounting Research*, 4, 71â111.
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5â32.
- Bris, A., Welch, I., & Zhu, N. (2006). The costs of bankruptcy: Chapter 7 liquidation versus Chapter 11 reorganization. *The Journal of Finance*, 61(3), 1253â1303.
- Chan, J., Leow, S., Bea, K., Cheng, W. K., Phoong, S. W., Hong, Z. W., & Chen, Y. L. (2022). Mitigating the multicollinearity problem and its machine learning approach: A review. *Mathematics*, 10, 1283. <https://doi.org/10.3390/math10081283>
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321â357.
- Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785â794). ACM.
- Garcia, J. (2022). Bankruptcy prediction using synthetic sampling. *Machine Learning with Applications*, 9, 100343. <https://doi.org/10.1016/j.mlwa.2022.100343>
- Golze, Jens & Zourlidou, Stefania & Sester, Monika. (2020). Traffic Regulator Detection Using GPS Trajectories. *KN - Journal of Cartography and Geographic Information*

Hosmer, D. W., Jr., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression* (3rd ed.). Wiley.

Lennox, C. (1999). Identifying failing companies: A re-evaluation of the logit, probit and DA approaches. *Journal of Economics and Business*, 51(4), 347â364.

Lombardo, G., Pellegrino, M., Adosoglou, G., Cagnoni, S., Pardalos, P. M., & Poggi, A. (2022). Machine learning for bankruptcy prediction in the American stock market: Dataset and benchmarks. *Future Internet*, 14(8), 244. <https://doi.org/10.3390/fi14080244>

Ohlson, J. A. (1980). Financial ratios and the probabilistic prediction of bankruptcy. *Journal of Accounting Research*, 18(1), 109â131.

Thölke, P., Mantilla-Ramos, Y. J., Abdelhedi, H., Maschke, C., Dehgan, A., Harel, Y., Kemtur, A., Mekki Berrada, L., Sahraoui, M., Young, T., Bellemare PÃ©pin, A., El Khantour, C., Landry, M., Pascarella, A., Hadid, V., Combrisson, E., OâByrne, J., & Jerbi, K. (2023). Class imbalance should not throw you off balance: Choosing the right classifiers and performance metrics for brain decoding with imbalanced data. *NeuroImage*, 277, 120253. <https://doi.org/10.1016/j.neuroim>

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1), 267â288.

Tsai, C. F., Hsu, Y. F., & Yen, D. C. (2014). A comparative study of classifier ensembles for bankruptcy prediction. *Applied Soft Computing*, 24, 977â984.

