Bankruptcy prediction

Rabail Adwani

2022-12-14

Introduction to problem

We have taken a quest of identifying financial distressed listed companies that can go bankrupt to help financial institutions make appropriate lending decisions. The loans extended to corporations represent a significant amount of assets for banks. These loans are financed from deposits which are the liabilities for banks. An event of higher number of defaults, when banks lose the ability to settle their obligations, can lead to a run on the bank where many clients withdraw their money from the bank because they think it may cease its operations. Therefore it is essential to accurately assess risk before extending credit to avoid loss. Additionally, the early prediction would allow financial institutions sufficient time to take credit management measures to limit their economic losses.

Literature review

The insolvency of a company can have a negative impact on the overall economy. A great deal of studies have been conducted to predict bankruptcy to help reduce economic loss (Balleisen, 2001; Zywicki, 2008). Prior studies have identified that financial ratios are considered as one of the most significant predictors in predicting bankruptcy (Altman, 1968; Beaver, 1966; Ohlson, 1980). However, several studies have also been conducted identifying the importance of utilizing corporate governance indicators to predict bankruptcy (Bredart, 2014; Chen, 2014; Lee & Yeh, 2004; Lin, Liang, & Chu, 2010; Wu, 2007). The paper related to this dataset has focused on using financial ratios in association with corporate governance indicators to produce the best model that identifies financial distress. Using the concepts and theories learned in the classroom of Applied Statistics for Data Science. I would filter out unrepresentative features from the dataset to find the most significant financial features.

Dataset description:

The dataset was collected from the Taiwan Economic Journal for the years 1999-2009. Company bankruptcy was defined based on the business regulations of the Taiwan Stock Exchange. There were two criteria used to select companies. Firstly, the selected companies should have at least three years of public financial disclosure before the financial crisis. Secondly, there should be a sufficient number of comparable companies at a similar scale for comparison of the bankrupt and non-bankrupt cases. The final raw dataset has 6,819 rows with 95 predictors and a binary classified target indicating if the company can go bankrupt or not. Of the 95 predictors, two of the variables are categorical while 93 attributes are financial ratios of six different types including solvency, profitability, cash flow, capital structure, turnover, and others. The final sample of companies that we will be working on includes companies from the manufacturing industry comprising industrial and electronics companies (346), the service industry composed of shipping, tourism, and retail companies (39), and others (93), but not financial companies.

Target variable:

The target variable is the binary classification of bankruptcy based on the 95 predictor variables.

Variable information

- Y - Bankrupt?: Class label (Categorical)
- X1 - ROA(C) before interest and depreciation before interest: Return On Total Assets(C) (Numeric)
- X2 - ROA(A) before interest and % after tax: Return On Total Assets(A) (Numeric)
- X3 - ROA(B) before interest and depreciation after tax: Return On Total Assets(B) (Numeric)
- X4 - Operating Gross Margin: Gross Profit/Net Sales (Numeric)
- X5 - Realized Sales Gross Margin: Realized Gross Profit/Net Sales (Numeric)
- X6 - Operating Profit Rate: Operating Income/Net Sales (Numeric)
- X7 - Pre-tax net Interest Rate: Pre-Tax Income/Net Sales (Numeric)
- X8 - After-tax net Interest Rate: Net Income/Net Sales (Numeric)
- X9 - Non-industry income and expenditure/revenue: Net Non-operating Income Ratio (Numeric)
- X10 - Continuous interest rate (after tax): Net Income-Exclude Disposal Gain or Loss/Net Sales (Numeric)
- X11 - Operating Expense Rate: Operating Expenses/Net Sales (Numeric)
- X12 - Research and development expense rate: (Research and Development Expenses)/Net Sales (Numeric)
- X13 - Cash flow rate: Cash Flow from Operating/Current Liabilities (Numeric)
- X14 - Interest-bearing debt interest rate: Interest-bearing Debt/Equity (Numeric)
- X15 - Tax rate (A): Effective Tax Rate (Numeric)
- X16 - Net Value Per Share (B): Book Value Per Share(B) (Numeric)
- X17 - Net Value Per Share (A): Book Value Per Share(A) (Numeric)
- X18 - Net Value Per Share (C): Book Value Per Share(C) (Numeric)
- X19 - Persistent EPS in the Last Four Seasons: EPS-Net Income (Numeric)
- X20 - Cash Flow Per Share (Numeric)
- X21 - Revenue Per Share (Yuan ¥): Sales Per Share (Numeric)
- X22 - Operating Profit Per Share (Yuan ¥): Operating Income Per Share (Numeric)
- X23 - Per Share Net profit before tax (Yuan ¥): Pretax Income Per Share (Numeric)
- X24 - Realized Sales Gross Profit Growth Rate (Numeric)
- X25 - Operating Profit Growth Rate: Operating Income Growth (Numeric)
- X26 - After-tax Net Profit Growth Rate: Net Income Growth (Numeric)
- X27 - Regular Net Profit Growth Rate: Continuing Operating Income after Tax Growth (Numeric)
- X28 - Continuous Net Profit Growth Rate: Net Income-Excluding Disposal Gain or Loss Growth (Numeric)
- X29 - Total Asset Growth Rate: Total Asset Growth (Numeric)

