ISL Chaper 2 Exercises

Jonathan Bryan

May 7, 2017

1. For each of parts (a) through (d), indicate whether we would generally expect the performance of a flexible statistical learning method to be better or worse than an inflexible method. Justify your answer.

(a) The sample size n is extremely large, and the number of predictors p is small.

Flexible models perform well when the sample size of the training set is large and a large number of parameters can be estimated to build the flexible model. Therefore a flexible model may be preferred.

(b) The number of predictors p is extremely large, and the number of observations n is small.

Flexible models can be difficult or impossible to fit when there are not enough observations available. A flexible model may also over fit the small number of observation, performing poorly on out-of-sample testing. Therefore a flexible model may not be preferred.

(C) The relationship between the predictors and response is highly non-linear.

Many flexible models can more easily and accurately model non-linear behavior so a non-linear model may be preferred.

(d) The variance of the error terms, i.e. $\sigma^2 = Var(\epsilon)$, is extremely high.

Flexible models tend to "chase" or closely follow the variation found in the training set data. This leads to high variability each time the model is run on different sets of training data. Therefore, increasing the variability of the noise in the data will increase the variability in the flexible model which is sub optimal. Therefore, a more flexible model may not be preferred.

2. Explain whether each scenario is a classification or regression problem, and indicate whether we are most interested in inference or prediction. Finally, provide n and p.

(a) We collect a set of data on the top 500 firms in the US. For each firm we record profit, number of employees, industry and the CEO salary. We are interested in understanding which factors affect CEO salary.

This is a regression problem with the goal of inference (n = 500 and p = 4).

(b) We are considering launching a new product and wish to know whether it will be a success or a failure. We collect data on 20 similar products that were previously launched. For each product we have recorded whether it was a success or failure, price charged for the product, marketing budget, competition price, and ten other variables.

This is a classification problem with the goal of prediction (n = 20 and p = 13).

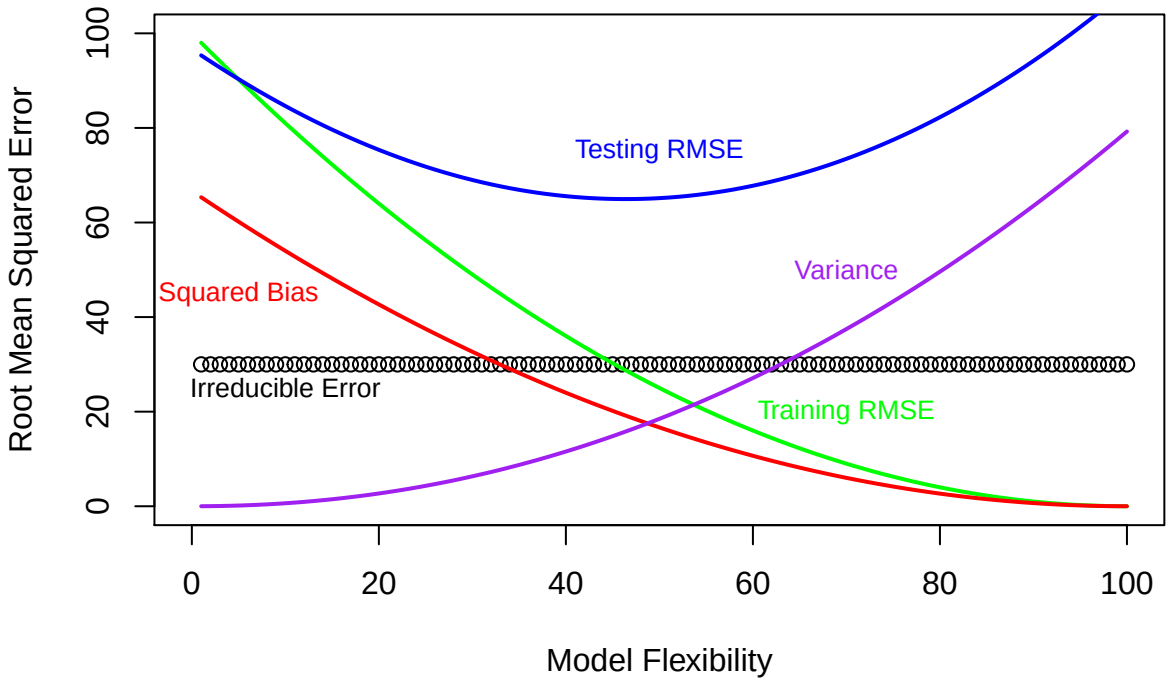
(c) We are interesting in predicting the % change in the US dollar in relation to the weekly changes in the world stock markets. Hence we collect weekly data for all of 2012. For each week we record the % change in the dollar, the % change in the US market, the % change in the British market, and the % change in the German market.

This is a regression problem with the goal of prediction (n = 366 and p = 3).

3. We now revisit the bias-variance decomposition.

(a) Provide a sketch of typical (squared) bias, variance, training error, test error, and Bayes (or irreducible) error curves, on a single plot, as we go from less flexible statistical learning methods towards more flexible approaches. The x-axis should represent the amount of flexibility in the method, and the y-axis should represent the values for each curve. There should be five curves. Make sure to label each one.

```
x = seq(1,100,1)
irre_y = rep(30,100)
train_y = (1/100)*(x-100)^2
bias_y = (1/150)*(x-100)^2
variance_y = (1/200)*(x)^(2.1)
test_y = bias_y + variance_y + irre_y


plot(x, irre_y, xlim = c(0,100), ylim=c(0,100), xlab = "Model Flexibility", ylab = "Root Mean Squared E
lines(x,train_y, col="green", lwd = 2) #Training RMSE
lines(x,bias_y, col = "red", lwd = 2) #Squared Bias
lines(x,variance_y, col = "purple", lwd = 2) #Variance
lines(x,test_y, col = "blue", lwd = 2) #Testing RMSE
text(x = 10 , y = 25, labels = "Irreducible Error", cex = 0.8)
text(x = 70 , y = 20, labels = "Training RMSE", cex = 0.8, col="green")
text(x = 5 , y = 45, labels = "Squared Bias", cex = 0.8, col = "red")
text(x = 70 , y = 50, labels = "Variance", cex = 0.8, col = "purple")
text(x = 50 , y = 75, labels = "Testing RMSE", cex = 0.8, col = "blue" )
```



(b)

Bias decreases as the model is better able to account for the variation in individual data points through increasingly flexibility and relaxing of persistent model constraints. Variance of the model increases as the model tracks the training data more closely and subsequent iterations of model generation produce very different results depending on the training data used. Training error consistently decreases as more flexible models are able to account for the variation in the training data and bias steadily decreases. Test error consistently decrease as the rate of bias reduction is higher than the rate of increase in model variance. Eventually, the rate of model variance overtakes the rate of model bias reduction and the overall test error begins to increase. The test error can not go below the level of irreducible error. Irreducible error is the level of measurement error inherent in the data and cannot be reduced in the model.

4. You will now think of some real-life applications for statistical learning.

(a) Describe three real-life applications in which classification might be useful. Describe the response, as well as the predictors. Is the goal of each application inference or prediction? Explain your answer.

1. A zoologist wants to predict the taxonomy of animal remains found in a keystone predator. The zoologist has over 1,000 possible classifications of species and uses the genetic material to build the model. This is a classification problem with the goal of prediction.The response variable can be 1,000 different discrete categories that have no "order" or "ranking".

2. An investment firm wants to know whether positive or negative earnings reports by media companies are influenced by public sentiment on Twitter. This is an inferential classification problem with positive

and negative earnings reports of media companies as the response and metrics that represent public sentiment on Twitter as the predictor variables.

3. A financial consulting company uses a stress test on banking clients to determine their firms ability to weather financial crises. The firm predicts has multiple different scenarios, each with a pass or fail response. The consulting company also uses the stress test to advise clients on best practices to avoid failure. This is a classification problem with both prediction and inference as a goal. The response is the multiple different stress test scenarios each with a pass or fail factor.

(b) Describe three real-life applications in which regression might be useful. Describe the response, as well as the predictors. Is the goal of each application inference or prediction? Explain your answer.