Appendices

A Results: Individual ROC curves

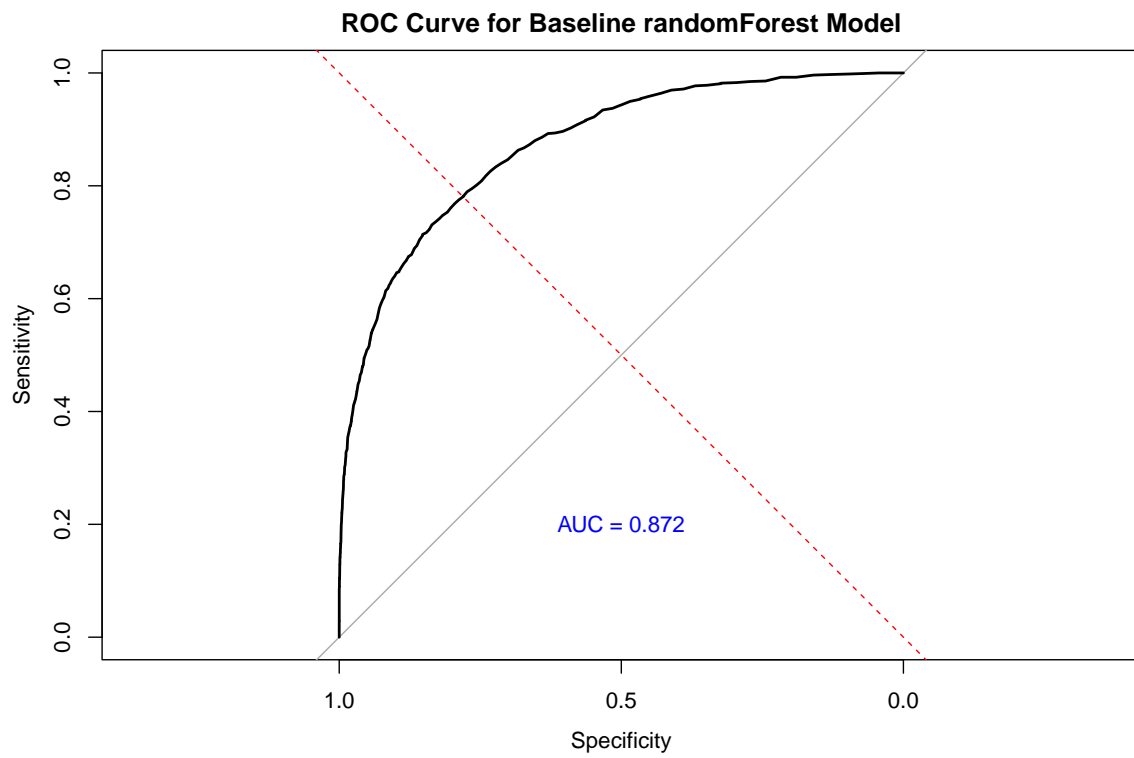


Figure 4: ROC curve for baseline randomForest model