- X30 - Net Value Growth Rate: Total Equity Growth (Numeric)
- X31 - Total Asset Return Growth Rate Ratio: Return on Total Asset Growth (Numeric)
- X32 - Cash Reinvestment %: Cash Reinvestment Ratio (Numeric)
- X33 - Current Ratio (Numeric)
- X34 - Quick Ratio: Acid Test (Numeric)
- X35 - Interest Expense Ratio: Interest Expenses/Total Revenue (Numeric)
- X36 - Total debt/Total net worth: Total Liability/Equity Ratio (Numeric)
- X37 - Debt ratio %: Liability/Total Assets (Numeric)
- X38 - Net worth/Assets: Equity/Total Assets (Numeric)
- X39 - Long-term fund suitability ratio (A): (Long-term Liability+Equity)/Fixed Assets (Numeric)
- X40 - Borrowing dependency: Cost of Interest-bearing Debt (Numeric)
- X41 - Contingent liabilities/Net worth: Contingent Liability/Equity (Numeric)
- X42 - Operating profit/Paid-in capital: Operating Income/Capital (Numeric)
- X43 - Net profit before tax/Paid-in capital: Pretax Income/Capital (Numeric)
- X44 - Inventory and accounts receivable/Net value: (Inventory+Accounts Receivables)/Equity (Numeric)
- X45 - Total Asset Turnover (Numeric)
- X46 - Accounts Receivable Turnover (Numeric)
- X47 - Average Collection Days: Days Receivable Outstanding (Numeric)
- X48 - Inventory Turnover Rate (times) (Numeric)
- X49 - Fixed Assets Turnover Frequency (Numeric)
- X50 - Net Worth Turnover Rate (times): Equity Turnover (Numeric)
- X51 - Revenue per person: Sales Per Employee (Numeric)
- X52 - Operating profit per person: Operation Income Per Employee (Numeric)
- X53 - Allocation rate per person: Fixed Assets Per Employee (Numeric)
- X54 - Working Capital to Total Assets (Numeric)
- X55 - Quick Assets/Total Assets (Numeric)
- X56 - Current Assets/Total Assets (Numeric)
- X57 - Cash/Total Assets (Numeric)
- X58 - Quick Assets/Current Liability (Numeric)
- X59 - Cash/Current Liability (Numeric)
- X60 - Current Liability to Assets (Numeric)
- X61 - Operating Funds to Liability (Numeric)
- X62 - Inventory/Working Capital (Numeric)
- X63 - Inventory/Current Liability (Numeric)
- X64 - Current Liabilities/Liability (Numeric)
- X65 - Working Capital/Equity (Numeric)

- X66 - Current Liabilities/Equity (Numeric)
- X67 - Long-term Liability to Current Assets (Numeric)
- X68 - Retained Earnings to Total Assets (Numeric)
- X69 - Total income/Total expense (Numeric)
- X70 - Total expense/Assets (Numeric)
- X71 - Current Asset Turnover Rate: Current Assets to Sales (Numeric)
- X72 - Quick Asset Turnover Rate: Quick Assets to Sales (Numeric)
- X73 - Working capitcal Turnover Rate: Working Capital to Sales (Numeric)
- X74 - Cash Turnover Rate: Cash to Sales (Numeric)
- X75 - Cash Flow to Sales (Numeric)
- X76 - Fixed Assets to Assets (Numeric)
- X77 - Current Liability to Liability (Numeric)
- X78 - Current Liability to Equity (Numeric)
- X79 - Equity to Long-term Liability (Numeric)
- X80 - Cash Flow to Total Assets (Numeric)
- X81 - Cash Flow to Liability (Numeric)
- X82 - CFO to Assets (Numeric)

- X83 - Cash Flow to Equity (Numeric)
- X84 - Current Liability to Current Assets (Numeric)
- X85 - Liability-Assets Flag: 1 if Total Liability exceeds Total Assets, 0 otherwise (Categorical)
- X86 - Net Income to Total Assets (Numeric)
- X87 - Total assets to GNP price (Numeric)
- X88 - No-credit Interval (Numeric)
- X89 - Gross Profit to Sales (Numeric)
- X90 - Net Income to Stockholder’s Equity (Numeric)
- X91 - Liability to Equity (Numeric)
- X92 - Degree of Financial Leverage (DFL) (Numeric)
- X93 - Interest Coverage Ratio (Interest expense to EBIT) (Numeric)
- X94 - Net Income Flag: 1 if Net Income is Negative for the last two years, 0 otherwise (Categorical)
- X95 - Equity to Liability (Numeric)

Research questions

The questions that we are aiming to answer include:

Q1. Will the company go bankrupt? Q2. What features or types of ratios are significant in predicting bankruptcy?

With these questions answered, we would be helping all the stakeholders of financial institutions (customers, shareholders, management, etc) to minimize the economic fallout of the event of bankruptcy.

Business value

The early prediction of bankruptcy not only helps the financial institutions make an appropriate lending decision but also forces the company itself to take corrective actions. With the identification of financial distress, the credit rating agencies take action accordingly and revise the credit risk rating of the company, which then aids investors in making better investment decisions and minimizing the loss. Moreover, it is also beneficial to identify the industry having the most bankrupt cases. This would provide a heads-up to the policymakers to address any systematic issues causing bankruptcies.

Statistical methods

Partitioning plan:

The classes of "Bankruptcy" are imbalanced thus, the aim is to utilize stratified sampling to create balanced training and testing set. However, the classifier model will not distinguish between 0 and 1 well because bankruptcy cases are just 3%. Therefore, we will be under sampling the training data to take the proportion of bankruptcy cases of 10% versus 3% in the original dataset. Meanwhile, the testing set would reflect the proportion of bankruptcy classes similar to the actual data to reflect real-word scenario for test model comparison.

Feature selection:

The aim of the feature selection is to discard the redundant predictor variables to keep the model as simple as possible. We will be using stepwise logistic regression (SLR) to select the most significant variables. Other algorithms selected also help determine which variables are most important e.g. Gini Index in Decision tree.

Modeling work:

There are many models and techniques that can be leveraged to develop predictive models of bankruptcy. Within the coursework of Applied Statistics for Data Science, we will be using four classification techniques namely Logistic regression, Decision tree, Random forest, and Gradient boosting.