1. A large retail chain wants to predict the change in a specific product's sales based upon the store placement and the combination of other store products in close proximity. This is a regression question, where the response is the change in store sales and the predictors are store placement location and categorical variables representing different items placed next to the product.
2. An educational institution wants to predict graduate student's "grit" through physical and mental assessments. Grit is measured on a continuous numerical scale from 1 to 10, while the physical and mental assessments are a combination of categorical and numerical data. This is a regression problem with the goal of prediction. The response variable is grit while the assessment metrics are the predictors.
3. A healthcare system wants to understand the greatest sources of medical waste in dollars at the medical department level using data from physician, supply chain and medical records. This is a regression problem with the goal of inference. The response variable is the dollars of medical waste, on a continuous scale.

(c) Describe three real-life applications in which cluster analysis might be useful.

1. A large retail chain is testing a new product for launch and wants to segment the customer population currently buying the product in test markets. A Cluster analysis could be run to determine the distance "types" of customers who enjoy the product.
2. An oil exploration company would like to classify different types of oil extraction sites based upon return on investment, safety incidents, environmental impact, and other variables. There is no response variable so we can cluster the oil extraction sites based on similarity.
3. A toy company would like to market a new toy to a particular high-value customer segment. They would like to to know which attributes are clustered around different market segments and then blend those attributes together to create the new toy to appeal to the new customer segment. Cluster analysis can be run on the current toys to ascertain which attributes are closely associated and then match those clusters to market segments.

8.

(a) Use the read.csv() function to read the data into R. Call the loaded data college. Make sure that you have the directory set

to the correct location for the data.

```
library(ISLR)
college = College
```

(b) Look at the data using the fix() function. You should notice that the first column is just the name of each university.We don't really want R to treat this as data. However, it may be handy to have these names for later.

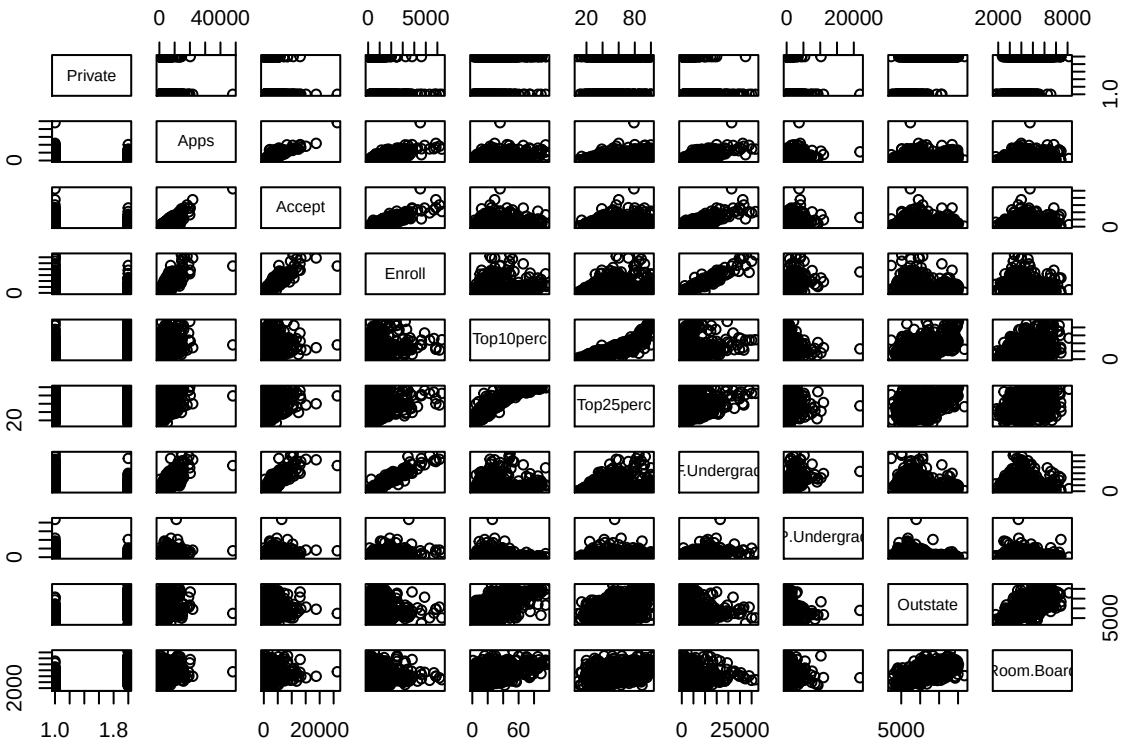
```
fix(college)
```

(c)

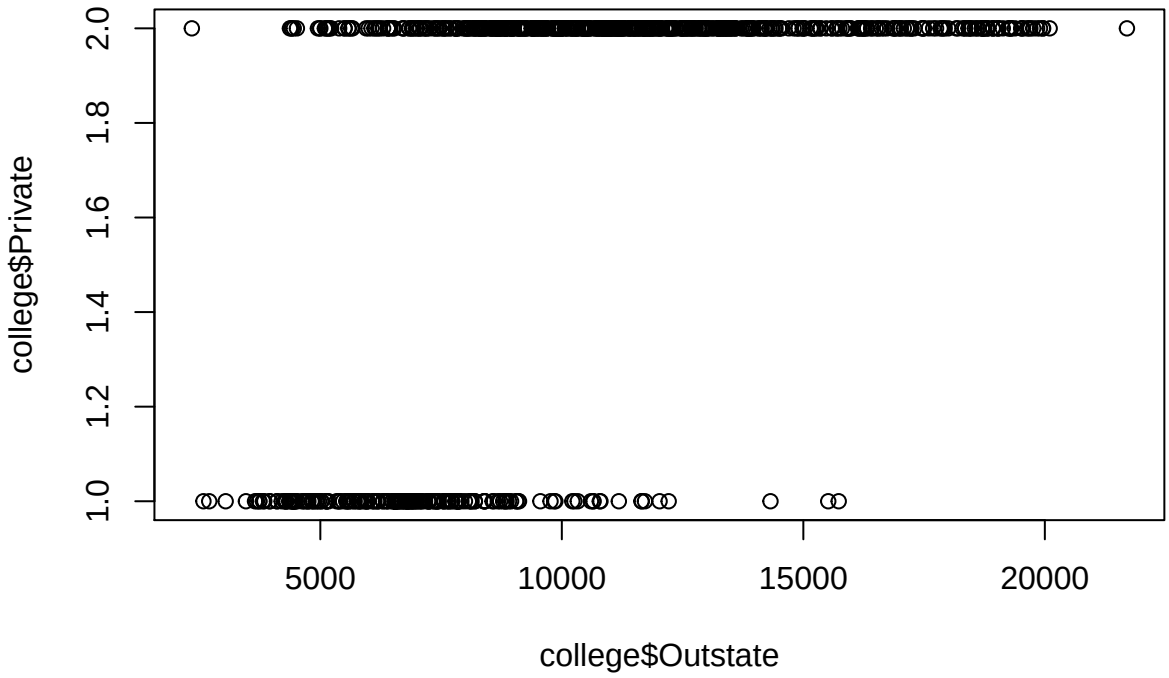
```
summary(college) #i
```

```
##  Private        Apps           Accept          Enroll       Top10perc
##  No :212   Min.   :   81   Min.   :   72   Min.   :  35   Min.   : 1.00
##  Yes:565   1st Qu.:  776   1st Qu.:  604   1st Qu.: 242   1st Qu.:15.00
##            Median : 1558   Median : 1110   Median : 434   Median :23.00
##            Mean   : 3002   Mean   : 2019   Mean   : 780   Mean   :27.56
##            3rd Qu.: 3624   3rd Qu.: 2424   3rd Qu.: 902   3rd Qu.:35.00
##            Max.   :48094   Max.   :26330   Max.   :6392   Max.   :96.00
##    Top25perc      F.Undergrad     P.Undergrad         Outstate
##  Min.   :  9.0   Min.   :  139   Min.   :    1.0   Min.   : 2340
##  1st Qu.: 41.0   1st Qu.:  992   1st Qu.:   95.0   1st Qu.: 7320
##  Median : 54.0   Median : 1707   Median :  353.0   Median : 9990
##  Mean   : 55.8   Mean   : 3700   Mean   :  855.3   Mean   :10441
##  3rd Qu.: 69.0   3rd Qu.: 4005   3rd Qu.:  967.0   3rd Qu.:12925
##  Max.   :100.0   Max.   :31643   Max.   :21836.0   Max.   :21700
##    Room.Board       Books           Personal         PhD
##  Min.   :1780   Min.   :  96.0   Min.   : 250   Min.   :  8.00
##  1st Qu.:3597   1st Qu.: 470.0   1st Qu.: 850   1st Qu.: 62.00
##  Median :4200   Median : 500.0   Median :1200   Median : 75.00
##  Mean   :4358   Mean   : 549.4   Mean   :1341   Mean   : 72.66
##  3rd Qu.:5050   3rd Qu.: 600.0   3rd Qu.:1700   3rd Qu.: 85.00
##  Max.   :8124   Max.   :2340.0   Max.   :6800   Max.   :103.00
##     Terminal       S.F.Ratio      perc.alumni        Expend
##  Min.   : 24.0   Min.   : 2.50   Min.   : 0.00   Min.   : 3186
##  1st Qu.: 71.0   1st Qu.:11.50   1st Qu.:13.00   1st Qu.: 6751
##  Median : 82.0   Median :13.60   Median :21.00   Median : 8377
##  Mean   : 79.7   Mean   :14.09   Mean   :22.74   Mean   : 9660
##  3rd Qu.: 92.0   3rd Qu.:16.50   3rd Qu.:31.00   3rd Qu.:10830
##  Max.   :100.0   Max.   :39.80   Max.   :64.00   Max.   :56233
##    Grad.Rate
##  Min.   : 10.00
##  1st Qu.: 53.00
##  Median : 65.00
##  Mean   : 65.46
##  3rd Qu.: 78.00
##  Max.   :118.00
```

```
pairs(college[,1:10]) #ii
```



