College Classification Project

Kevin Torres

**Task:** I will explore the use of tree methods to classify schools as Private or Public based off their features.

**Dataset:** I will be using data which is included in the ISLR library in R, the College data frame.

Call the necessary library and re-assign the data

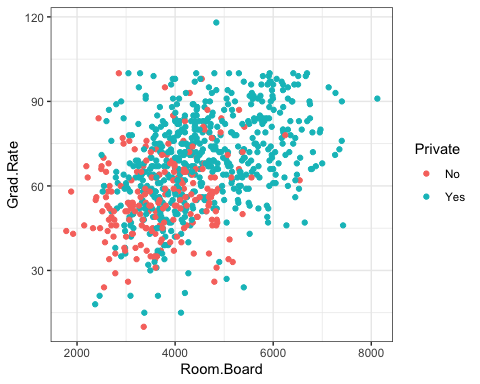
library(ISLR)  
  
head(College)

## Private Apps Accept Enroll Top10perc Top25perc  
## Abilene Christian University Yes 1660 1232 721 23 52  
## Adelphi University Yes 2186 1924 512 16 29  
## Adrian College Yes 1428 1097 336 22 50  
## Agnes Scott College Yes 417 349 137 60 89  
## Alaska Pacific University Yes 193 146 55 16 44  
## Albertson College Yes 587 479 158 38 62  
## F.Undergrad P.Undergrad Outstate Room.Board Books  
## Abilene Christian University 2885 537 7440 3300 450  
## Adelphi University 2683 1227 12280 6450 750  
## Adrian College 1036 99 11250 3750 400  
## Agnes Scott College 510 63 12960 5450 450  
## Alaska Pacific University 249 869 7560 4120 800  
## Albertson College 678 41 13500 3335 500  
## Personal PhD Terminal S.F.Ratio perc.alumni Expend  
## Abilene Christian University 2200 70 78 18.1 12 7041  
## Adelphi University 1500 29 30 12.2 16 10527  
## Adrian College 1165 53 66 12.9 30 8735  
## Agnes Scott College 875 92 97 7.7 37 19016  
## Alaska Pacific University 1500 76 72 11.9 2 10922  
## Albertson College 675 67 73 9.4 11 9727  
## Grad.Rate  
## Abilene Christian University 60  
## Adelphi University 56  
## Adrian College 54  
## Agnes Scott College 59  
## Alaska Pacific University 15  
## Albertson College 55

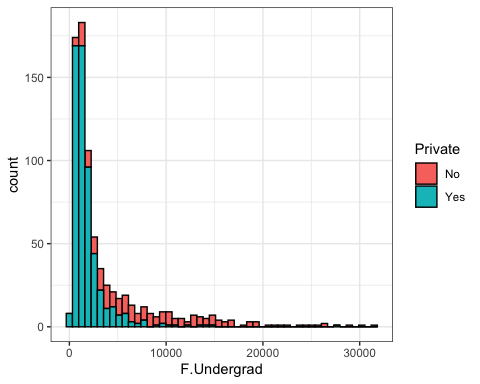
df <- data.frame(College)

Exploratory Data Analysis

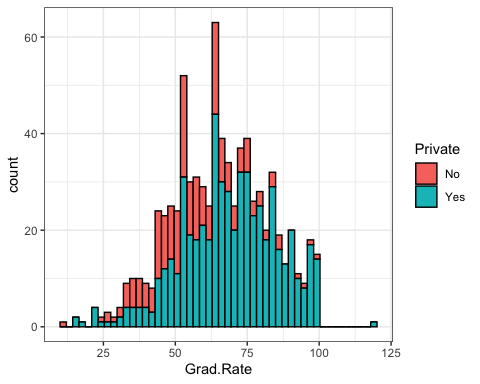
library(ggplot2)  
# Scatterplot of graduation rate versus room and board rate, colored by the Private column  
ggplot(df, aes(Room.Board, Grad.Rate, color = Private)) + geom\_point() + theme\_bw()



# Histogram of full time undergraduate students, colored by the Private column  
ggplot(df, aes(F.Undergrad, fill = Private)) + geom\_histogram(bins = 50, color = 'black') + theme\_bw()