The rationale behind choosing logistic regression is that it is easier to implement, interpret, and very efficient to train. It makes no assumptions about distributions of classes of the features. Additionally, it has good explainability because it provides coefficient size with its direction (positive or negative). As for the decision tree, it has built-in variable selection mechanism and has no assumptions of linear relationship thus, it is easy to use. Importantly, it is a white-box model which means if a given situation is observable in a model, the explanation for the condition is easily explained by Boolean logic. With regards to random forest and xgboost, they are known to perform well in high dimensional settings without being subject to overfitting because they are ensembles. Additionally, They are widely used in the banking industry to predict creditworthiness of a loan applicant.

#Renaming the target:

In the raw dataset, the target is named "bankrupt?". Since we will be using the response variable in our script many times, it is important to give it a name without a question mark. Therefore, we renamed it to "Bankruptcy".

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
library(rpart)
library(rpart.plot)
```

```
## Warning: package 'rpart.plot' was built under R version 4.2.2
```

```
library(ranger)
```

```
## Warning: package 'ranger' was built under R version 4.2.2
```

```
library(vip)
```

```
## Warning: package 'vip' was built under R version 4.2.2
```

```
##
## Attaching package: 'vip'
```

```
## The following object is masked from 'package:utils':
##
##     vi
```

```
library(xgboost)
```

```
## Warning: package 'xgboost' was built under R version 4.2.2
```

```
library(Matrix)
library(glmnet)
```

```
## Loaded glmnet 4.1-4
```

```
library(ROSE)
```

```
## Warning: package 'ROSE' was built under R version 4.2.2
```

```
## Loaded ROSE 0.0-4
```

```
library(ggplot2)
library(reshape2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following object is masked from 'package:xgboost':
##
##     slice

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(broom)
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

```
setwd("F:/MSDS/Applied Statistics for Data Science")
co.data <- read.csv("Data/bankruptcy.csv")
colnames(co.data)[1]="Bankruptcy"
```

Encoding categorical variables:

The dataset has only three categorical variables including Bankruptcy (the target variable), Liability Assets Flag, and Net Income Flag. We will be encoding those as factor variables.

```
cat.cols <- c("Bankruptcy", "Liability.Assets.Flag", "Net.Income.Flag")
num.cols <- which((lapply(co.data, class))=="numeric")
co.data[,cat.cols] <- lapply(co.data[,cat.cols], factor)
```

Dealing with missing values:

With a quick check of NAs, we can see that the data has zero NAs. However, upon diving deeper into the dataset, we found that some of the attributes have many values of exactly zero, which do not make sense e.g. tax rate cannot be zero. Therefore, we removed the Tax rate and long-term liability to current assets, which had values of exactly zero of 16% and 68%, respectively.

```
table(is.na(co.data))
```

```
##
##  FALSE
## 654624
```

```
co.data[,num.cols][co.data[,num.cols]==0] <- NA
table(is.na(co.data))
```

```
##
##  FALSE   TRUE
## 646819   7805
```

```
which(colMeans(is.na(co.data))>0.1)
```

```
## Research.and.development.expense.rate   Interest.bearing.debt.interest.rate
##                                    13                                    15
##                          Tax.rate..A. Long.term.Liability.to.Current.Assets
##                                    16                                    68
```

```
co.data.t <- subset(co.data, select=-c(Tax.rate..A.,
                                       Long.term.Liability.to.Current.Assets))
```

Unique classes in categorical variables:

Upon checking the unique classes of the two other categorical variables except for target including Liability Assets Flag and Net Income Flag, we found that the Net Income Flag only had one unique label while Liability Assets Flag had 8 ones and 6,811 zeroes. Hence, we will be removing these as they will not benefit our analysis at all.

```
lapply(co.data[,cat.cols], unique)
```

```
## $Bankruptcy
## [1] 1 0
## Levels: 0 1
##
## $Liability.Assets.Flag
## [1] 0 1
## Levels: 0 1
##
## $Net.Income.Flag
## [1] 1
## Levels: 1
```

```
table(co.data$Liability.Assets.Flag)
```

