Comparative Analysis of Classification Methods for Bankruptcy Prediction

Rong Duan

Department of Biostatistics, University of California.

Abstract

With the completion of the bankruptcy clause, bankruptcy procedure are more likely to protect business owners, however, it still has a lot of deficiencies which can't be omitted. The dataset of this analysis is from a Korean bank and consists of 250 cases. There are 6 predictors which are all categorical with three levels: "average", "negative", and "positive". Three classification methods are applied to predict bankruptcy. The models are built with predictors selected from the logistic regression model. 10-fold cross-validation and leave-one-out cross-validation are utilized to compare the prediction ability of the three models. Among all the risky features, negative financial flexibility, Negative management risk, Negative credibility, and Negative operating risk are the most important situations that need to be avoided, and the positive credibility level should be achieved in the best effort.

Key words: bankruptcy prediction, classification, logistic regression, probit regression, random forest

1 Introduction

Bankruptcy is a legal process through which people or other entities cannot repay debts to creditors and may seek relief from their debts [1]. A bankruptcy case normally begins with a petition from the debtor in federal courts [2]. Although the bankruptcy can immediately protect the company from creditors seeking to collect on a debt [3], bankruptcy is a signal of overall unsatisfactory operation, which may further discourage the investors. Many aspects could be the cause of the bankruptcy, like

management ability, financial flexibility, the credibility of the company, etc [4]. This research aims to predict bankruptcy with three different methods: logistic discriminate analysis, probit discriminate analysis, and random forest, and compare these methods in terms of misclassification rate at this case.

2 Method

2.1 Dataset Exploration

The data set is from UCI Machine Learning Repository [5]. The samples were collected from one of the largest commercial banks in Korea during 2001–2002 [6]. There are 250 cases in total, 107 (42.8%) are in the bankrupt group while 143 (57.2%) are in the non-bankrupt group. The company in the bankrupt group are either in receivership or liquidation. The data set consists of 6 predictors, and all of them are risk factors rated by the experts who have about 9 years of rating experience on average [6]. These factors are industry risk, management risk, financial flexibility, credibility, competitiveness, and operating risk. Each of them has three levels: negative, average, positive, the percentage of each level is in table [3]. This data analysis refers the average level as the baseline level and code negative and positive level as indicator variables.

2.2 Logistic Discriminant Analysis

The response variable in this data analysis has two categories, bankruptcy (coded as 1) and no bankruptcy (coded as 0). There are many common ways to deal with this kind of response, logistic discriminate analysis is one of them. To do the analysis, the first step is to build a prediction model. Predictors (including interaction terms) are selected with AIC criteria, and Pearson and deviance residual plots are utilized to conduct model diagnostics. Then, check the leverage plot and cook's distance plot to identify influential outliers. Delete the influential outliers if any. The last is to evaluate the prediction performance of the model. Leave-one-out cross-validation (CV) and 10-fold CV are utilized to calculate the misclassification rate of the model. And detect whether the misclassification rate of bankruptcy or non-bankruptcy is much higher than the other.

2.3 Probit Discriminate Analysis and Random Forest

Some other classification methods are also considered to predict bankruptcy, the prediction abilities are compared in this case. To make the comparison reasonable, only the predictors selected from the logistic model are utilized in the models.

Since the quality of the classification depends on the choice of the link function, the probit link function is utilized to predict bankruptcy. On the other hand, all the predictors are categorical variables, random forest seems to be a good choice as well. Like the logistic discriminate analysis, the prediction ability of the probit regression model and random forest are tested by Leave-one-out CV and 10-fold CV.

3 Results

3.1 Logistic Regression Model Building

With AIC criteria, the selected predictors are negative management risk(NMA), negative financial flexibility(NFF), positive credibility(PCR), negative credibility(NCR), and negative operation risk(NOR). The interaction terms, in this case, do not show any significant effect. The model diagnostic result shows that the box-plot of Pearson residual and deviance residual shows a similar distribution, and residual v.s. the fitted plot does not show any systematic pattern (Appendix: Figure[1]). Thus, the model selected by AIC criteria fits the data well. The leverage plot and cook's distance plot doesn't show any influential outliers (Appendix: Fogure[2]). and the model format is:

$$logit(E(Y|X = x)) = \beta_0 + \beta_1 * NMA + \beta_2 * NFF + \beta_3 * PCR + \beta_4 * NCR + \beta_5 * NOR$$

3.2 Prediction ability of logistic model

The prediction ability is tested by leave-one-out Cross-validation and 10-fold CV. The result is in table [1]. It shows that the misclassification rate is similar by 10-fold CV and leave-one-out CV, which are around 2.5%. Moreover, about 60% of the misclassification is from the non-bankruptcy group, while around 40% of the misclassification is from the bankruptcy group.

