Author: Yang Yue

Practice 2.

2020-07-03

Question 1 (based on JWHT Chapter 2, Problem 9)

Use the Auto data set from the textbook's website. When reading the data, use the options as.is = TRUE and na.strings="?". Remove the unavailable data using the na.omit() function.

```r
#insert r code here

Auto=read.csv(file="C:/Users/Administrator/Desktop/Auto.csv", header=T,na.strings="?")
Auto=na.omit(Auto)
head(Auto)
```

```
##   mpg cylinders displacement horsepower weight acceleration year origin
## 1  18         8          307        130   3504         12.0   70      1
## 2  15         8          350        165   3693         11.5   70      1
## 3  18         8          318        150   3436         11.0   70      1
## 4  16         8          304        150   3433         12.0   70      1
## 5  17         8          302        140   3449         10.5   70      1
## 6  15         8          429        198   4341         10.0   70      1
##                        name
## 1 chevrolet chevelle malibu
## 2         buick skylark 320
## 3        plymouth satellite
## 4             amc rebel sst
## 5               ford torino
## 6          ford galaxie 500
```

1. List the names of the variables in the data set.

```r
#insert r code here
names(Auto)
```

```
## [1] "mpg"          "cylinders"    "displacement" "horsepower"   "weight"
## [6] "acceleration" "year"         "origin"       "name"
```

2. The columns origin and name are unimportant variables. Create a new data frame called cars that contains none of these unimportant variables

```r
#insert r code here
cars= Auto[-c(8:9)]
names(cars)
```

```
## [1] "mpg"          "cylinders"    "displacement" "horsepower"   "weight"
## [6] "acceleration" "year"
```

3. What is the range of each quantitative variable? Answer this question using the range() function with the sapply() function (e.g., sapply(cars, range). Print a simple table of the ranges of the variables.The columns should be suitably labeled.

```r
#insert r code here
sapply(cars, range)
```

```
##       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
```

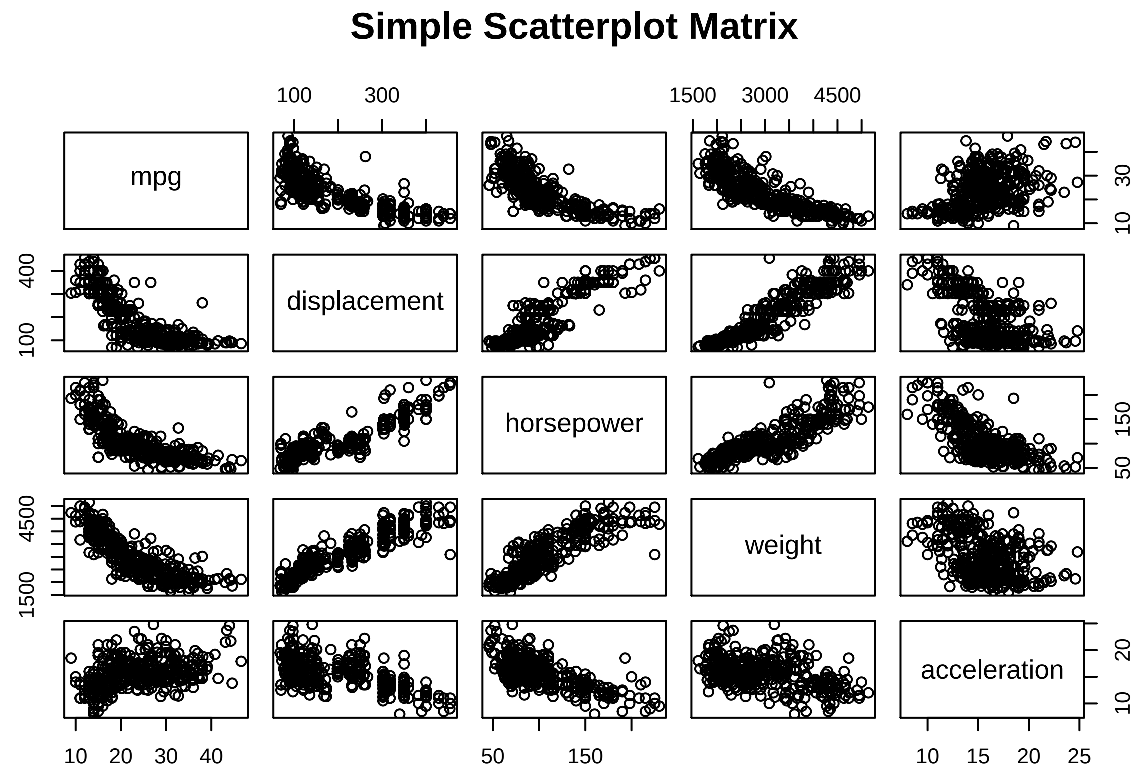
4. What is the mean and standard deviation of each variable? Create a simple table of the means and standard deviations.

```r
#insert r code here
Mean=sapply(cars, mean)
SD=sapply(cars,sd)
table=cbind(Mean,SD)
table
```

```
##                     Mean         SD
## mpg            23.445918   7.805007
## cylinders       5.471939   1.705783
## displacement  194.411990 104.644004
## horsepower    104.469388  38.491160
## weight       2977.584184 849.402560
## acceleration   15.541327   2.758864
## year           75.979592   3.683737
```