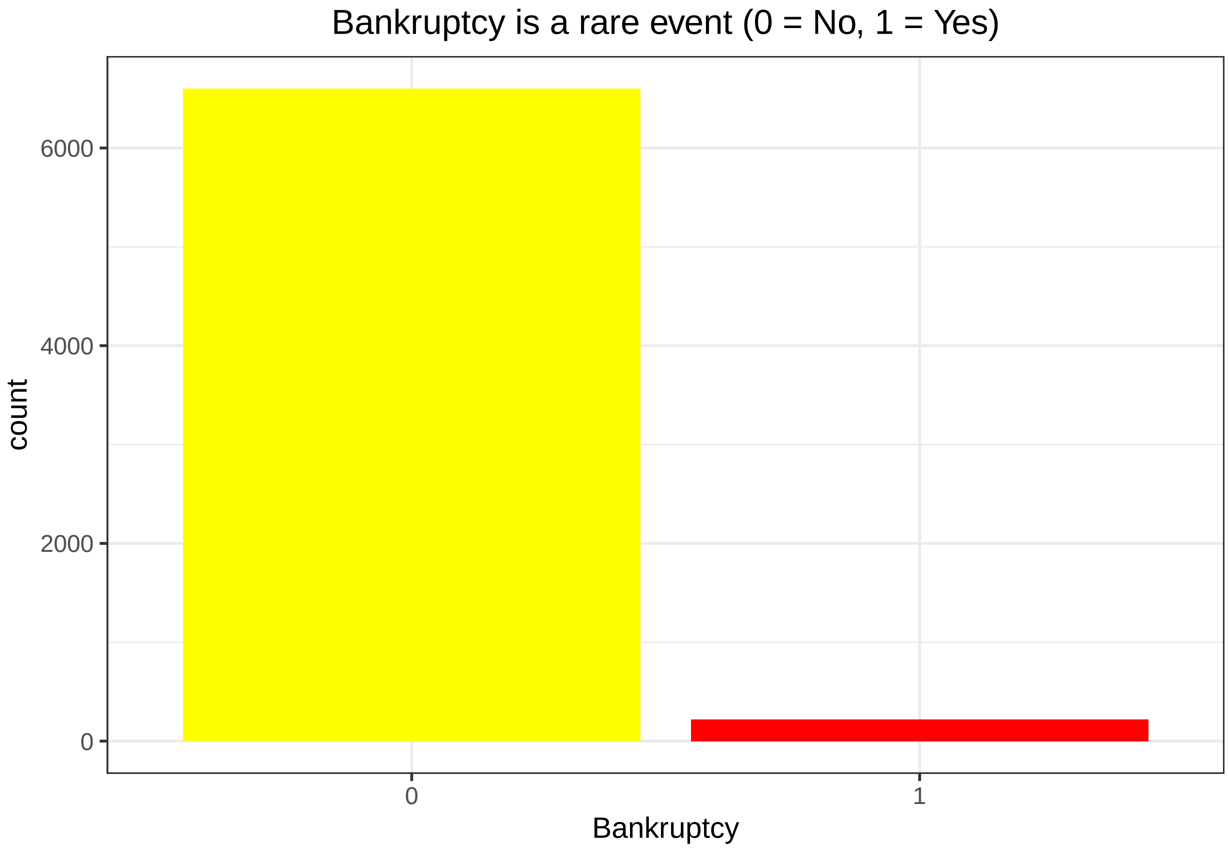
```
##
##    0    1
## 6811    8
```

```
bankrupt.data <- subset(co.data.t, select=-c(Net.Income.Flag, Liability.Assets.Flag))
num.cols2 <- which((lapply(bankrupt.data, class))=="numeric")
bankrupt.data[,num.cols2][is.na(bankrupt.data[,num.cols2])]=0
```

Bankrupt cases are rare:

Bankrupt cases are quite rare. 97% of the cases in the dataset are classified as non-bankrupt, while about 3% of the time, companies classify as bankrupt.

```
ggplot(bankrupt.data, aes(Bankruptcy,)) + geom_bar(fill=c("Yellow","Red")) + theme_bw()+ggtitle("Bankru
theme(plot.title = element_text(hjust = 0.5))
```



# Partitioning the data into training and testing set:

Since bankrupt cases are rare in the dataset. We decided to create balanced samples of training and testing sets to ensure that the model is trained on distributions representing the real-world case. However, the dataset is very imbalanced to train a useful model. The classifier was not able to distinguish well between 0 and 1. Therefore, we also did under-sampling for the training set to keep the proportion of bankrupt cases at 10% and non-bankrupt at 90% to train the model.

```r
# Training and testing split in 80-20
set.seed(123457)
train.prop <- 0.80
strats <- bankrupt.data$Bankruptcy
rr <- split(1:length(strats), strats)
idx <- sort(as.numeric(unlist(sapply(rr, function(x) sample(x, length(x)*train.prop )))))
bankrupt.data.train <- bankrupt.data[idx, ]
bankrupt.data.test <- bankrupt.data[-idx, ]

# Bankrupt cases are really rate at about 3%
# Therefore, we will under-sample non-bankrupt cases to train the model
under_sample <- ovun.sample(Bankruptcy ~ ., data=bankrupt.data.train,
                            method='under', p=0.1)$data
summary(under_sample$Bankruptcy)/nrow(under_sample)
```

```
##         0         1
## 0.8996007 0.1003993
```

```r
summary(bankrupt.data.test$Bankruptcy)/nrow(bankrupt.data.test)
```

```
##          0          1
## 0.96774194 0.03225806
```

Filtering variable to see if model converges

Upon running the model with all 91 predictor variables, we received a warning that the fitted probabilities of one or more observations in the dataset are indistinguishable from 0 or 1. After a careful analysis, we filtered out variables manually to see if it converges. Detailed code can be found in the R script file.

Check for multicollinearity

We also took care of the curse of the dimensionality that is multicollinearity. Seven of the variables which were highly correlated (>0.95) were discarded. However, we also checked VIFs of the full model. There were five variables that had high VIFs that were also removed. Detailed code can be found in the R scipt file.

After the data cleaning part, we were left with 41 predictor variables in the end. A large number of variables were excluded from analysis due to the convergence issue.

```r
model.full <- glm(Bankruptcy~ ., data=under_sample, family=binomial("probit"))
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```r
pdv <- c(1, 3, 7, 12, 13, 14, 15, 20, 21, 26, 28, 29, 30, 31, 34, 35, 36, 37,
          39, 40, 41, 42, 45, 46, 55, 56, 58, 59, 60, 61, 62, 63, 69, 70, 71, 73,
          74, 75, 78, 79, 81, 83)
under_sample46 <- under_sample[,pdv]
bankrupt.data.test <- bankrupt.data.test[,pdv]
```

Logistic regression using forward Stepwise selection

We utilized Stepwise Regression to reduce the computing complexity of best subsets regression by sequentially selecting variables. It uses partial F-statistics and associated p-values to decide whether to include or exclude each of the p predictors into the model. By using forward selection, we ended up with 11 variables that are significant in predicting bankruptcy. They include;

1. ROA.A..before.interest.and…after.tax
2. Operating.Profit.Rate
3. Continuous.Net.Profit.Growth.Rate
4. Interest.Expense.Ratio
5. Total.debt.Total.net.worth
6. Debt.ratio
7. Total.Asset.Turnover
8. Accounts.Receivable.Turnover
9. Quick.Assets.Current.Liability
10. Inventory.Working.Capital
11. Cash.Turnover.Rate

Evaluation

The model achieved a total accuracy rate of 96% and missclassification rate of 4%. Meanwhile, the sensitivity, specificity, and area under the curve stood at 98%, 36%, and 0.92, respectively. Logistic regression has the highest AUC among all the other models that have been tested.