	Logistic	Probit	Random forest
10-fold CV	2.62%	3.13%	3.2%
leave-one-out CV	2.51%	2.99%	2.8%

Table 1: Misclassification rate of logistic, probit and random forest model

3.3 Prediction Ability Comparison

When utilizing the probit model and random forest, the result from table [1] shows that the misclassification rates are similar among the three methods, and the logistic model tends to perform

slightly better. Also, among the three methods, the leave-one-out CV tends to have a smaller prediction error, which is because more data is used to fit the model.

3.4 Model Explanation

The Probit model with the specified predictors has the following format:

$$P(Y = 1|X = x) = \Phi(\beta_0 + \beta_1 * NMA + \beta_2 * NFF + \beta_3 * PCR + \beta_4 * NCR + \beta_5 * NOR)$$

Table [2] shows the fitted result of the logistic model and probit model with the whole dataset. All predictors are significant and the 95% CI does not contain 0. Both results show that NFF is the most influential feature and NCR is the second influential feature which a higher probability of bankruptcy. NMA and NCR behave similarly in terms of increment of bankruptcy, while the PCR is the predictor which significantly lowers the probability of bankruptcy. The intercept is the case when all the predictors are categorized to the "average" level. The estimated value in both models is all negative, which is an indication that the company does not tend to bankrupt with all average-level features.

The result in the random forest is similar by comparing with the regression models. It shows NFF is the most important feature in a random forest since the mean decrease accuracy is 50.67%. The second-largest mean decrease accuracy is PCR (23.61%), which agrees with the regression models that the absolute estimated parameter of PCR is the second largest.

	Logistic regression model			Probit regression model		
	Estimated	95% CI	P-value	Estimated	95% CI	P-value
(Intercept)	-7.02	[-10.11, -3.93]	8.198e-06	-2.92	[-4.08, -1.77]	6.320e-07
NMA	2.35	[0.65, 4.05]	6.684e-03	0.86	[0.127, 1.59]	2.140e-02
NFF	6.35	[4.15, 8.55]	1.549e-08	3.01	[2.13, 3.88]	1.288e-11
PCR	-2.69	[-4.68, -0.69]	8.080e-03	-1.61	[-2.61, -0.62]	1.389e-03
NCR	3.39	[1.53, 5.26]	3.628e-04	1.42	[0.62, 2.22]	4.7213e-04
NOR	2.16	[0.31, 4.03]	2.240e-02	0.83	[0.02, 1.65]	4.429e-02

Table 2: Model fit result: Logistic regression model and probit regression model

4 Discussion

4.1 Bankruptcy Avoidance

Although all predictors are correlated and all aspects need to be considered to successfully run a business, some features are more important than others to avoid bankruptcy. Among all the features in the dataset, negative financial flexibility (NFF) is the situation that a company needs to try most to avoid since it is the most influential feature leads to bankruptcy. And positive credibility (PCR) is the most important level that a company needs to achieve. Negative management risk (NMA), Negative credibility (NCR), and Negative operating risk (NOR) are also the risky categories that a company needs to avoid.

4.2 Model Application

As the report mentioned above, bankruptcy is a legal process, and different places have different policies [7]. This dataset is collected from Korean banks, and the predictor levels are assigned by the Korean financial experts. Thus the model can only be applied to the companies operating in Korea. As for other countries, if the bankruptcy policies and risk assessment criteria are different, then the model might not be appropriate.

4.3 Data Set Collection

Although the data set is collected from the loan management database of the bank [6], the collection mechanism is not specified. Since in reality, the bankruptcy case is relatively rare. If the data set is collected by filtering some non-bankruptcy cases, some other aspects need to be considered for this situation. On the other hand, the high bankruptcy rate might also because of fraud bankruptcy, in which case the debtor conceals assets and counterfeit bankruptcy documents to avoid pay their debts [8].

5 Conclusion

In this data analysis report, three classification methods are utilized to predict bankruptcy. The three methods behave similarly in terms of the misclassification rate. Among all the risky features, negative financial flexibility, Negative management risk, Negative credibility, and Negative operating risk are the most important situations that need to be avoided, and the positive credibility level should be achieved in the best effort.

