**Homework: This homework will test and develop your knowledge on predictive model construction and using their predictive performance for decision making. Please submit your answers for each section in a word document with your explanations with a supporting R script file you used for this assignment.**



|  |  |  |
| --- | --- | --- |
| **Attribute**  **No** | **Short**  **Name** | **Long**  **Name** |
| **1** | fyear | Fiscal year |
| **2** | at | Assets-total |
| **3** | bkvlps | Book value per share |
| **4** | invt | Inventories- total |
| **5** | Lt | Liabilities-total |
| **6** | rectr | Receivables-total |
| **7** | cogs | Cost of goods sold |
| **8** | dvt | Dividends-Total |
| **9** | ebit | Earnings before interests and taxes |
| **10** | gp | Gross profit |
| **11** | ni | Net income (loss) |
| **12** | oiadp | Operating income after depreciation |
| **13** | revt | Revenue- total |
| **14** | dvpsx\_f | Dividends per share -ex-date fiscal |
| **15** | mkvalt | Market value -Total -Fiscal |
| **16** | prch\_f | Price High -Annual-Fiscal |
| **17** | bankruptcy | Output (Y/N) |

1. Are there any variables with missing values? If so, remove records with any missing value.

Write your script for checking and removing records with missing values.

bankruptcy.data = as.data.frame(read.csv("C:/Users/Cassie Gill/Desktop/BankruptcyData.csv", header = TRUE, stringsAsFactors=T,

sep = ","))

head(bankruptcy.data)

summary(bankruptcy.data)

bankruptcy.data <- subset(bankruptcy.data, select = -fyear)

bankruptcy.data <- subset(bankruptcy.data, select = -oiadp)

bankruptcy.data <- subset(bankruptcy.data, select = -revt)

summary(bankruptcy.data)

No Missing values

1. Build a logistic regression model using the 70% of the data for training and 30% for the testing to predict bankruptcy in R. Present a summary table of the odds ratios for the significant predictors (p<0.05).

set.seed(1)

train.index = sample(c(1:dim(bankruptcy.data)[1]), dim(bankruptcy.data)[1]\*0.7)

train.data = bankruptcy.data[train.index, ]

test.data = bankruptcy.data[-train.index, ]

logit.reg = glm(formula = Bankrupt ~., data = train.data, family = "binomial")

options(scipen=999)

summary(logit.reg)

logit.reg.pred = predict(logit.reg, test.data, type = "response")

data.frame(actual = test.data$Bankrupt, predicted = logit.reg.pred)

exp(coef(logit.reg))

exp(cbind(OR = coef(logit.reg), confint(logit.reg)))

1. Use a variable selection algorithm to remove multicollinearity. Write the final model with the list of final predictors in the model.

library(car)

vif(logit.reg)

logit.reg = glm(formula = Bankrupt ~ bkvlps + invt + Lt + rectr +

cogs + dvt + gp + dvpsx\_f + mkvalt , data = train.data, family = "binomial")

1. Report the confusion matrix using the final logistic regression model and the test data. Calculate model accuracy, precision, sensitivity and F1 score.

confusionmatrix=table(test.data$Bankrupt, logit.reg.pred>0.5)

confusionmatrix

write.csv(confusionmatrix,"confusion\_matrix.csv")

sum(diag(confusionmatrix))/sum(sum(confusionmatrix))

confusionmatrix[2, 2]/(confusionmatrix[2, 2] + confusionmatrix[1, 2])

confusionmatrix[2, 2]/(confusionmatrix[2, 2] + confusionmatrix[2, 1])

(2\*confusionmatrix[2, 2])/((2\*confusionmatrix[2, 2]) + confusionmatrix[2, 1]+confusionmatrix[1, 2])

FALSE TRUE

no 163 (TN) 26 (FP)

yes 68 (FN) 140 (TP)

Accuracy =(TP+TN)/Total = (140+163)/397=0.7632

Precision =TP/(TP+FP)= 140/(140 + 26)=0.8434

Sensitivity =TP/(TP+FN)= 140/(140+68)=0.6731

F1 =2TP/2TP+FP+FN= (2\*140)/(2\*140)+26+68=.7487

1. Chart

   Description automatically generatedPlot the ROC curve and report the AUC statistic.

library(pROC)

log.reg.pred <- predict(logit.reg, test.data, decision.values=TRUE, type="response")

par(pty="s")

test.data\_prob = as.numeric(predict(logit.reg, test.data, decision.values=TRUE,type = "response"))-1

test.data\_roc = roc(test.data$Bankrupt ~ test.data\_prob, plot = TRUE, print.auc = TRUE, legacy.axes=TRUE, percent=TRUE, xlab="False Positive Rate", ylab="True Positive Rate", lwd=4)

auc(test.data\_roc)

par(pty="m")

Area under the curve: 85.95%

1. Diagram

   Description automatically generatedConstruct a decision tree model (called DT-1) to predict the bankruptcy with 70% of the data for training and 30% for the testing.

library(rpart)

library(party)

library(partykit)

library(caret)

library(caTools)

set.seed(1)

train2 = sample(c(1:dim(bankruptcy.data)[1]), dim(bankruptcy.data)[1]\*0.7)

DTtrain = bankruptcy.data[train.index, ]

DTtest = bankruptcy.data[-train.index, ]