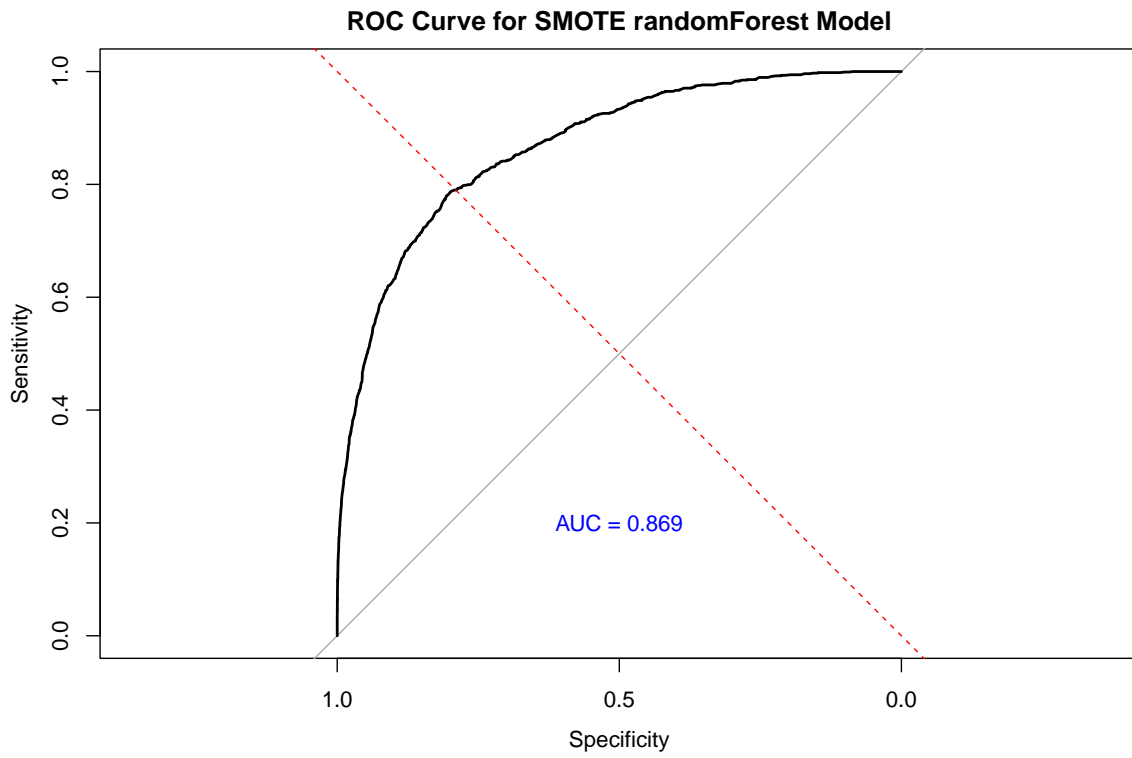


Figure 5: ROC curve for SMOTE randomForest model

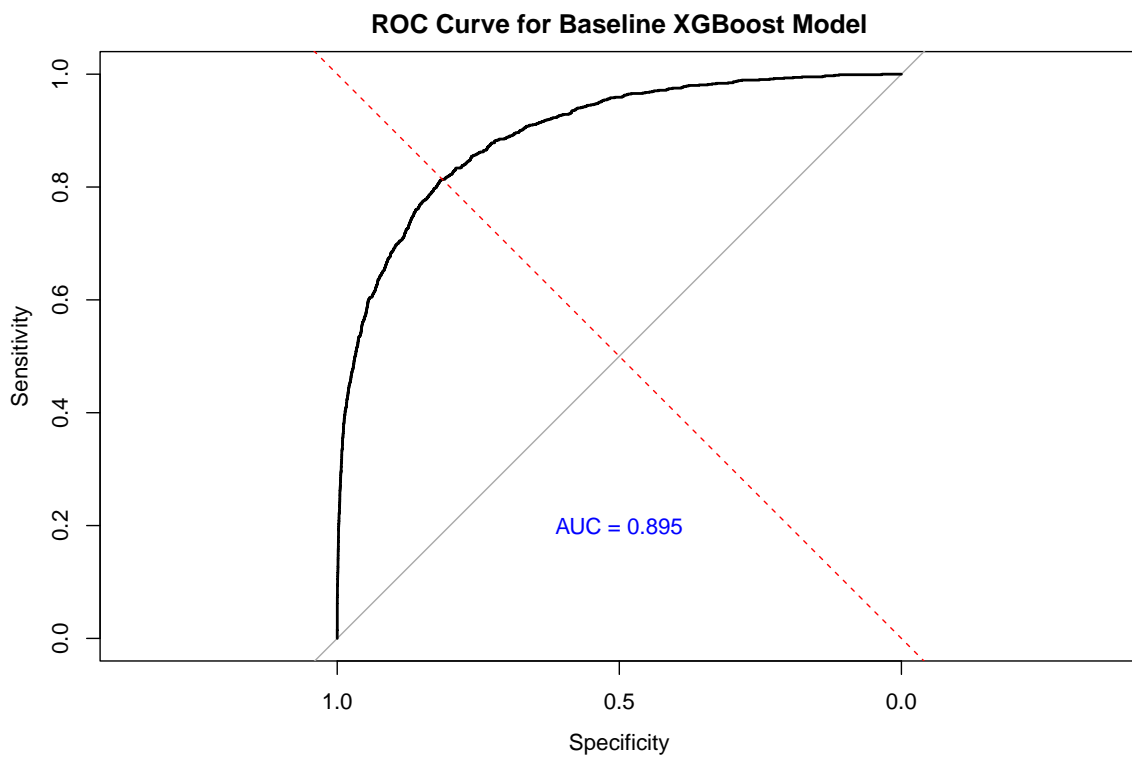