References

[1] "Bankruptcy." Wikipedia, Wikimedia Foundation, 8 March 2021, https:// en.wikipedia.org/w/ index.php?title=Bankruptcyaction=history.

[2] Bankruptcy, United States Courts. https://www.uscourts.gov/services-forms/bankruptcy

[3] United States Courts. "Chapter 7 - Bankruptcy Basics." Accessed Sept. 15, 2020.

[4] Nurfauzi, R., Firmansyah, A. (2018). Managerial Ability , Management Compensation, Bankruptcy risk, Tax Aggressiveness. Media Riset Akuntansi, Auditing Informasi, 18(1), 75-100. doi:http://dx.doi.org/10.25105/mraai.v18i1.2775

[5] Qualitative Bankruptcy Data Set, UCI Machine Learning Repository, https://archive.ics.uci.edu/ml/datasets/QualitativeBankruptcy

[6] Myoung-Jong Kim*, Ingoo Han. (2003) The discovery of experts' decision rules from qualitative bankruptcy data using genetic algorithms. Expert Systems with Applications 25. 637–646.

[7] Stijn Claessens, Leora F. Klapper, Bankruptcy around the World: Explanations of Its Relative Use, American Law and Economics Review, Volume 7, Issue 1, Spring 2005, Pages 253–283.

[8] Ralph C. II McCullough, Bankruptcy Fraud: Crime without Punishment II, 102 COM. L.J.

Appendix: Table and Graph

	Average	Negative	Positive
industry risk	32.4%	35.6%	32.0%
management risk	27.6%	47.6%	24.8%
financial flexibility	29.6%	47.6%	22.8%
credibility	30.8%	37.6%	31.6%
competitiveness	22.4%	41.2%	36.4%
operating risk	22.8%	45.6%	31.6%

Table 3: Predictors description: percentage at each level (Average, Negative, and Positive)