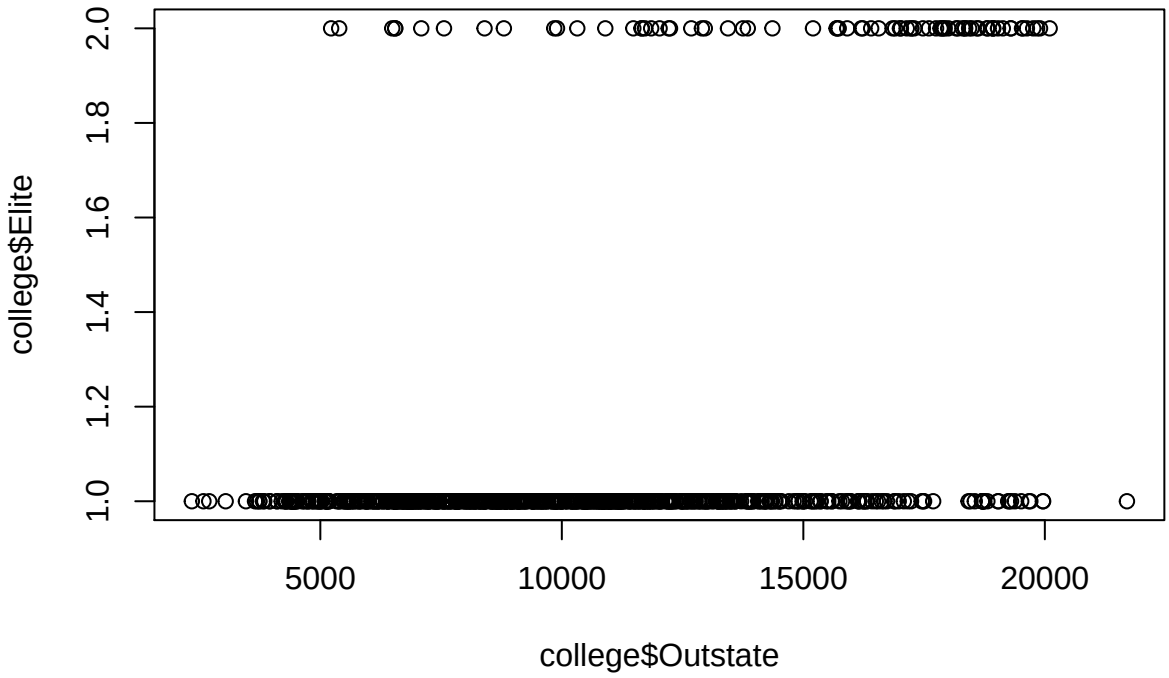

```r
plot(college$Outstate,college$Private ) #iii
```



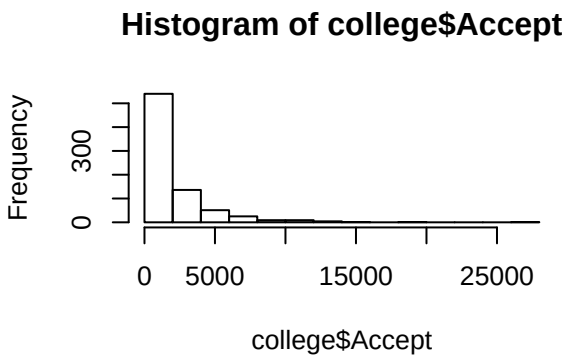

```
college$Elite = rep("No", nrow(college)) #iv
college$Elite[college$Top10perc > 50 ] = "Yes"
college$Elite = as.factor(college$Elite)
summary(college$Elite)#Number of elite colleges
```

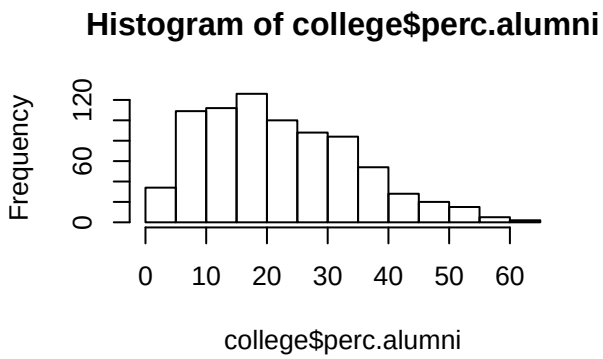
```
##  No Yes
## 699  78
```

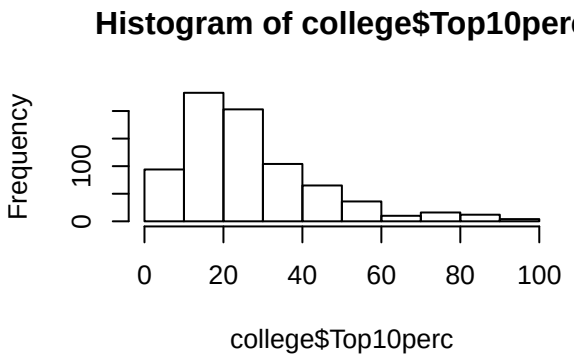
```
plot(college$Outstate, college$Elite)
```

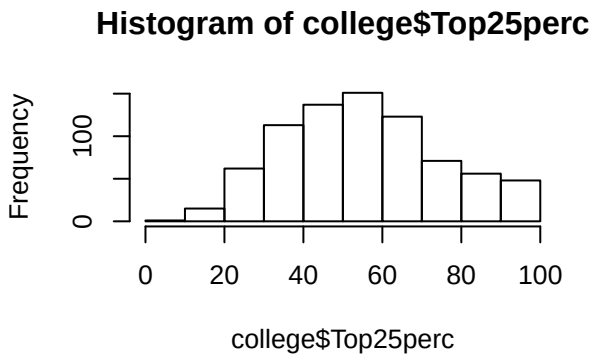



```
par(mfrow=c(2,2)) #υ
hist(college$Accept)
hist(college$perc.alumni)
hist(college$Top10perc)
hist(college$Top25perc)
```