Figure 6: ROC curve for baseline XGBoost model

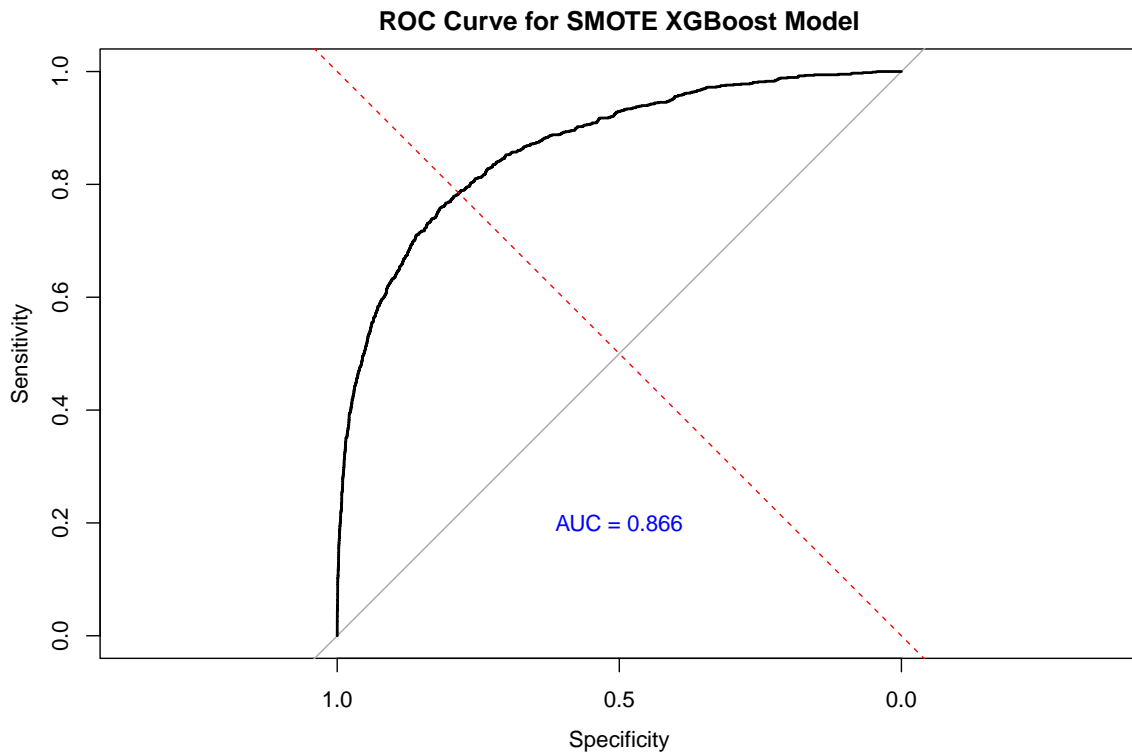


Figure 7: ROC curve for SMOTE XGBoost model