Appendix: R code

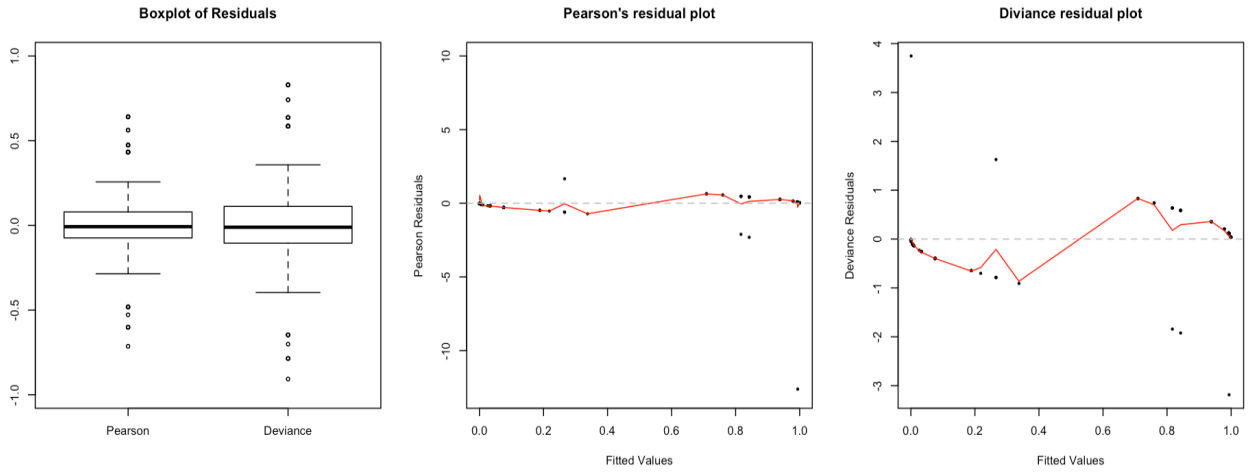

Figure 1: Model diagnostic: Residual plot of logistic model

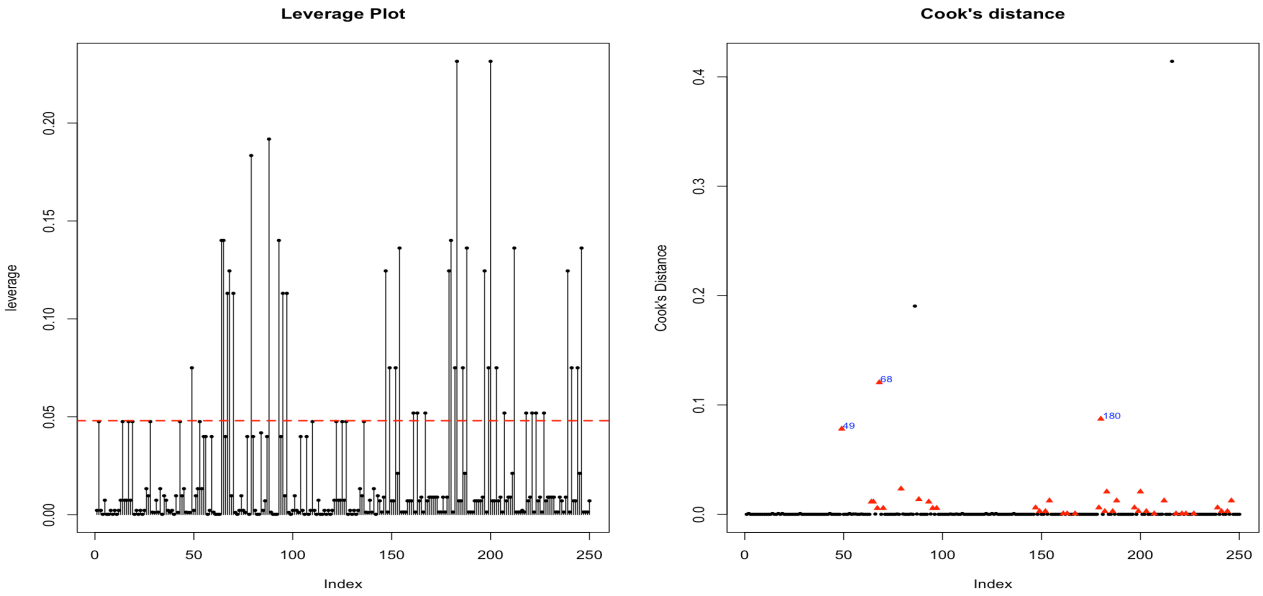

Figure 2: Leverage plot and cook's distance plot of logistic regression model

```
library(MASS)
library(dplyr)
install.packages("randomForest")
library(randomForest)
```

```
library(lawstat)
library(boot)
bank=read.table("/Users/rduan/Desktop/Qualitative_Bankruptcy/Qualitative_Bankruptcy
names(bank)=c("IR","MR","FF","CR","CO","OP","Class")
str(bank)
bank$Class=as.numeric(bank$Class)
bank$Class[1:143]=0
bank$Class[144:250]=1
bank$IR=as.numeric(bank$IR)
bank$IR[which(bank$IR==1)]=0
bank$IR[which(bank$IR==3)]=1
bank$IR[which(bank$IR==2)]=-1
bank$MR=as.numeric(bank$MR)
bank$MR[which(bank$MR==1)]=0
bank$MR[which(bank$MR==3)]=1
bank$MR[which(bank$MR==2)]=-1
bank$FF=as.numeric(bank$FF)
bank$FF[which(bank$FF==1)]=0
bank$FF[which(bank$FF==3)]=1
bank$FF[which(bank$FF==2)]=-1
bank$CR=as.numeric(bank$CR)
bank$CR[which(bank$CR==1)]=0
bank$CR[which(bank$CR==3)]=1
bank$CR[which(bank$CR==2)]=-1
bank$CO=as.numeric(bank$CO)
bank$CO[which(bank$CO==1)]=0
bank$CO[which(bank$CO==3)]=1
bank$CO[which(bank$CO==2)]=-1
bank$OP=as.numeric(bank$OP)
bank$OP[which(bank$OP==1)]=0
bank$OP[which(bank$OP==3)]=1
bank$OP[which(bank$OP==2)]=-1
bank1=bank
# Convert variables
bank1['IR_P']=0
bank1$IR_P[which(bank1$IR==1)]=1
bank1['IR_N']=0
```

```
bank1$IR_N[which(bank1$IR==-1)]=1
bank1['MR_P']=0
bank1['MR_N']=0
bank1$MR_P[which(bank1$MR==1)]=1
bank1$MR_N[which(bank1$MR==-1)]=1
bank1['FF_P']=0
bank1['FF_N']=0
bank1$FF_P[which(bank1$FF==1)]=1
bank1$FF_N[which(bank1$FF==-1)]=1
bank1['CR_P']=0
bank1['CR_N']=0
bank1$CR_P[which(bank1$CR==1)]=1
bank1$CR_N[which(bank1$CR==-1)]=1
bank1['CO_P']=0
bank1['CO_N']=0
bank1$CO_P[which(bank1$CO==1)]=1
bank1$CO_N[which(bank1$CO==-1)]=1
bank1['OP_P']=0
bank1['OP_N']=0
bank1$OP_P[which(bank1$OP==1)]=1
bank1$OP_N[which(bank1$OP==-1)]=1
bank2=bank1[,-c(1:6)]



# fit_glm
fit0=glm(Class~ (MR_N + MR_P+ FF_P+ FF_N + CR_P + CR_N + OP_P + OP_N)*(MR_N + MR_P+
              FF_P+ FF_N + CR_P + CR_N + OP_P + OP_N),family=binomial(),data=bank2,ma
stepAIC(fit0)
fit0=glm(Class~ (MR_N + FF_N + CR_P + CR_N + OP_N)*(MR_N + FF_N + CR_P + CR_N + OP_
stepAIC(fit0,direction="both")
fit1=glm(Class~ MR_N + FF_N + CR_P + CR_N + OP_N,family=binomial(),data=bank2,maxit
summary(fit1)
#  model diagnostic
par(mfrow=c(1,3))
res.P = residuals(fit1, type="pearson")
res.D = residuals(fit1, type="deviance") #or residuals(fit), by default
```

```
boxplot(cbind(res.P, res.D), names = c("Pearson", "Deviance"),ylim=c(-1,1),main="Bo