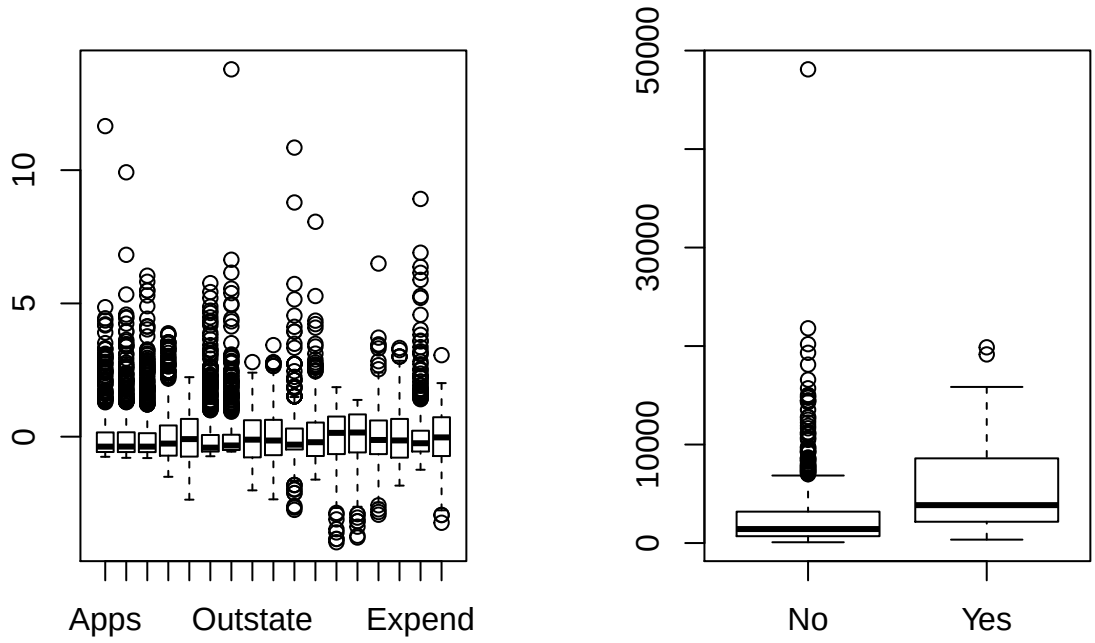









```
#vi Continue exploring data and brief summary
college_mcs = sapply(college, is.numeric)
college_mcs = college[college_mcs]
college_mcs = sapply(college_mcs, function(college_mcs) (college_mcs - mean(college_mcs))/sd(college_mcs

par(mfrow = c(1,2))
boxplot(college_mcs)
plot(college$Elite, college$Apps)
```



We observe that many of the numeric variables have fat tail empirical distributions that skew towards positive values. We also observe that elite institutions receive many more applications than non-elite institutions.

9. This exercise involves the Auto data set studied in the lab. Make sure that the missing values have been removed from the data.

(a) Which of the predictors are quantitative, and which are qualitative?

```
auto = Auto

categorical = sapply(auto, is.factor) #Determine which variables are categorical
print(paste0( "The categorical variable is: ",colnames(auto[categorical]))) #Get names of categorical v
```

```
## [1] "The categorical variable is: name"
```

```
numeric = sapply(auto, is.numeric) #Determine which variables are categorical
print(paste0("The numeric variables are: ", colnames(auto[numeric]))) #Get names of categorical variabl
```

```
## [1] "The numeric variables are: mpg"
## [2] "The numeric variables are: cylinders"
## [3] "The numeric variables are: displacement"
## [4] "The numeric variables are: horsepower"
## [5] "The numeric variables are: weight"
## [6] "The numeric variables are: acceleration"
## [7] "The numeric variables are: year"
## [8] "The numeric variables are: origin"
```

(b) What is the range of each quantitative predictor? You can answer this using the range() function.

```
sapply(auto[numeric], range) #Range of each numeric variable
```

```
##       mpg cylinders displacement horsepower weight acceleration year
## [1,]  9.0         3           68         46   1613          8.0   70
## [2,] 46.6         8          455        230   5140         24.8   82
##      origin
## [1,]      1
## [2,]      3
```

(c) What is the mean and standard deviation of each quantitative predictor?

```
print("Mean of each Numeric Variable")
```

```
## [1] "Mean of each Numeric Variable"
```

```
sapply(auto[numeric], mean) #Mean of each numeric variable
```

```
##          mpg    cylinders displacement   horsepower       weight
##    23.445918     5.471939   194.411990   104.469388  2977.584184
## acceleration         year       origin
##    15.541327    75.979592     1.576531
```

```
print("Standard Deviation of each Numeric Variable")
```

```
## [1] "Standard Deviation of each Numeric Variable"
```

```
sapply(auto[numeric], sd) #Standard deviation of each numeric variable
```

```
##          mpg    cylinders displacement   horsepower       weight
##    7.8050075    1.7057832  104.6440039   38.4911599  849.4025600
## acceleration         year       origin
##    2.7588641    3.6837365    0.8055182
```

(d) Now remove the 10th through 85th observations. What is the range, mean, and standard deviation of each predictor in the subset of the data that remains?

```
auto_new = auto[-c(10,85),]

print("Mean of each Numeric Variable")
```

```
## [1] "Mean of each Numeric Variable"
```

```
sapply(auto_new[numeric], mean) #Mean of each numeric variable
```

```
##          mpg    cylinders displacement   horsepower       weight
##    23.494359     5.458974   193.511538   104.069231  2972.469231
## acceleration         year       origin
##    15.565897    76.002564     1.579487
```

```
print("Standard Deviation of each Numeric Variable")
```

```
## [1] "Standard Deviation of each Numeric Variable"
```

```r
sapply(auto_new[numeric], sd) #Standard deviation of each numeric variable
```

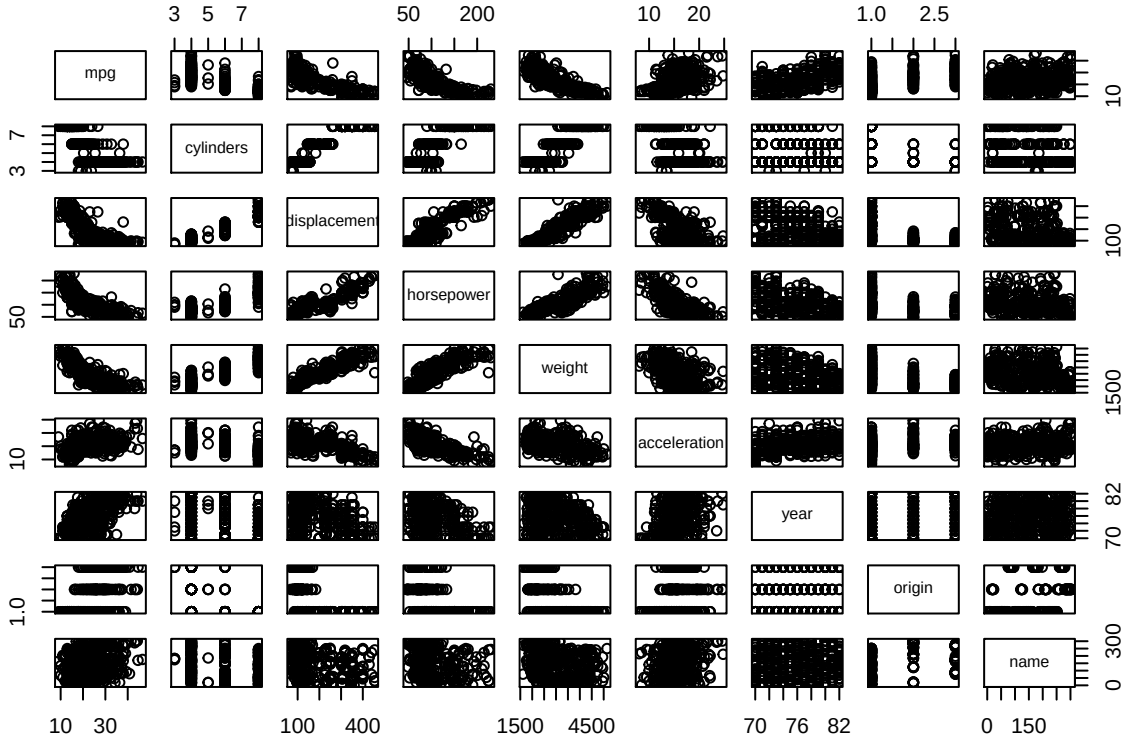
```
##          mpg    cylinders displacement   horsepower       weight
##    7.7951978    1.7004790  104.1406899   38.1763305  848.5120670
## acceleration         year       origin
##    2.7396716    3.6775556    0.8065221
```

(e)