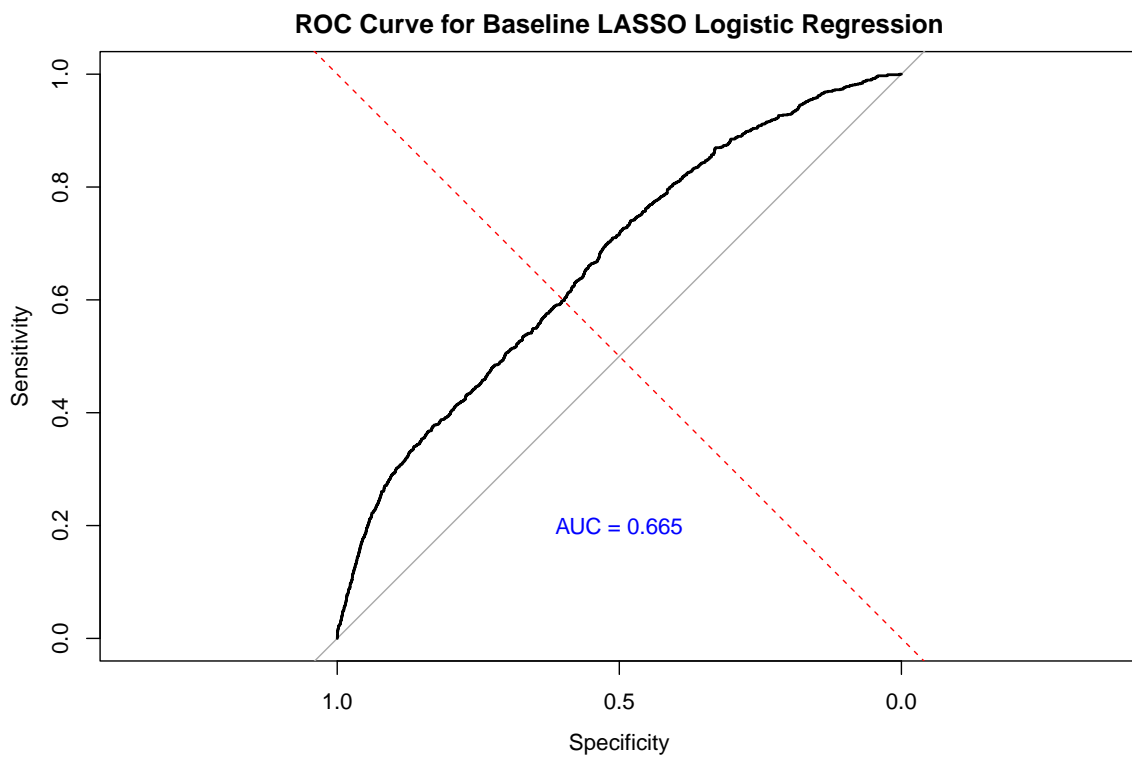


Figure 8: ROC curve for LASSO Logistic regression model

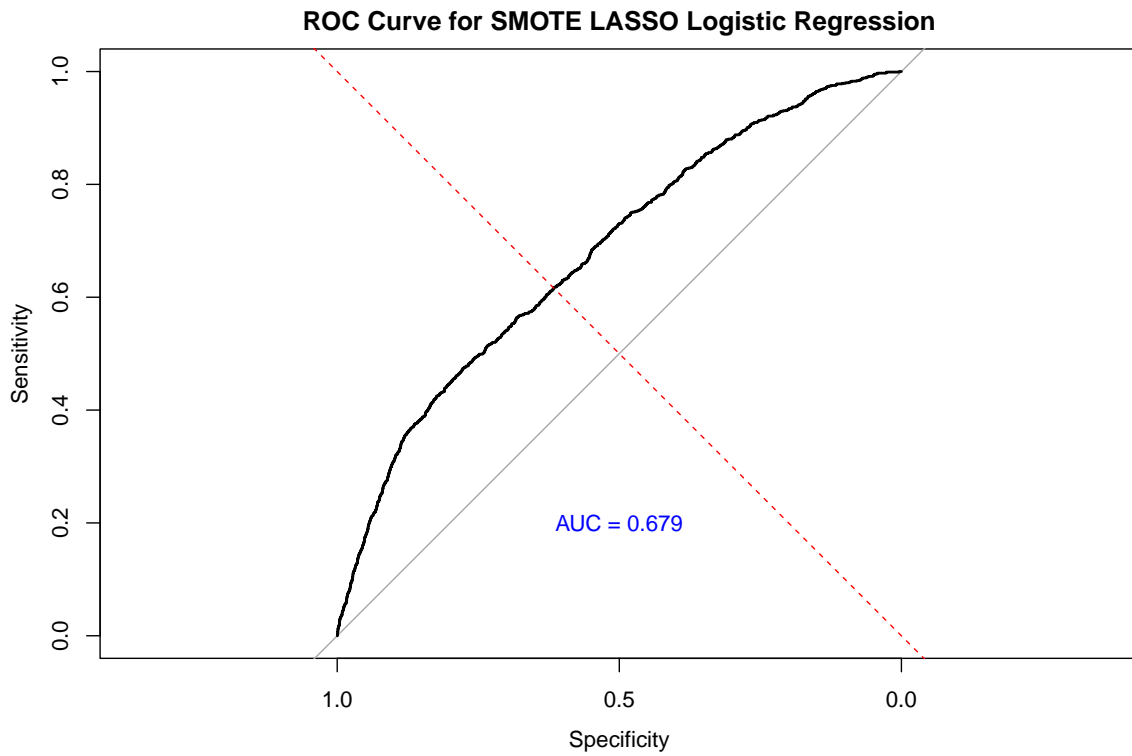