# Histogram of graduation rate colored by Private  
ggplot(df, aes(Grad.Rate, fill = Private)) + geom\_histogram(bins = 50, color = 'black') + theme\_bw()



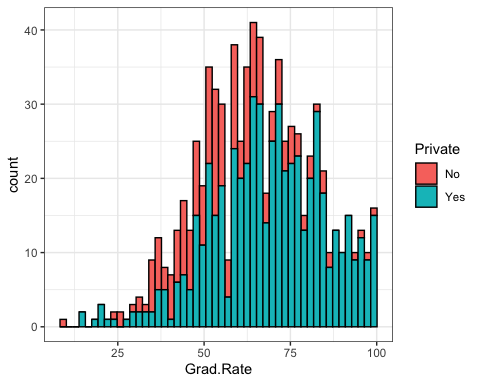
# We find there is a college that has a graduation rate over 100

Find and change the graduation rate of this university to 100

subset(df, Grad.Rate > 100)

## Private Apps Accept Enroll Top10perc Top25perc F.Undergrad  
## Cazenovia College Yes 3847 3433 527 9 35 1010  
## P.Undergrad Outstate Room.Board Books Personal PhD Terminal  
## Cazenovia College 12 9384 4840 600 500 22 47  
## S.F.Ratio perc.alumni Expend Grad.Rate  
## Cazenovia College 14.3 20 7697 118

df['Cazenovia College', 'Grad.Rate'] <- 100  
  
# Check the histogram to see if the value is no longer above 100  
ggplot(df, aes(Grad.Rate, fill = Private)) + geom\_histogram(bins = 50, color = 'black') + theme\_bw()



Create a train and test data set (call necessary libraries)

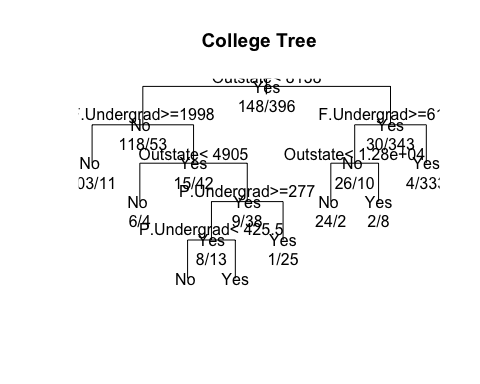
library(caTools)  
  
sample <- sample.split(df$Private, SplitRatio = .7)  
train <- subset(df, sample == T)  
test <- subset(df, sample == F)

Build the decision tree (call necessary libraries)

library(rpart)  
  
# Pass in method as class since we are doing classification  
tree <- rpart(Private ~ ., method = 'class', data = train)  
printcp(tree)

##   
## Classification tree:  
## rpart(formula = Private ~ ., data = train, method = "class")  
##   
## Variables actually used in tree construction:  
## [1] F.Undergrad Outstate P.Undergrad  
##   
## Root node error: 148/544 = 0.27206  
##   
## n= 544   
##   
## CP nsplit rel error xerror xstd  
## 1 0.439189 0 1.00000 1.00000 0.070132  
## 2 0.182432 1 0.56081 0.66892 0.060805  
## 3 0.108108 2 0.37838 0.48649 0.053404  
## 4 0.040541 3 0.27027 0.33784 0.045529  
## 5 0.013514 4 0.22973 0.32432 0.044699  
## 6 0.010000 7 0.18919 0.32432 0.044699

plot(tree, main = 'College Tree', uniform = T)  
text(tree, use.n = T, all = T)



Now predict using the test data

tree.pred <- predict(tree, test)  
head(tree.pred)

## No Yes  
## Adrian College 0.01186944 0.9881306  
## Albertson College 0.01186944 0.9881306  
## Alma College 0.01186944 0.9881306  
## Alverno College 0.01186944 0.9881306  
## American International College 0.01186944 0.9881306  
## Amherst College 0.01186944 0.9881306

tree.pred <- as.data.frame(tree.pred)  
  