DT.1 <- rpart(Bankrupt ~., method="class", data=DTtest)

plot(DT.1)

text(DT.1)

plot(as.party(DT.1))

printcp(DT.1)

plotcp(DT.1)

1. Calculate model accuracy, precision, sensitivity and F1 score. Plot the ROC curve with the AUC statistic reported.

DT.1pred <- predict(DT.1, DTtest, decision.values=TRUE, type="class")

DT.1CM=table(DT.1pred = DT.1pred, true = DTtest$Bankrupt)

DT.1CM

sum(diag(DT.1CM))/sum(sum(DT.1CM))

DT.1CM[2, 2]/(DT.1CM[2, 2] + DT.1CM[1, 2])

DT.1CM[2, 2]/(DT.1CM[2, 2] + DT.1CM[2, 1])

(2\*DT.1CM[2, 2])/((2\*DT.1CM[2, 2]) + DT.1CM[2, 1]+DT.1CM[1, 2])

library(pROC)

DT.1pred <- predict(DT.1, DTtest, decision.values=TRUE, type="class")

par(pty="s")

DTtest\_prob = as.numeric(predict(DT.1, DTtest, decision.values=TRUE,type = "class"))-1

DTtest\_roc = roc(DTtest$Bankrupt ~ DTtest\_prob, plot = TRUE, print.auc = TRUE, legacy.axes=TRUE, percent=TRUE, xlab="False Positive Rate", ylab="True Positive Rate", lwd=4)

auc(DTtest\_roc)

par(pty="m")

FALSE TRUE

no 176 (TN) 23 (FP)

yes 13 (FN) 185 (TP)

Accuracy =(TP+TN)/Total = (185+176)/397=0.9093

Precision =TP/(TP+FP)= 185/(185 + 23)=0.8894

Sensitivity =TP/(TP+FN)= 185/(185+13)=0.9343

F1 =2TP/2TP+FP+FN= (2\*185)/(2\*185)+23+13=.9113

AUC = .9103

1. Diagram, schematic

   Description automatically generatedFor pruning the decision tree what complexity parameter value would you use? Prune the tree and construct the confusion matrix for the pruned model (called DT-2). Calculate model accuracy, precision, sensitivity and F1 score for DT-2. Plot the ROC curve with the AUC statistic reported.

m <- which.min(DT.1$cptable[, "xerror"])

DT.1$cptable[m, "CP"]

DT.2 <- prune(DT.1, cp = DT.1$cptable[which.min(DT.1$cptable[, "xerror"]), "CP"])

plot(as.party(DT.2))

printcp(DT.2)

plotcp(DT.2)

DT.2pred <- predict(DT.2, DTtest, decision.values=TRUE, type="class")

DT.2CM=table(DT.2pred = DT.2pred, true = DTtest$Bankrupt)

DT.2CM

sum(diag(DT.2CM))/sum(sum(DT.2CM))

DT.2CM[2, 2]/(DT.2CM[2, 2] + DT.2CM[1, 2])

DT.2CM[2, 2]/(DT.2CM[2, 2] + DT.2CM[2, 1])

(2\*DT.2CM[2, 2])/((2\*DT.2CM[2, 2]) + DT.2CM[2, 1]+DT.2CM[1, 2])

library(pROC)

DT.2pred <- predict(DT.2, DTtest, decision.values=TRUE, type="class")

par(pty="s")

DTtest\_prob = as.numeric(predict(DT.2, DTtest, decision.values=TRUE,type = "class"))-1

DTtest\_roc = roc(DTtest$Bankrupt ~ DTtest\_prob, plot = TRUE, print.auc = TRUE, legacy.axes=TRUE, percent=TRUE, xlab="False Positive Rate", ylab="True Positive Rate", lwd=4)

auc(DTtest\_roc)

par(pty="m")

I would use a complexity parameter of .01851852.

FALSE TRUE

no 173 (TN) 34 (FP)

yes 16 (FN) 174 (TP)

Testing

Accuracy =(TP+TN)/Total = (174+173)/397=0.8741

Precision =TP/(TP+FP)= 174/(174 + 34)=0.8365

Sensitivity =TP/(TP+FN)= 174/(174+16)=0.9158

F1 =2TP/2TP+FP+FN= (2\*174)/(2\*174)+34+16=.8744

AUC = .8759

1. Which model would you use for predicting the new applications? Explain why.

I would use the unpruned decision tree model because it has the highest model accuracy, precision, sensitivity and F1 score. The decision tree method also has the advantage that it could be used manually if a system outage occurred.

1. The credit organization received new applications with the attribute values in the new Bankrupcynew.csv data file. Use one of the models you constructed to predict if the companies will undergo bankruptcy or not. Report the resulting table.

bankruptcynew.data = as.data.frame(read.csv("C:/Users/Cassie Gill/Desktop/BankruptcyNew.csv", header= TRUE, stringsAsFactors=T, sep = ","))

head(bankruptcynew.data)

summary(bankruptcynew.data)

bankruptcynew.data <- subset(bankruptcynew.data, select = -fyear)

bankruptcynew.data <- subset(bankruptcynew.data, select = -oiadp)

bankruptcynew.data <- subset(bankruptcynew.data, select = -revt)

summary(bankruptcynew.data)

Final.Pred = predict(DT.1, bankruptcynew.data, type = "class")

Final.Pred

1 2 3 4 5

no no yes yes yes

Levels: no yes