Figure 9: ROC curve for SMOTE LASSO Logistic regression model

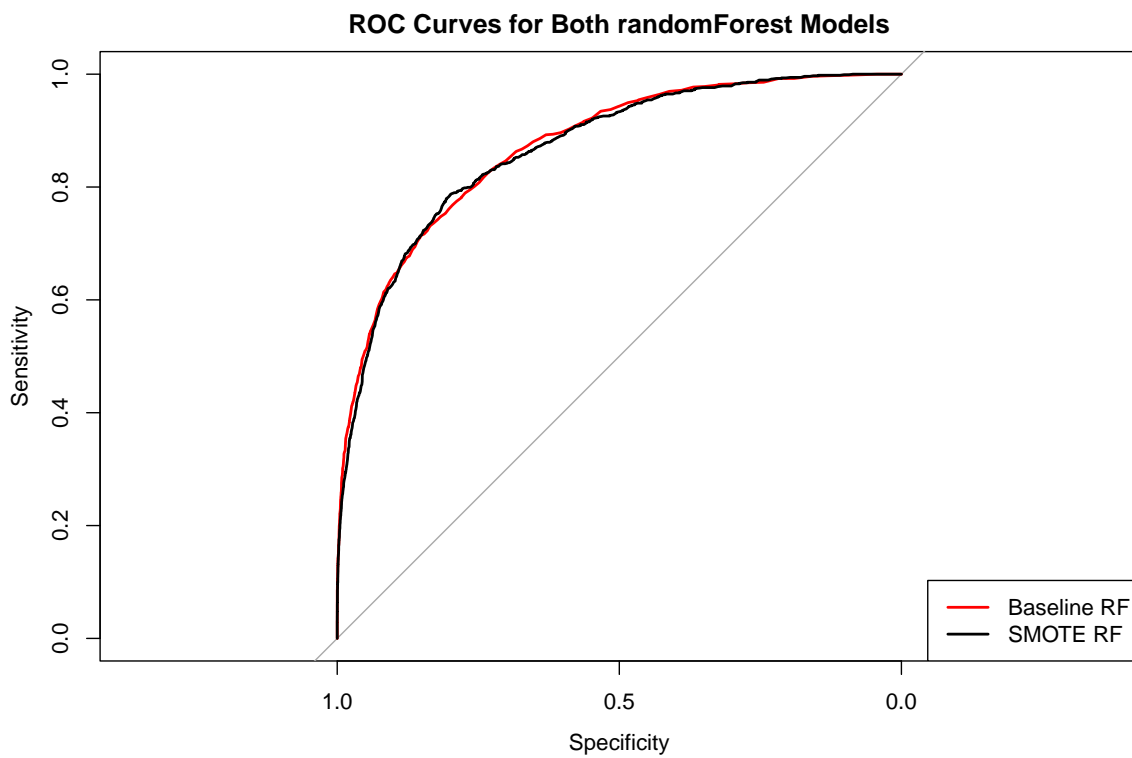


Figure 10: ROC curves for both randomForest models