plot(fit1$fitted.values, res.P, pch=16, cex=0.6, ylab='Pearson Residuals', xlab='Fi
     ylim=c(-13,10),main="Pearson's residual plot")
lines(smooth.spline(fit1$fitted.values, res.P, spar=0.6), col=2)
abline(h=0, lty=2, col='grey')
plot(fit1$fitted.values, res.D, pch=16, cex=0.6, ylab='Deviance Residuals', xlab='F
lines(smooth.spline(fit1$fitted.values, res.D, spar=0.9), col=2)
abline(h=0, lty=2, col='grey')

# ** Leverage Points ----------------

leverage = hatvalues(fit1)
W = diag(fit1$weights)
X = cbind(rep(1,nrow(bank2)), bank2$MR_N, bank2$FF_N,bank2$CR_P,bank2$CR_N,bank2$OP
Hat = sqrt(W) %*% X %*% solve(t(X) %*% W %*% X) %*% t(X) %*% sqrt(W)
all(abs(leverage - diag(Hat)) < 1e-15)

plot(names(leverage), leverage, xlab="Index", type="h",main="Leverage Plot")
points(names(leverage), leverage, pch=16, cex=0.6)
p <- length(coef(fit1))
n <- nrow(bank2)
abline(h=2*p/n,col=2,lwd=2,lty=2)
infPts <- which(leverage>2*p/n)

# ** Cook's Distance ----------------
cooks = cooks.distance(fit1)
plot(cooks, ylab="Cook's Distance", pch=16, cex=0.6,main="Cook's distance")
points(infPts, cooks[infPts], pch=17, cex=0.8, col=2)
susPts <- as.numeric(names(sort(cooks[infPts], decreasing=TRUE)[1:3]))
text(susPts, cooks[susPts], susPts, adj=c(-0.1,-0.1), cex=0.7, col=4)

# Split data

# Logistic model
fit1=glm(Class~ MR_N + FF_N + CR_P + CR_N + OP_N,family=binomial(),data=bank2,maxit
s=summary(fit1)$coefficients
level=0.95
```

```
z=qnorm((1-level)/2,lower.tail = FALSE)
s[,1]
CI=cbind(cbind(s[,1],t(matrix(c(s[,1]-z*s[,2],s[,1]+z*s[,2]),2,6,byrow = TRUE))),s[
CI
cv.glm(bank2,fit1,K=10)$delta #0.02622236
cv.glm(bank2,fit1)$delta #0.02518603

index=list()
total=c(1:250)
for(i in 1:9){
  index[[i]]=sample(total,25,replace=FALSE)
  total=total[-which(total %in% index[[i]])]
  index[[10]]=total
}
err=0
error=0
for(i in 1:10){
  train=bank2[-index[[i]],]
  train$Class = as.factor(train$Class)
  test=bank2[index[[i]],]
  test$Class = as.factor(test$Class)
  fit=glm(Class~ MR_N + FF_N + CR_P + CR_N + OP_N,family=binomial(),data=train,maxi
  pred=predict(fit,test,type="response")
  pred=round(pred)
  table=table(test$Class,pred)
  print(table)
  error=error+table[1,2]/25
  err=err+(table[1,2]+table[2,1])/25
}
err/10 #0.02
error/10 #0.012 predict1/ truth=0



# probit
fit3=glm(Class~MR_N + FF_N + CR_P + CR_N + OP_N,family=binomial(link="probit"),data
summary(fit3)
```

```
s=summary(fit3)$coefficients
level=0.95
z=qnorm((1-level)/2,lower.tail = FALSE)
CI=cbind(cbind(s[,1],t(matrix(c(s[,1]-z*s[,2],s[,1]+z*s[,2]),2,6,byrow = TRUE))),s[
CI
cv.glm(bank2,fit3,K=10)$delta # 0.03133156