Formula

P(Bankruptcy=1) = Phi coefficient(-1.329e+02 - 8.335e+00 (ROA.A..before.interest.and…after.tax)+ 1.565e+02(Operating.Profit.Rate) - 3.683e+01(Continuous.Net.Profit.Growth.Rate) - 1.171133e+01(Interest.Expense.Ratio) + 1.258534e-09(Total.debt.Total.net.worth) + 1.497397e+01(Debt.ratio..) - 3.459563e+00(Total.Asset.Turnover) - 5.763139e-10(Accounts.Receivable.Turnover) - 8.739555e-10(Quick.Assets.Current.Liability) - 2.420958e+01 (Inventory.Working.Capital) - 4.326824e-11(Cash.Turnover.Rate) + epsilon

```r
model.full <- glm(Bankruptcy~ ., data=under_sample46,family=binomial("probit"))
model.null <- glm(Bankruptcy~1, data=under_sample46,family=binomial("probit"))

pred.step <- c(2, 3, 11, 16, 17, 18, 23, 24, 27, 31, 36)
df.step1 <- under_sample46[,c(1,pred.step)]

model.step <- glm(formula = Bankruptcy ~ ., family = binomial(link = "probit"),
    data = df.step1)

predict.test.step <- predict(model.step, newdata=bankrupt.data.test,
                             type="response")
(table.test.step <- table(bankrupt.data.test$Bankruptcy,
                          ifelse(predict.test.step>0.5,1,0)))
```

```
##
##        0    1
##   0 1295   25
##   1   30   14
```

```
(accuracy.step.test <- round((sum(diag(table.test.step))/sum(table.test.step))*100,2))
```

```
## [1] 95.97
```

```
sensitivity(table.test.step)
```

```
## [1] 0.9773585
```

```
specificity(table.test.step)
```

```
## [1] 0.3589744
```

```
100 - accuracy.step.test
```

```
## [1] 4.03
```

```
roc.test <- roc(bankrupt.data.test$Bankruptcy, predict.test.step, levels=c(1,0))
```

```
## Setting direction: controls > cases
```

```
auc(bankrupt.data.test$Bankruptcy, predict.test.step)
```

```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

## Area under the curve: 0.9183
```

Decision tree (Pruned)

A decision tree divides the data into different classes. It uses an algorithm to select features and create split points until a predetermined termination criterion is reached and a suitable tree is constructed. The tree repeatedly splits a node into two child nodes.

With the decision tree using all 41 filtered predictors, we found that the smallest value of xerror is 0.818 at a Cp value of 0.0227. To make the decision tree simpler, we pruned at the Cp value where the xerror is minimum.

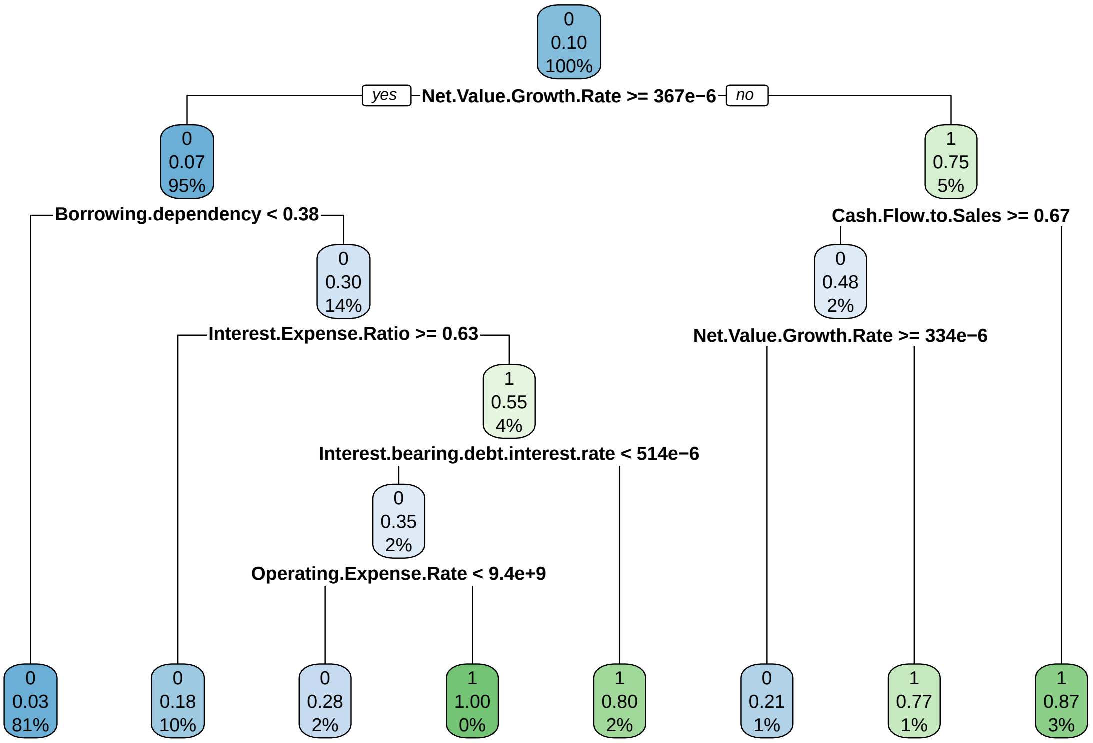
Evaluation

The model achieved a total accuracy rate of 97% and missclassification rate of 3%. Meanwhile, the sensitivity, specificity, and area under the curve stood at 98%, 49%, and 0.70, respectively. In comparison with other models, the decision tree has lower AUC. The specificity is higher than the logistic regression and random forest. It is important to have higher specificity as it indicates that there are fewer false positive cases. At root node of the decision tree is the net value growth rate followed by the leaf nodes of borrowing dependency, cash flow to sales, and others.

```r
fit.allp <- rpart(Bankruptcy~.,method="class", data=under_sample46,
                  control=rpart.control(minsplit=1, cp=0.001))

pfit.allp <- prune(fit.allp, cp=0.02272727)

rpart.plot(pfit.allp, extra = "auto")
```