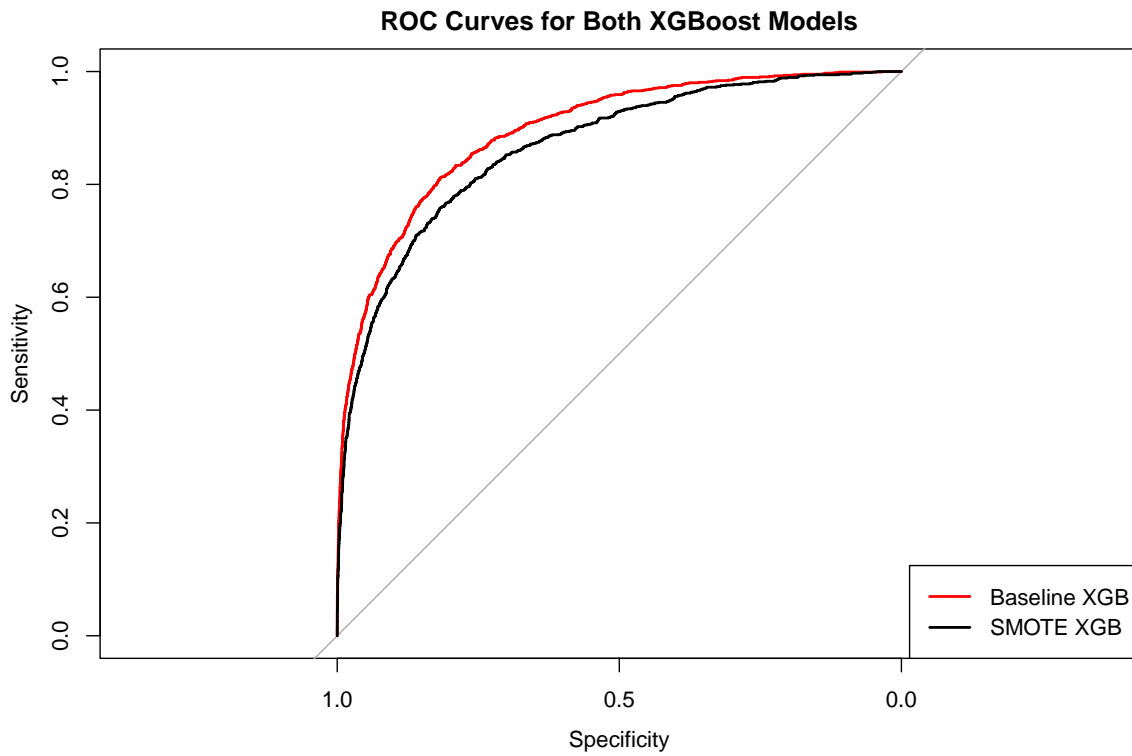


Figure 11: ROC curves for both XGBoost models

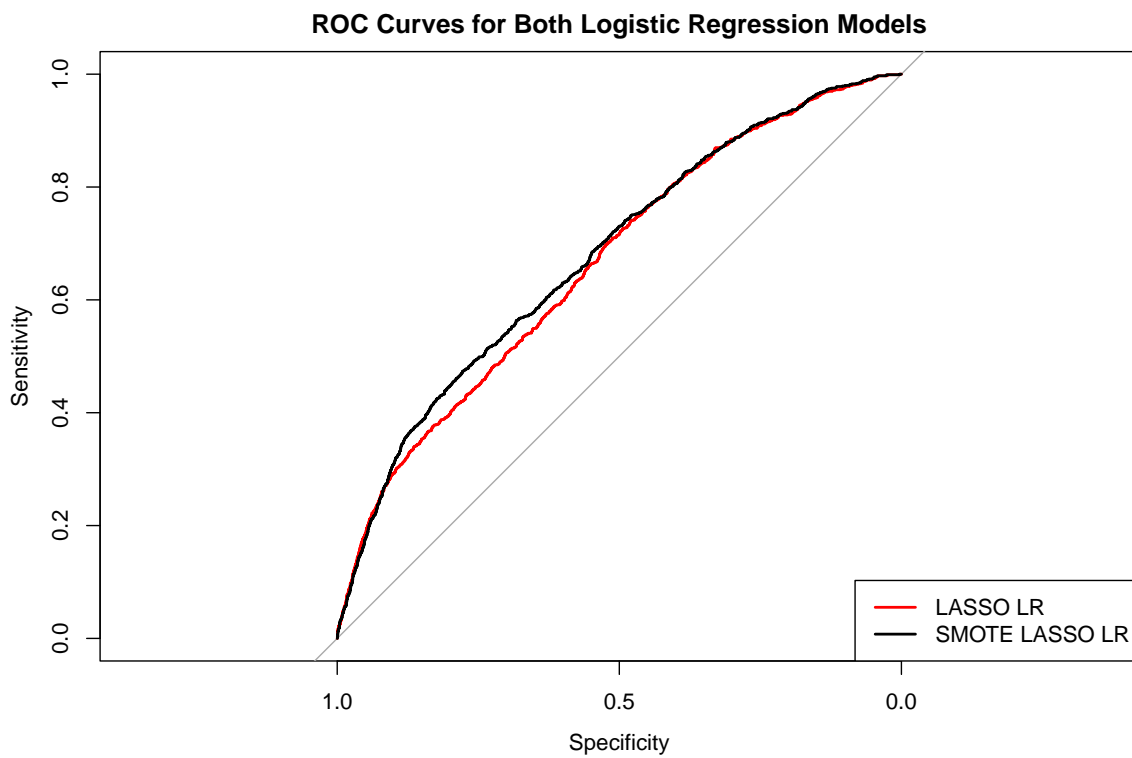