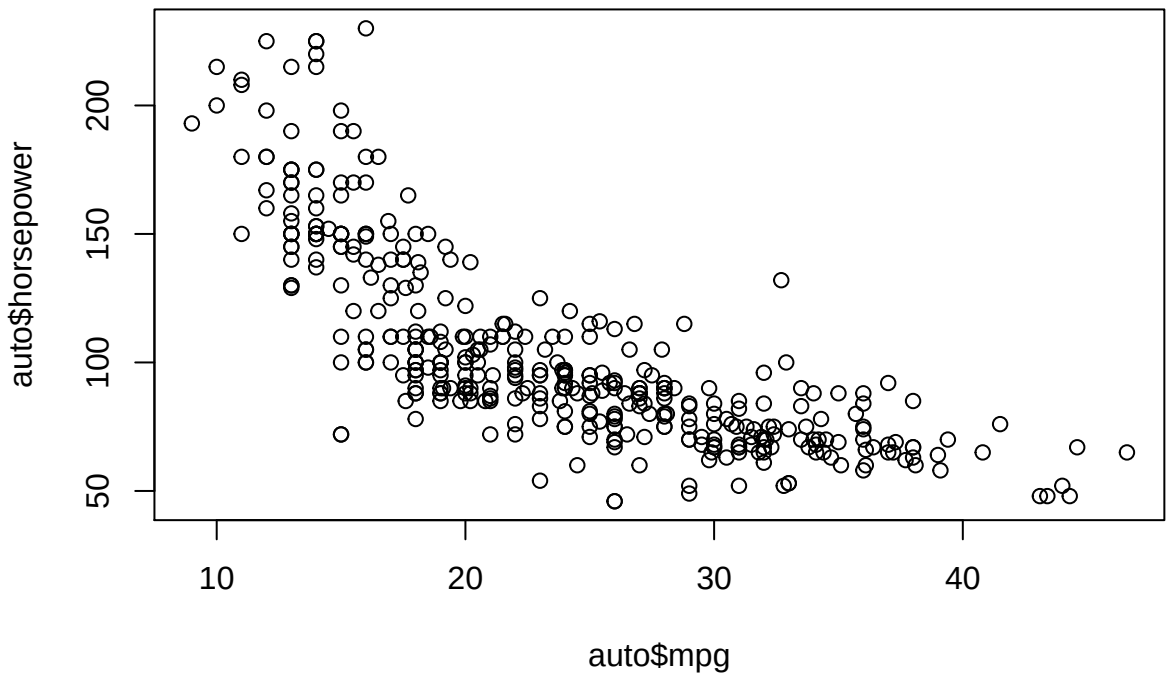
```r
library(vioplot)
```

```
## Loading required package: sm
## Package 'sm', version 2.2-5.4: type help(sm) for summary information
```

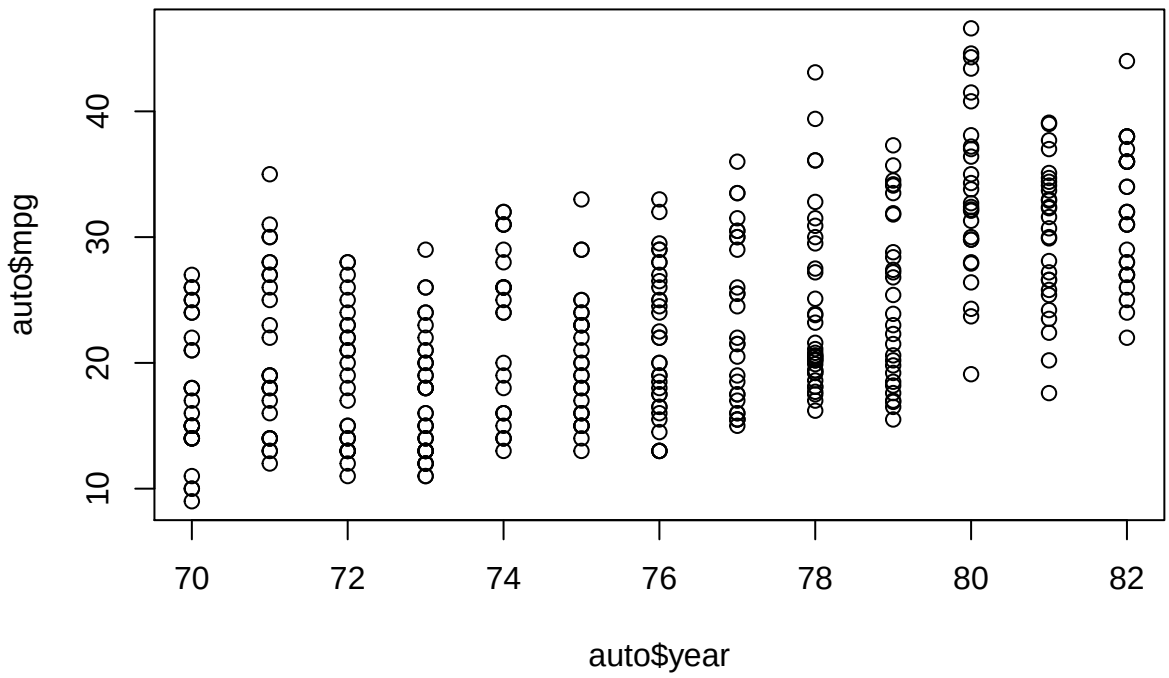
```r
pairs(auto)
```



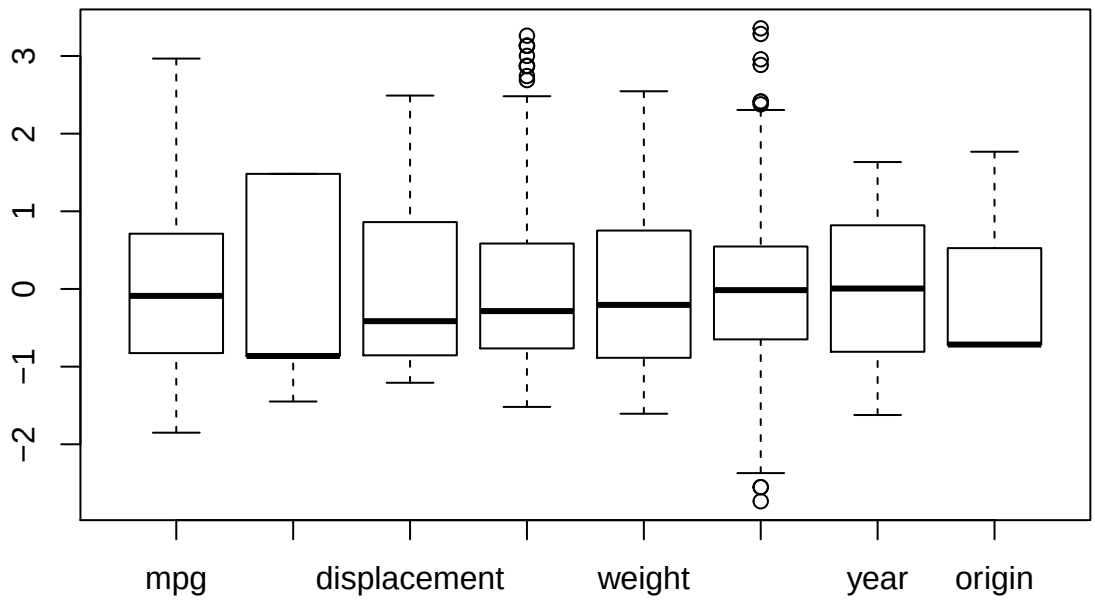

```r
plot(auto$mpg, auto$horsepower)
```



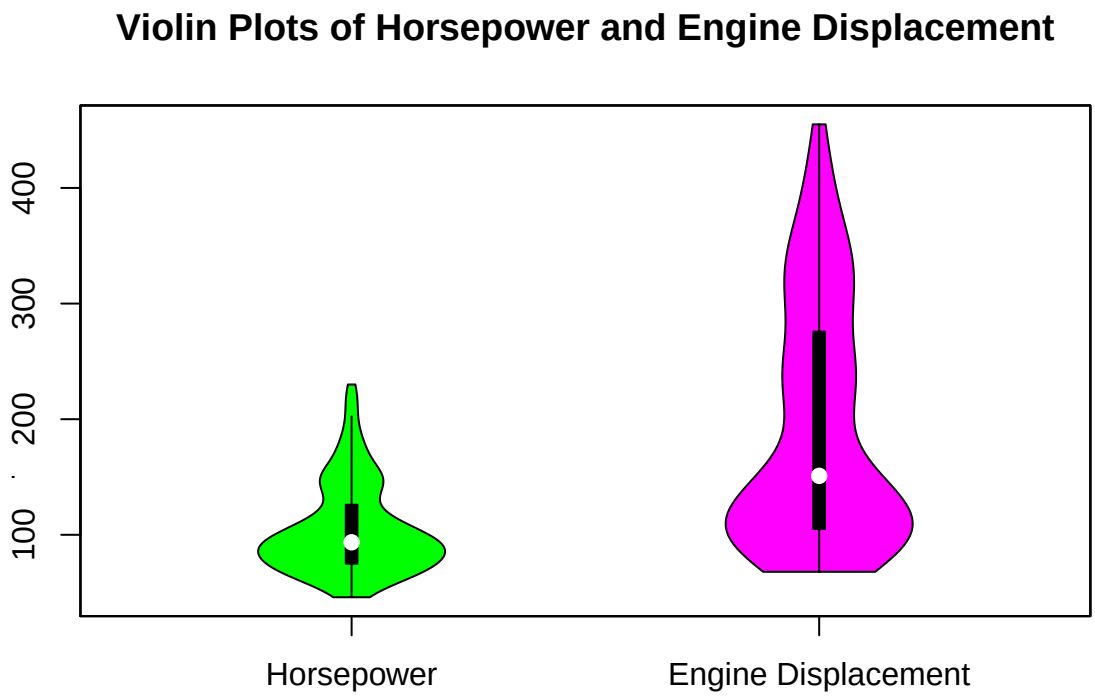

```
plot(auto$year, auto$mpg)
```




```
auto_mcs = scale(auto[numeric])#Mean center and scale data
boxplot(auto_mcs)
```




```
#Violin plots (blends kernel density plots with boxplots)
plot(1, 1, xlim = c(0, 4), ylim = range(c(auto$horsepower, auto$displacement)), type = 'n', xlab = '', y
vioplot(auto$horsepower, at = 1, add = T, col = 'green')
vioplot(auto$displacement, at = 3, add = T, col = 'magenta')
axis(1, at = c(1,3), labels = c('Horsepower', 'Engine Displacement'))
axis(2, at = 150, pos = -0.45, tck = 0, labels="")
title(main = 'Violin Plots of Horsepower and Engine Displacement')
```