5. Create a scatterplot matrix that includes the variables mpg, displacement, horsepower, weight, and acceleration using the pairs() function.

```r
#insert r code here
pairs(~mpg+displacement+horsepower+weight+acceleration,data=cars, main="Simple Scatterplot Matrix")
```



6. From the scatterplot, it should be clear that mpg has an almost linear relationship to predictors, and higher-order relationships to other variables. Using the regsubsets function in the leaps library, regress mpg onto

- displacement
- displacement squared
- horsepower
- horsepower squared
- weight
- weight squared
- acceleration

```
#insert r code here
library(leaps)
dissq=cars$displacement^2
horsq=cars$horsepower^2
weisq=cars$weight^2
regfit.full=regsubsets(mpg ~ +dissq+horsepower+horsq+weight+
                         weisq+acceleration,cars,nvmax = NULL)
summary(regfit.full)
```

```
## Subset selection object
## Call: regsubsets.formula(mpg ~ +dissq + horsepower + horsq + weight +
##     weisq + acceleration, cars, nvmax = NULL)
## 6 Variables  (and intercept)
##              Forced in Forced out
## dissq            FALSE      FALSE
## horsepower       FALSE      FALSE
## horsq            FALSE      FALSE
## weight           FALSE      FALSE
## weisq            FALSE      FALSE
## acceleration     FALSE      FALSE
## 1 subsets of each size up to 6
## Selection Algorithm: exhaustive
##          dissq horsepower horsq weight weisq acceleration
## 1  ( 1 ) " "   " "        " "   "*"    " "   " "
## 2  ( 1 ) " "   " "        " "   "*"    "*"   " "
## 3  ( 1 ) " "   "*"        "*"   "*"    " "   " "
## 4  ( 1 ) " "   "*"        "*"   "*"    "*"   " "
## 5  ( 1 ) " "   "*"        "*"   "*"    "*"   "*"
## 6  ( 1 ) "*"   "*"        "*"   "*"    "*"   "*"
```

Print a table showing what variables would be selected using best subset selection for all model orders.

```r
#insert r code here
reg.summary=summary(regfit.full)
reg.summary$which
```

```
##   (Intercept) dissq horsepower horsq weight weisq acceleration
## 1        TRUE FALSE      FALSE FALSE   TRUE FALSE        FALSE
## 2        TRUE FALSE      FALSE FALSE   TRUE  TRUE        FALSE
## 3        TRUE FALSE       TRUE  TRUE   TRUE FALSE        FALSE
## 4        TRUE FALSE       TRUE  TRUE   TRUE  TRUE        FALSE
## 5        TRUE FALSE       TRUE  TRUE   TRUE  TRUE         TRUE
## 6        TRUE  TRUE       TRUE  TRUE   TRUE  TRUE         TRUE
```

What is the most important variable affecting fuel consumption?

```r
# The most important variable affecting fuel consumption is weight
```

What is the second most important variable affecting fuel consumption?