Figure 12: ROC curves for both logistic regression models

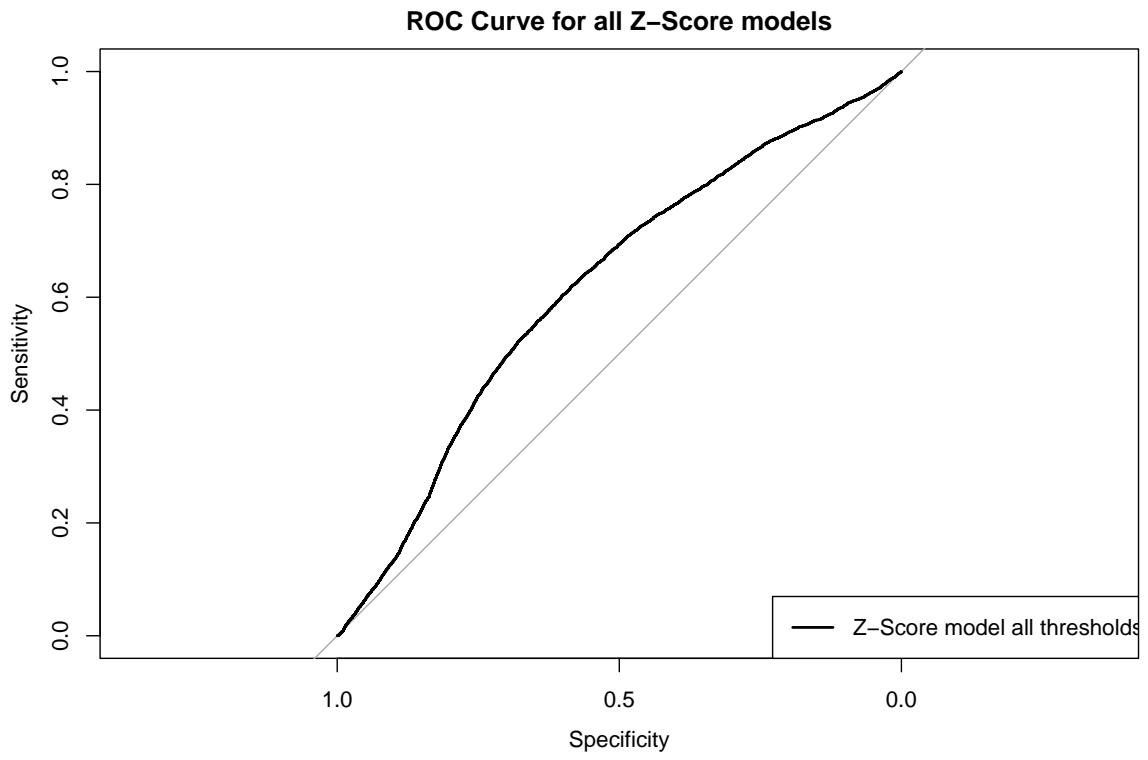


Figure 13: ROC curve for all Z-Score models