0
0.10
100%
yes
Net.Value.Growth.Rate >= 367e−6
no
0
0.07
95%
1
0.75
5%
Borrowing.dependency < 0.38
Cash.Flow.to.Sales >= 0.67
0
0.30
14%
0
0.48
2%
Interest.Expense.Ratio >= 0.63
Net.Value.Growth.Rate >= 334e−6
1
0.55
4%
Interest.bearing.debt.interest.rate < 514e−6
0
0.35
2%
Operating.Expense.Rate < 9.4e+9
0
0.03
81%
0
0.18
10%
0
0.28
2%
1
1.00
0%
1
0.80
2%
0
0.21
1%
1
0.77
1%
1
0.87
3%

```r
test_df1 <- data.frame(actual=bankrupt.data.test$Bankruptcy, pred=NA)
test_df1$pred <- predict(pfit.allp, newdata=bankrupt.data.test, type="class")

(test_conf_matrix_pruned <- table(test_df1$actual,test_df1$pred))
```

```
##
##         0    1
##    0 1301   19
##    1   26   18
```

```r
sensitivity(test_conf_matrix_pruned)
```

```
## [1] 0.9804069
```

```r
specificity(test_conf_matrix_pruned)
```

```
## [1] 0.4864865
```

```
(test_conf_matrix_pruned[1,2] + test_conf_matrix_pruned[2,1])/sum(test_conf_matrix_pruned)
```

```
## [1] 0.0329912
```

```
round((test_conf_matrix_pruned[1,1]+test_conf_matrix_pruned[2,2])/sum(test_conf_matrix_pruned)*100,2)
```

```
## [1] 96.7
```

```
roc.test2 <- roc(bankrupt.data.test$Bankruptcy, as.numeric(test_df1$pred), levels=c(1,0))
```

```
## Setting direction: controls < cases
```

```
auc(bankrupt.data.test$Bankruptcy, as.numeric(test_df1$pred))
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
## Area under the curve: 0.6973
```

Random Forest

Random forest is an ensemble learning method that combines the output of multiple decision trees to reach a single result to avoid overfitting. The word "Random" implies the selection of predictors at random while "Forest" tells us that output from different trees is used to make a decision. There are two types of randomness that go into this algorithm 1) random selection of sample to build each new tree and 2) random selection of features at each tree node to create best split.

Below, we have built a random forest with 41 predictors on a under sample that we had created. Looking at the variable importance plot, we can derive that the Net Value Growth Rate is most important variable followed by Borrowing dependency, Debt ratio, and others.

Evaluation

The model achieved a total accuracy rate of 96% and missclassification rate of 4%. Meanwhile, the sensitivity, specificity, and area under the curve stood at 98%, 44%, and 0.67, respectively. Although the total accuracy rate and missclassification rate are similar to logistic regression, it has lower AUC. Meanwhile, the specificity is also lower than decision tree.

```
num.pred <- 41
(mtry.1 <- floor(sqrt(num.pred)))
```