# Apply this joiner function to the new Private column in this data frame to be used for comparison later  
joiner <- function(x){  
 if (x >= 0.5) {  
 return('Yes')  
 } else {  
 return('No')  
 }  
}  
  
tree.pred$Private <- sapply(tree.pred$Yes, joiner)  
print(head(tree.pred))

## No Yes Private  
## Adrian College 0.01186944 0.9881306 Yes  
## Albertson College 0.01186944 0.9881306 Yes  
## Alma College 0.01186944 0.9881306 Yes  
## Alverno College 0.01186944 0.9881306 Yes  
## American International College 0.01186944 0.9881306 Yes  
## Amherst College 0.01186944 0.9881306 Yes

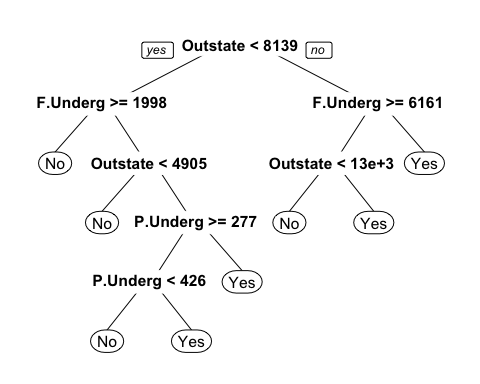
# Now we can compare this column we created to the Private column in the test data set to measure the success  
  
table(tree.pred$Private, test$Private)

##   
## No Yes  
## No 59 11  
## Yes 5 158

# Here we see that the model was fairly successful in it's predictions and we can compare it to our Random Forest Model

Now we can plot out the tree

library(rpart.plot)  
prp(tree)



Now let’s build a Random Forest Model

library(randomForest)

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##   
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':  
##   
## margin

# We can see how the model performs based on the train data set  
priv.rf.model <- randomForest(Private ~ ., train, importance = T)  
priv.rf.model$confusion

## No Yes class.error  
## No 126 22 0.14864865  
## Yes 18 378 0.04545455

# We find it is easier to predict whether a college is Private rather than if it is not, which may be due to the amount of private school data  
  
# We can get the importance based on the Gini Impurity Index values since we included importance  
priv.rf.model$importance

## No Yes MeanDecreaseAccuracy MeanDecreaseGini  
## Apps 3.066101e-02 0.0107999636 1.619015e-02 9.698686  
## Accept 2.787785e-02 0.0123021843 1.654143e-02 12.618363  
## Enroll 5.546281e-02 0.0323621986 3.862097e-02 24.721933  
## Top10perc 6.929319e-03 0.0033067615 4.275216e-03 4.502142  
## Top25perc 3.410539e-03 0.0039170290 3.737945e-03 3.809620  
## F.Undergrad 1.410533e-01 0.0493339239 7.420610e-02 33.512033  
## P.Undergrad 5.358710e-02 0.0047857433 1.788106e-02 14.630197  
## Outstate 1.516064e-01 0.0582112908 8.329487e-02 49.003999  
## Room.Board 1.075274e-02 0.0168016311 1.512545e-02 10.439864  
## Books 7.817635e-05 -0.0001301818 -5.857684e-05 1.872663  
## Personal 4.153603e-03 0.0010241904 1.872511e-03 4.135083  
## PhD 1.357230e-02 0.0059839627 8.004325e-03 5.202193  
## Terminal 1.101808e-02 0.0055377230 6.948083e-03 4.308670  
## S.F.Ratio 3.000339e-02 0.0066745679 1.300838e-02 14.126343  
## perc.alumni 2.945245e-02 0.0014540658 8.943165e-03 5.120744  
## Expend 2.273490e-02 0.0113371331 1.439755e-02 9.801495  
## Grad.Rate 2.321041e-02 0.0040041816 9.253532e-03 7.327891

Now we can predict

rf.preds <- predict(priv.rf.model,test)  
table(rf.preds, test$Private)

##   
## rf.preds No Yes  
## No 60 5  
## Yes 4 164

# We could say this model performed better than the single decision tree  
# This will be due to the risk that comes with the Type 1 and Type 2 errors  
 # Once that risk is evaluated, then we could more easily decide on a model