We observe that horsepower is seemingly negatively correlated with miles per gallon. The average miles per gallon a car has seems to be increasing over the time period given. Acceleration and horsepower seem to have possible outliers.

(f) Suppose that we wish to predict gas mileage (mpg) on the basis of the other variables. Do your plots suggest that any of the other variables might be useful in predicting mpg? Justify your answer.

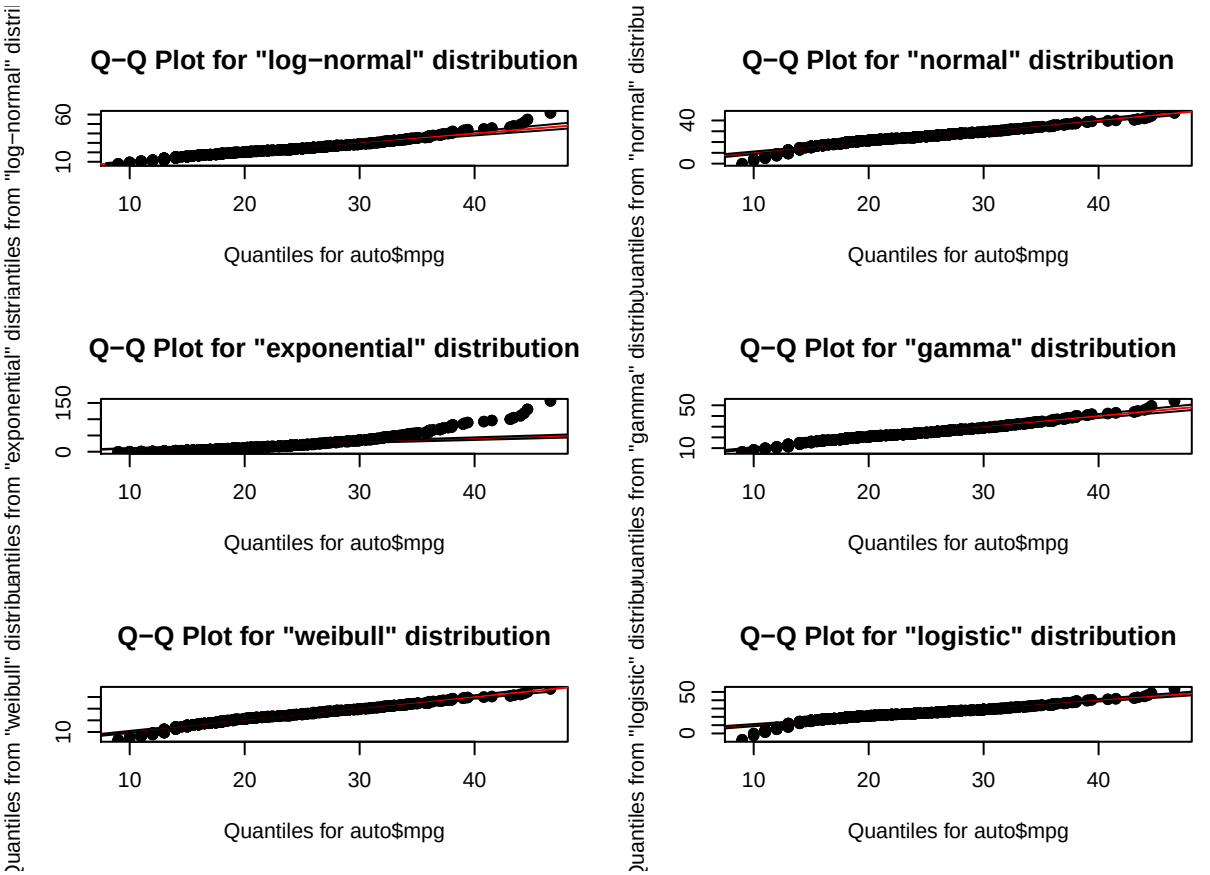
```
suppressMessages(library(qualityTools))
suppressMessages(library(corrplot))

# Quantile-Quantile plots for different distributions
par(mfrow = c(3,2))
qqPlot(auto$mpg, "log-normal")
qqPlot(auto$mpg, "normal")
qqPlot(auto$mpg, "exponential", DB = TRUE)
qqPlot(auto$mpg, "gamma")
qqPlot(auto$mpg, "weibull")
```

```
## Warning in densfun(x, parm[1], parm[2], ...): NaNs produced
```

```
## Warning in densfun(x, parm[1], parm[2], ...): NaNs produced
```

```
qqPlot(auto$mpg, "logistic")
```

```
## Warning in densfun(x, parm[1], parm[2], ...): NaNs produced
```




```
#Correlation matrix
print("Pearson correlation")
```

```
## [1] "Pearson correlation"
```

```
cor(auto[,1:8], method = "pearson")[,1]
```

```
##          mpg    cylinders displacement   horsepower       weight
##    1.0000000   -0.7776175   -0.8051269   -0.7784268   -0.8322442
## acceleration         year       origin
##    0.4233285    0.5805410    0.5652088
```

```
print("Kendall rank correlation")
```

```
## [1] "Kendall rank correlation"
```

```
cor(auto[,1:8], method = "kendall")[,1]
```

```
##          mpg    cylinders displacement   horsepower       weight
##    1.0000000   -0.6874442   -0.6785863   -0.6791658   -0.6942431
## acceleration         year       origin
##    0.3031459    0.4152465    0.4683835
```

We observe that the empirical distribution for mpg is close to normal but has fat tails at the lower range of its distribution. Our previous scatter plots of horsepower and year show negative and positive relationships respectively. We observe that all of the variables have Pearson correlations higher than 0.50 except for acceleration. Using the Kendall non-parametric rank correlation we find that only cylinders, horsepower and weight have correlations higher than 0.50. It is suggested most of the variables should be used in model building except for possibly acceleration.