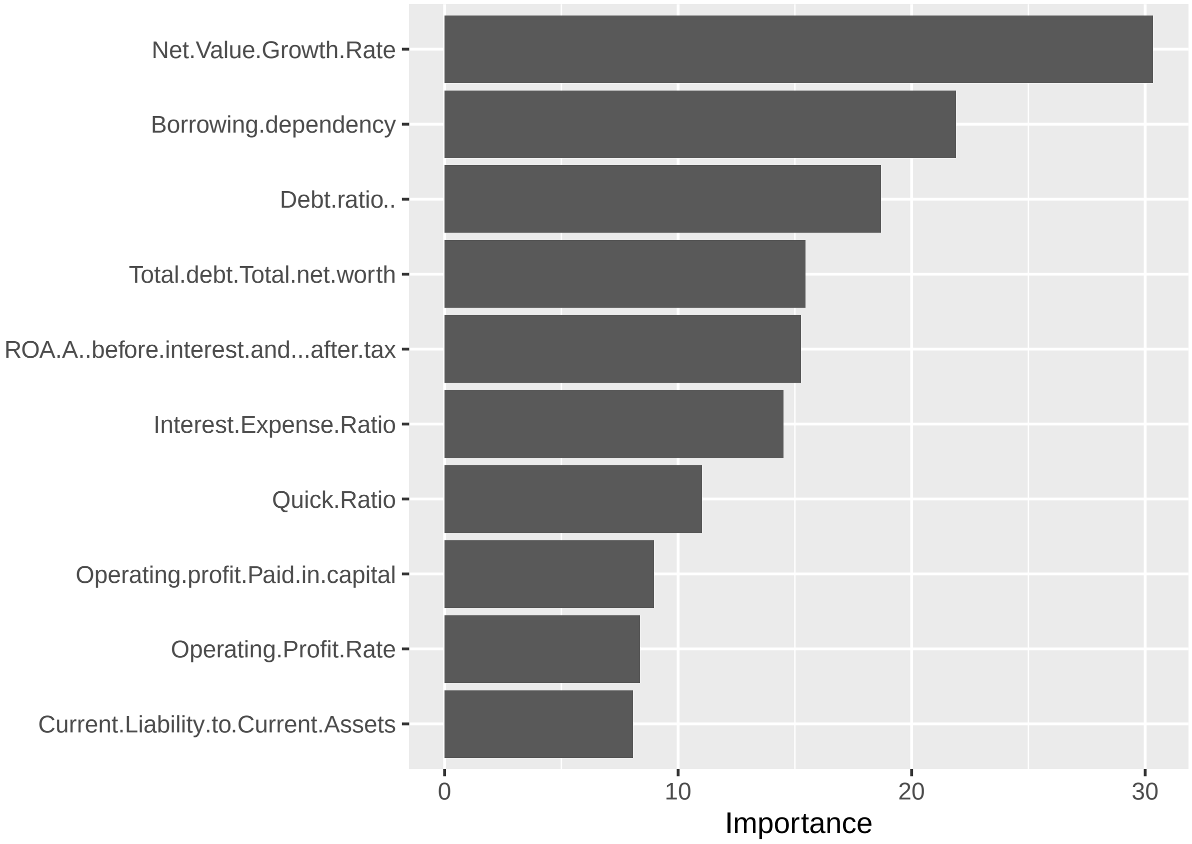
```
## [1] 6
```

```
fit.rf.ranger <- ranger(Bankruptcy ~ ., data=under_sample46,
                        importance='impurity', mtry=mtry.1)

(default_rmse <- sqrt(fit.rf.ranger$prediction.error))
```

```
## [1] 0.2616373
```

```
vip(fit.rf.ranger)    # plot the variable importance as a bar graph
```



```
test_df2 <- data.frame(actual=bankrupt.data.test$Bankruptcy, pred=NA)
predict.test2 <- predict(fit.rf.ranger, data=bankrupt.data.test)
test_df2$pred <- predict.test2$predictions

(test_conf_matrix_rf <- table(test_df2$actual, test_df2$pred))
```

```
##
##          0    1
##    0 1298   22
##    1   28   16
```

```
sensitivity(test_conf_matrix_rf)
```

```
## [1] 0.9788839
```

```
specificity(test_conf_matrix_rf)
```

```
## [1] 0.4210526
```

```
(test_conf_matrix_rf[1,2] + test_conf_matrix_rf[2,1])/sum(test_conf_matrix_rf)
```

```
## [1] 0.03665689
```

```
round((1 - (test_conf_matrix_rf[1,2] + test_conf_matrix_rf[2,1])/sum(test_conf_matrix_rf)),2)
```

```
## [1] 0.96
```

```
roc.test3 <- roc(bankrupt.data.test$Bankruptcy, as.numeric(test_df2$pred), levels=c(1,0))
```

```
## Setting direction: controls < cases
```

```
auc(bankrupt.data.test$Bankruptcy, as.numeric(test_df2$pred))
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
## Area under the curve: 0.6735
```

Gradient boosting

Gradient boosting is known for its good predictive performance in high dimensional datasets. Unlike random forest which creates an ensemble of independent deep trees, gradient boosting creates an ensemble for shallow trees where each tree learns from the previous tree minimizing the overall prediction error. A combination of these shallow trees provides a highly predictive algorithm. It's popularity has increased intensively with increased performance in various Kaggle competitions.

Evaluation

The model achieved a total accuracy rate of 97% and missclassification rate of 3%. Meanwhile, the sensitivity, specificity, and area under the curve stood at 98%, 50%, and 0.7, respectively. In comparison with other models, the total accuracy rate is significant at 97% which means the missclassification rate is lower at 3%. The sensitivity for all the models is roughly the same at 98%. However, in terms of specificity, gradient boosting significantly outperforms other models as it stands at 50%. This indicates that with gradient boosting, we have fewer false positive cases. The AUC is second highest for this model at 0.7. It is second highest because it is lower than AUC for logistic regression which stands at 0.92. The important features in xgboost include net value growth rate, borrowing dependency, quick ratio, and others.

```
matrix_predictors.train <- as.matrix(sparse.model.matrix(Bankruptcy ~ .,
                                                          data=under_sample46))[,-1]
matrix_predictors.test <- as.matrix(sparse.model.matrix(Bankruptcy ~ .,
                                                         data = bankrupt.data.test))[,-1]

pred.train.gbm <- data.matrix(matrix_predictors.train)
bankrupt.data.train.gbm <- as.numeric(as.character(under_sample46$Bankruptcy))
dtrain <- xgb.DMatrix(data=pred.train.gbm, label=bankrupt.data.train.gbm)
```

```
pred.test.gbm <- data.matrix(matrix_predictors.test)
bankrupt.data.test.gbm <- as.numeric(as.character(bankrupt.data.test$Bankruptcy))
dtest <- xgb.DMatrix(data=pred.test.gbm, label=bankrupt.data.test.gbm)

watchlist <- list(train=dtrain, test=dtest)

param <- list(max_depth = 2, eta = 1, verbose = 0, nthread = 2,
              objective = "binary:logistic", eval_metric = "auc")

model.xgb <- xgb.train(param, dtrain, nrounds=4, watchlist, verbose=F, silent=T)
```

```
## [20:50:56] WARNING: amalgamation/../src/learner.cc:627:
## Parameters: { "silent", "verbose" } might not be used.
##
##   This could be a false alarm, with some parameters getting used by language bindings but
##   then being mistakenly passed down to XGBoost core, or some parameter actually being used
##   but getting flagged wrongly here. Please open an issue if you find any such cases.
```

```
predict.bankruptcy.test <- predict(model.xgb, pred.test.gbm)
prediction.test.xgb <- as.numeric(predict.bankruptcy.test>0.5)

test_conf_matrix_xgb <- table(bankrupt.data.test.gbm, prediction.test.xgb)

sensitivity(test_conf_matrix_xgb)
```

```
## [1] 0.9804217
```

```
specificity(test_conf_matrix_xgb)
```

```
## [1] 0.5
```

```
round(1 - ((sum(diag(test_conf_matrix_xgb)))/sum(test_conf_matrix_xgb)),3)
```

```
## [1] 0.032
```

```
round((sum(diag(test_conf_matrix_xgb))/sum(test_conf_matrix_xgb)),2)
```

```
## [1] 0.97
```

```
roc.test4 <- roc(bankrupt.data.test$Bankruptcy, prediction.test.xgb, levels=c(1,0))
```