cv.glm(bank2,fit3)$delta # 0.02993939

# random forest
# split data
index=list()
total=c(1:250)
for(i in 1:9){
  index[[i]]=sample(total,25,replace=FALSE)
  total=total[-which(total %in% index[[i]])]
  index[[10]]=total
}
err=0
for(i in 1:10){
  train=bank2[-index[[i]],]
  train$Class = as.factor(train$Class)
  test=bank2[index[[i]],]
  test$Class = as.factor(test$Class)
  rf <- randomForest(Class~ MR_N+FF_N + CR_P + CR_N + OP_N, data=train, importance=
  prediction_for_table <- predict(rf,test)
  table=table(observed=test$Class,predicted=prediction_for_table)
  err=err+(table[1,2]+table[2,1])/25
}
err/10 # 0.032
# leave-one-out
t=0
for(i in 1:250){
  train=bank2[-i,]
  train$Class = as.factor(train$Class)
  test=bank2[i,]
  test$Class = as.factor(test$Class)
  rf <- randomForest(Class~ MR_N+FF_N + CR_P + CR_N + OP_N, data=train, importance=
```

```
  prediction_for_table <- predict(rf,test)
  if(test$Class==prediction_for_table){t=t+1}
  t=t
}
(250-t)/250 # 0.028

#bank3=bank2
#bank3$Class = as.factor(bank3$Class)

fit4=randomForest(Class~MR_N + FF_N + CR_P + CR_N + OP_N, data=bank3, importance=TR
fit4
varImpPlot(fit4)
round(importance(rf), 2)
# ROC curve
library(pROC)
index=sample(1:250,50,replace=FALSE)
train=bank2[-index,]
test=bank2[index,]
train$Class=as.factor(train$Class)
test$Class=as.factor(test$Class)
fit11=glm(Class~ MR_N + FF_N + CR_P + CR_N + OP_N,family=binomial(),data=train,maxi
pred_l=predict(fit11,test,type="response")

fit44=randomForest(Class~MR_N + FF_N + CR_P + CR_N + OP_N, data=train, importance=T
pred_r=predict(fit44,test,type="prob")
ROC_lr= roc(test$Class, pred_l)
ROC_rf=roc(test$Class, pred_r[,2])
# Area Under Curve (AUC) for each ROC curve (higher -> better)
ROC_rf_auc <- auc(ROC_rf)
ROC_lr_auc <- auc(ROC_lr)
plot(ROC_rf, col = "green", main = "ROC For Random Forest (GREEN) vs Logistic Regre
lines(ROC_lr, col = "red")
paste("Accuracy % of random forest:", mean(test$Class == round(pred_r[,2], digits
paste("Accuracy % of logistic regression:", mean(test$Class == round(pred_l, digit
paste("Area under curve of random forest:", ROC_rf_auc)
paste("Area under curve of logistic regression:", ROC_lr_auc)
```

```
par(mfrow=c(1,1))
par(pty = "s")
roc(bank2$Class, fit1$fitted.values, plot=TRUE, legacy.axes=TRUE, percent=TRUE, xla
plot.r
plot.roc(bank2$Class, as.rumeric(fit4$predicted), percent=TRUE, col="#4daf4a", lwd=
legend("bottomright", legend=c("Logisitic Regression", "Random Forest"), col=c("#37
br=read.table("/Users/rduan/Desktop/Qualitative_Bankruptcy/Qualitative_Bankruptcy.d
head(fit4$predicted)
class(head(as.rumeric(fit4$predicted)))
```