10. This excercise involves the Boston housing data set.

(a)

```
library(MASS)

boston = Boston
rows = nrow(boston)
cols = ncol(boston)
print(paste0("There are ", rows, " rows and ", cols, " columns in the dataset"))
```

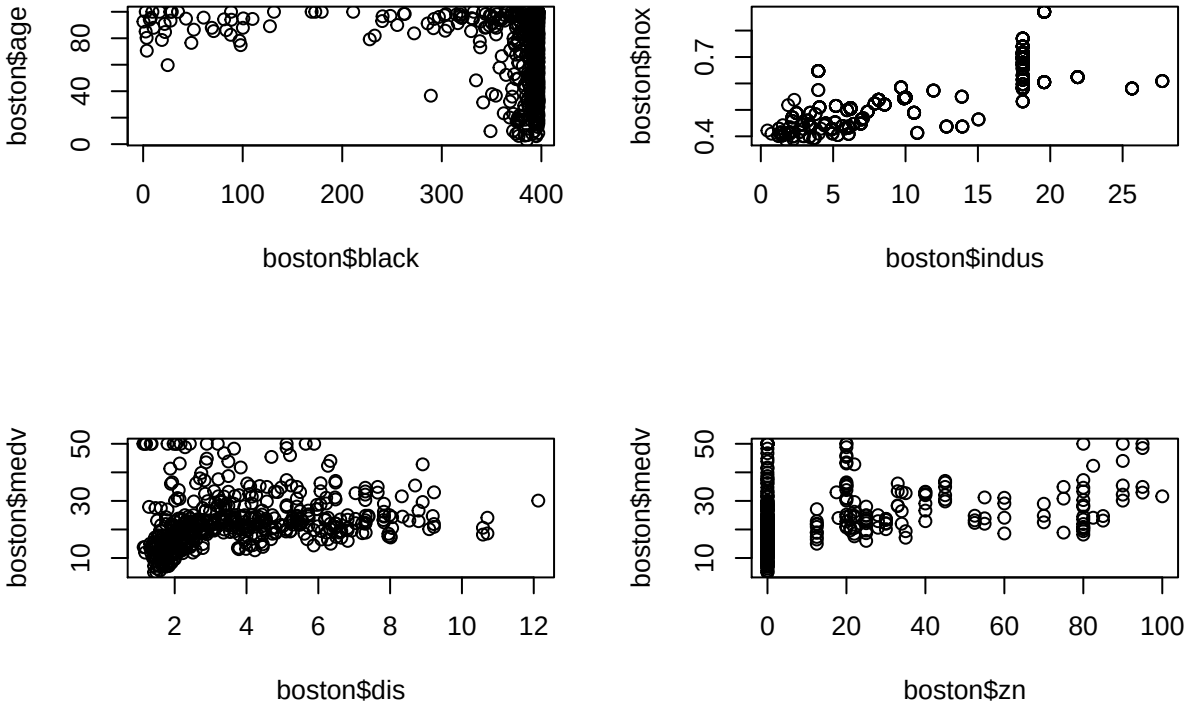
```
## [1] "There are 506 rows and 14 columns in the dataset"
```

```
print("The rows represent towns in Boston, while the columns are different characteristics of the towns
```

```
## [1] "The rows represent towns in Boston, while the columns are different characteristics of the towns
```

(b) Make some pairwise scatterplots of the predictors (columns) in this data set. Describe your findings.

```
par(mfrow= c(2,2))
plot(boston$black, boston$age)
plot(boston$indus, boston$nox)
plot(boston$dis, boston$medv)
plot(boston$zn, boston$medv)
```



The towns with a higher percentage black population have wider variation in the ages of the homes present in comparison to towns with lower percentages of blacks. Nitrogen oxide concentration appears to linearly

increase as the proportion of non-retail business acres per town increases. Median home values show wide variation across different weighted mean distances to the five Boston employment centers, however, a seeming majority of the higher priced and lower priced homes are found very close to the employment centers. This is not a surprising finding given that urban centers tend to have wide variation in housing prices reflecting a mix of luxury and subsidized housing. Finally median home values appear to show a slight increase as the proportion of residential land zoned for large lots increases.

(c)Are any of the predictors associated with per capita crime rate? If so, explain the relationship.

```
cor(boston, method = "pearson")[,1]
```

```
##        crim          zn       indus        chas         nox          rm
##  1.00000000 -0.20046922  0.40658341 -0.05589158  0.42097171 -0.21924670
##         age         dis         rad         tax     ptratio       black
##  0.35273425 -0.37967009  0.62550515  0.58276431  0.28994558 -0.38506394
##       lstat        medv
##  0.45562148 -0.38830461
```

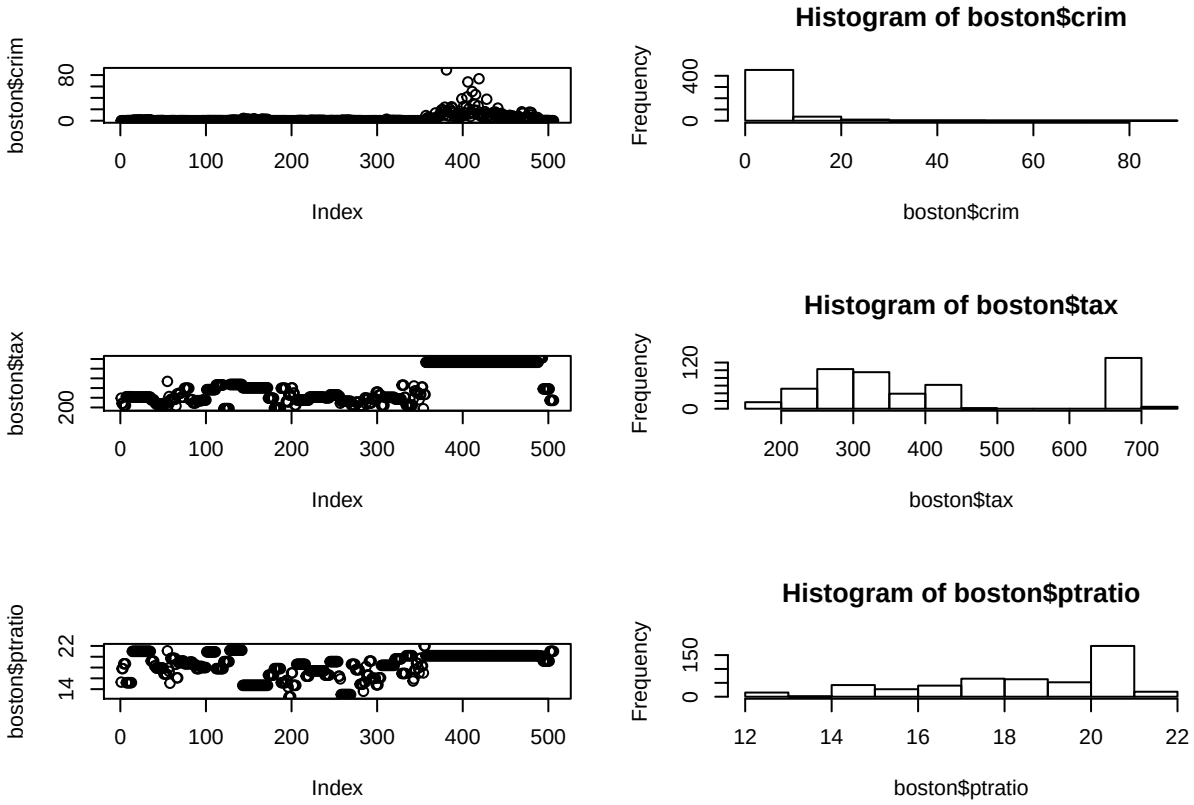
```
cor(boston, method = "kendall")[,1]
```

```
##        crim          zn       indus        chas         nox          rm
##  1.00000000 -0.46205658  0.52101384  0.03394849  0.60336119 -0.21171838
##         age         dis         rad         tax     ptratio       black
##  0.49729673 -0.53987837  0.56396872  0.54495622  0.31276768 -0.26437758
##       lstat        medv
##  0.45483655 -0.40396356
```

We observe that crime is appears to be correlated (correlation coefficient above 0.50) with higher nitrogen oxide concentrations, accessible radial highways, higher full-value property-tax rates, and towns closer to the five Boston employment centers.