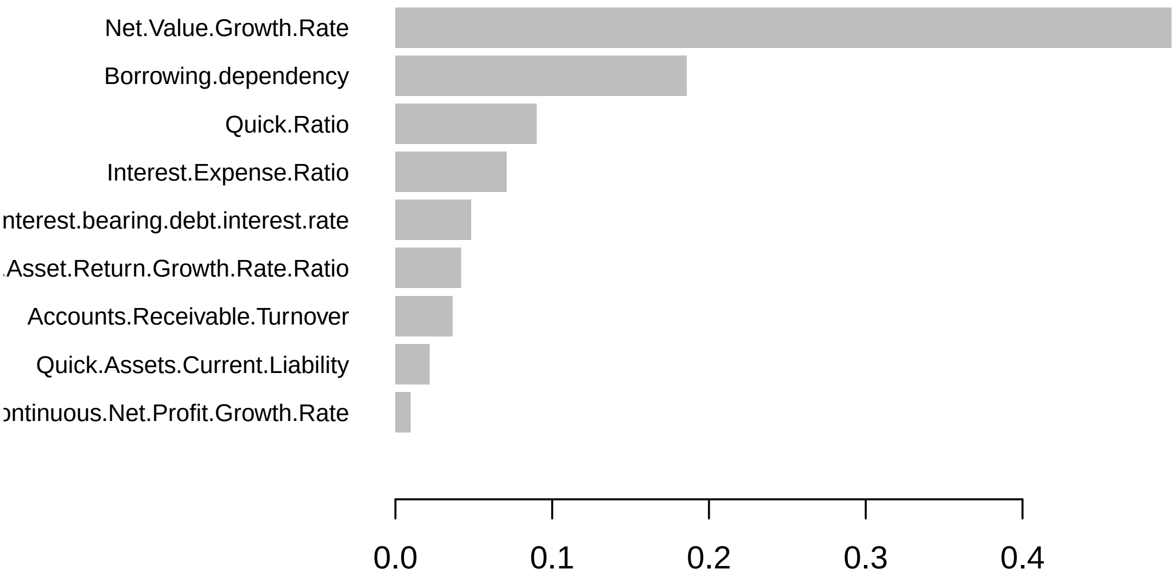
```
## Setting direction: controls < cases
```

```
auc(bankrupt.data.test$Bankruptcy, prediction.test.xgb)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
## Area under the curve: 0.6977
```

```
importance_matrix = xgb.importance(colnames(under_sample46[,2:42]), model=model.xgb)
xgb.plot.importance(importance_matrix[1:10,])
```



Conclusion

The project focuses on predicting bankruptcy accurately using the financial ratios of six different types including solvency, profitability, cash flow, capital structure, turnover, and others of Taiwanese companies for the years 1999-2009. To predict bankruptcy and identify the most important financial ratios, we have utilized four prediction models including logistic regression, decision tree, random forest, and gradient boosting. In addition, we have also used forward stepwise selection to select significant variables.

Broadly, the most important features in predicting bankruptcy include growth ratios (e.g. net value growth or total equity growth), solvency ratios (e.g. borrowing dependency), and profitability ratios (e.g. Asset return growth rate). As for the model performance, the gradient boosting algorithm performs the best. It outperforms other models as its specificity stands at 50%. This indicates that with gradient boosting, we have fewer false positive cases. The AUC is second highest for this model at 0.7. It is second highest because it is lower than AUC for logistic regression which stands at 0.92.

In the future, it would be interesting to combine these financial ratios with Environmental, Social, and Governance (ESG) data. The latter reflects on the negative externalities caused by an organization with respect to the environment. It can be used to assess the material risk that an organization undertakes. The combination of financial and non-financial (ESG) indicators would help assess systematic and unsystematic risk of the companies.

References

Balleisen, E. (2001). Navigating failure: Bankruptcy and commercial society in Antebel- lum America . Chapel Hill: University of North Carolina Press .

Zywicki, T. J. (2008). Bankruptcy. In D. R. Henderson (Ed.), Concise encyclopedia of economics (2nd ed.). Indianapolis: Library of Economics and Liberty .

Altman, E. I. (1968). Financial ratios, discriminant analysis and the prediction cor- porate bankruptcy. Journal of Finance, 23 (4), 589–609 .

Beaver, W. H. (1966). Financial ratios predictors of failure. Journal of Accounting Re- search, 4 , 71–111 .

Ohlson, J. A. (1980). Financial ratios and the probabilistic prediction of bankruptcy. Journal of Accounting Research, 18 , 109–131 .

Bredart, 2014; Chen, 2014; Lee & Yeh, 2004; Lin, Liang, & Chu, 2010; Wu, 2007).

Chen, I.-J. (2014). Financial crisis and the dynamics of corporate governance: Evi- dence from Taiwan's listed firms. International Review of Economics and Finance, 32 , 3–28 .

Lee, T.-S. , & Yeh, Y. H. (2004). Corporate governance and financial distress: Evidence from Taiwan. Corporate Governance: An International Review, 12 (3), 378–388 .

Lin, F.-Y. , Liang, D. , & Chu, W.-S. (2010). The role of non-financial features related to corporate governance in business crisis prediction. Journal of Marine Science and Technology, 18 (4), 504–513 .

Wu, J.-L. (2007). Do corporate governance factors matter for financial distress predic- tion of firms? Evidence from Taiwan Master's thesis. University of Nottingham .