(d)

```
#Scatter plots and histograms of crime rate, tax , and pupil teacher ratio
par(mfrow= c(3,2))
plot(boston$crim)
hist(boston$crim)
plot(boston$tax)
hist(boston$tax)
plot(boston$ptratio)
hist(boston$ptratio)
```



(d) Do any of the suburbs of Boston appear to have particularly high crime rates? Tax rates? Pupil-teacher ratios? Comment on the range of each predictor.

```
#Get back the rows that have variables in the 99th percentile of the data
print("Suburbs with crime rates in the 99th percentile")
```

```
## [1] "Suburbs with crime rates in the 99th percentile"
```

```
boston[which(boston$crim > as.numeric(quantile(boston$crim,.99))),]
```

```
##        crim zn indus chas   nox    rm   age    dis rad tax ptratio  black
## 381 88.9762  0  18.1    0 0.671 6.968  91.9 1.4165  24 666    20.2 396.90
## 405 41.5292  0  18.1    0 0.693 5.531  85.4 1.6074  24 666    20.2 329.46
## 406 67.9208  0  18.1    0 0.693 5.683 100.0 1.4254  24 666    20.2 384.97
## 411 51.1358  0  18.1    0 0.597 5.757 100.0 1.4130  24 666    20.2   2.60
## 415 45.7461  0  18.1    0 0.693 4.519 100.0 1.6582  24 666    20.2  88.27
## 419 73.5341  0  18.1    0 0.679 5.957 100.0 1.8026  24 666    20.2  16.45
##     lstat medv
## 381 17.21 10.4
## 405 27.38  8.5
## 406 22.98  5.0
## 411 10.11 15.0
## 415 36.98  7.0
## 419 20.62  8.8
```

```
print("Suburbs with tax rates in the 99th percentile")
```

```
## [1] "Suburbs with tax rates in the 99th percentile"
```

```
boston[which(boston$tax > as.numeric(quantile(boston$tax,.99))),]
```

```
##        crim zn indus chas   nox    rm  age    dis rad tax ptratio  black
## 489 0.15086  0 27.74    0 0.609 5.454 92.7 1.8209   4 711    20.1 395.09
## 490 0.18337  0 27.74    0 0.609 5.414 98.3 1.7554   4 711    20.1 344.05
## 491 0.20746  0 27.74    0 0.609 5.093 98.0 1.8226   4 711    20.1 318.43
## 492 0.10574  0 27.74    0 0.609 5.983 98.8 1.8681   4 711    20.1 390.11
## 493 0.11132  0 27.74    0 0.609 5.983 83.5 2.1099   4 711    20.1 396.90
##     lstat medv
## 489 18.06 15.2
## 490 23.97  7.0
## 491 29.68  8.1
## 492 18.07 13.6
## 493 13.35 20.1
```

```
print("Suburbs with pupil-teacher ratios in the 99th percentile")
```

```
## [1] "Suburbs with pupil-teacher ratios in the 99th percentile"
```

```
boston[which(boston$ptratio > as.numeric(quantile(boston$ptratio,.99))),]
```

```
##        crim zn indus chas   nox    rm  age     dis rad tax ptratio  black
## 355 0.04301 80  1.91    0 0.413 5.663 21.9 10.5857   4 334      22 382.80
## 356 0.10659 80  1.91    0 0.413 5.936 19.5 10.5857   4 334      22 376.04
##     lstat medv
## 355  8.05 18.2
## 356  5.57 20.6
```

We observe that there are 7, 5 and 2 suburbs that have crime, tax rates, and pupil-teacher ratios in the 99th percentile respectively.

(e) How many of the suburbs in this data set bound the Charles river?

```
cr = table(boston$chas)[2]
print(paste0(cr, " suburbs bound the Charles River."))
```

```
## [1] "35 suburbs bound the Charles River."
```

(f) What is the median pupil-teacher ratio among the towns in this data set?

```
med_pt = quantile(boston$ptratio,.50)
print(paste0("The median pupil-teacher ratio among the town is ", med_pt))
```

```
## [1] "The median pupil-teacher ratio among the town is 19.05"
```

(g) Which suburb of Boston has lowest median value of owner-occupied homes? What are the values of the other predictors for that suburb, and how do those values compare to the overall ranges for those predictors? Comment on your findings.

```
#Identify towns with lowest median value of owner-occupied homes
boston[which(boston$medv == min(boston$medv)),]
```

```
##        crim zn indus chas   nox    rm age    dis rad tax ptratio  black
## 399 38.3518  0  18.1    0 0.693 5.453 100 1.4896  24 666    20.2 396.90
## 406 67.9208  0  18.1    0 0.693 5.683 100 1.4254  24 666    20.2 384.97
```

```
##     lstat medv
## 399 30.59    5
## 406 22.98    5
```

```
#Create a new dataframe with each value now a percentile of their empirical distributions
perc.rank = function(x) trunc(rank(x))/length(x)
boston_perc = sapply(boston, perc.rank)
boston_perc = data.frame(boston_perc)
colnames(boston_perc) = names(boston)

#Identify towns with lowest median value of owner-occupied homes this time with the transformed variabl
boston_perc[which(boston_perc$medv == min(boston_perc$medv)),]
```

```
##          crim        zn    indus      chas       nox        rm      age
## 399 0.9881423 0.3675889 0.756917 0.4664032 0.8438735 0.0770751 0.958498
## 406 0.9960474 0.3675889 0.756917 0.4664032 0.8438735 0.1363636 0.958498
##            dis       rad       tax   ptratio     black     lstat
## 399 0.05731225 0.8695652 0.8596838 0.7509881 0.8814229 0.9782609
## 406 0.04150198 0.8695652 0.8596838 0.7509881 0.3498024 0.8992095
##            medv
## 399 0.001976285
## 406 0.001976285
```

Towns 399 and 406 have the lowest median value of owner-occupied homes. We observe that these towns are in the upper percentiles (variable > 0.75) for towns with the proportion of non-retail business acres per town, nitrogen oxides concentration, proportion of owner-occupied units built prior to 1940, accessibility to radial highways, full-value property-tax rate, pupil-teacher ratio, and lower status of the population. These towns are in the lower percentiles (variable < 0.25) for distance to the five Boston employment centers (most likely urban towns) and average number of rooms per dwelling.

(h) In this data set, how many of the suburbs average more than seven rooms per dwelling? More than eight rooms per dwelling? Comment on the suburbs that average more than eight rooms per dwelling.

```
#Boston towns that average more than 7 and 8 rooms per dwelling
boston_rm7 = boston[which(boston$rm > 7),]
boston_rm8 = boston[which(boston$rm > 8),]
rm7 = nrow(boston_rm7)
rm8 = nrow(boston_rm8)
print(paste0(rm7, " suburbs average seven or more rooms per dwelling and ", rm8, " average eight or more
```

```
## [1] "64 suburbs average seven or more rooms per dwelling and 13 average eight or more rooms per dwell
```

```
#Towns characteristics with 8 or more rooms per dwelling
summary(boston_perc[which(boston$rm > 8),])
```

```
##       crim                zn              indus              chas
##  Min.   :0.03557   Min.   :0.3676   Min.   :0.08893   Min.   :0.4664
##  1st Qu.:0.53953   1st Qu.:0.3676   1st Qu.:0.16996   1st Qu.:0.4664
##  Median :0.58498   Median :0.3676   Median :0.33992   Median :0.4664
##  Mean   :0.53816   Mean   :0.5458   Mean   :0.33521   Mean   :0.5433
##  3rd Qu.:0.60672   3rd Qu.:0.7806   3rd Qu.:0.33992   3rd Qu.:0.4664
##  Max.   :0.74308   Max.   :0.9941   Max.   :0.91700   Max.   :0.9664
##       nox                rm               age               dis
##  Min.   :0.08893   Min.   :0.9763   Min.   :0.01976   Min.   :0.1502
##  1st Qu.:0.38735   1st Qu.:0.9822   1st Qu.:0.43874   1st Qu.:0.3123
```

```
##  Median :0.40514   Median :0.9881   Median :0.50988   Median :0.4625
##  Mean   :0.47431   Mean   :0.9881   Mean   :0.47963   Mean   :0.4716
##  3rd Qu.:0.68775   3rd Qu.:0.9941   3rd Qu.:0.60672   3rd Qu.:0.5652
##  Max.   :0.92095   Max.   :1.0000   Max.   :0.74506   Max.   :0.9783
##       rad               tax             ptratio           black
##  Min.   :0.06324   Min.   :0.07708   Min.   :0.01779   Min.   :0.1779
##  1st Qu.:0.49407   1st Qu.:0.16601   1st Qu.:0.06719   1st Qu.:0.3478
##  Median :0.67589   Median :0.41502   Median :0.26680   Median :0.3854
##  Mean   :0.57175   Mean   :0.37382   Mean   :0.24324   Mean   :0.4398
##  3rd Qu.:0.71542   3rd Qu.:0.41502   3rd Qu.:0.26680   3rd Qu.:0.4368
##  Max.   :0.86957   Max.   :0.85968   Max.   :0.75099   Max.   :0.8814
##      lstat              medv
##  Min.   :0.007905   Min.   :0.5455
##  1st Qu.:0.035573   1st Qu.:0.9427
##  Median :0.071146   Median :0.9644
##  Mean   :0.089845   Mean   :0.9319
##  3rd Qu.:0.132411   3rd Qu.:0.9842
##  Max.   :0.278656   Max.   :0.9842
```

The towns that average 8 or more rooms per house tend to have higher median home values and a lower percentage of lower status population.