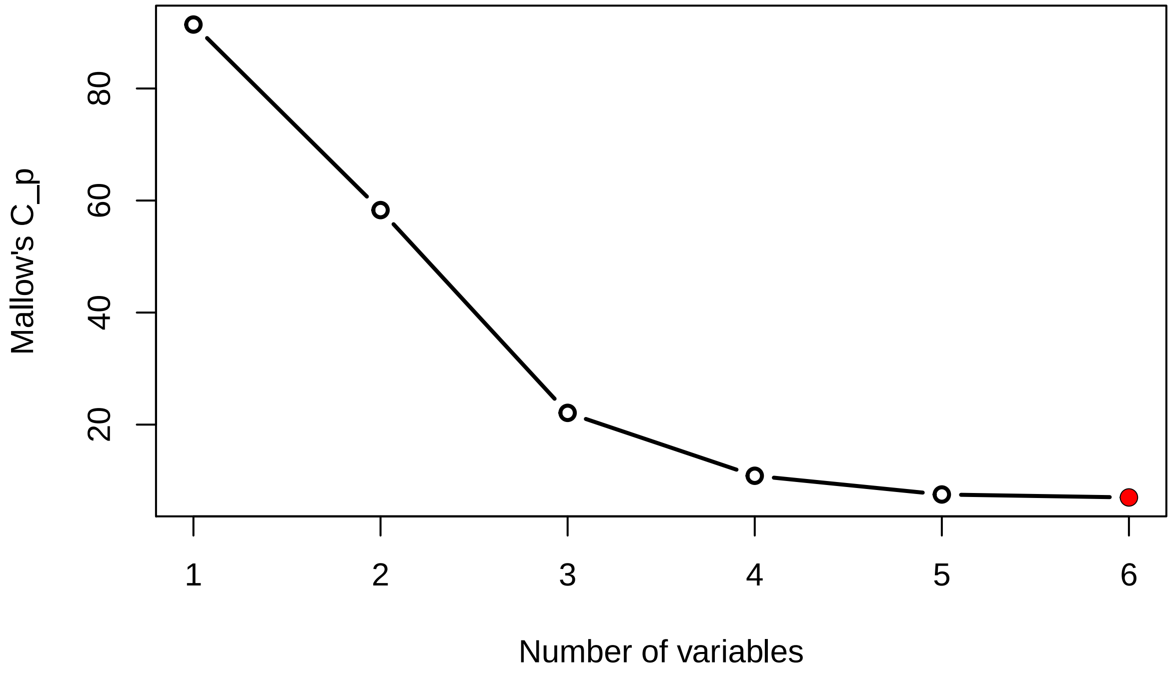
```r
#The second most important variable affecting fuel consumption is horsepower
```

What is the third most important variable affecting fuel consumption?

```r
#The third most important variable affecting fuel consumption is acceleration
```

7. Plot a graph showing Mallow's Cp as a function of the order of the model. Which model is the best?

```r
#insert r code here
cp=reg.summary$cp
p=which.min(cp)
plot(cp,xlab = "Number of variables", ylab = "Mallow's C_p", type = "b",lwd=2)
points(p,cp[p],col='red',pch=19)
```



If we apply Mallow'cp as a function of the order of the model, 6-variables model is the best.

Question 2 (based on JWHT Chapter 3, Problem 10)

This exercise involves the Boston housing data set.

1. Load in the Boston data set, which is part of the MASS library in R. The data set is contained in the object Boston. Read about the data set using the command ?Boston. How many rows are in this data set? How many columns? What do the rows and columns represent?

```r
#insert r code here
library(MASS)
data("Boston")
dim(Boston)
```

```
## [1] 506  14
```

```
names(Boston)
```

```
##  [1] "crim"    "zn"      "indus"   "chas"    "nox"     "rm"      "age"
##  [8] "dis"     "rad"     "tax"     "ptratio" "black"   "lstat"   "medv"
```

```
?Boston
```

There are 506 rows and 14 columns in Boston data set.Each column represents the predictor variable for 506 suburbs of Boston. Each row represents the predictor observations given by suburbs of Boston.

2. How many of the suburbs in this data set bound the Charles river?

```
#insert r code here
table(Boston$chas)
```

```
##
##   0   1
## 471  35
```

There are 35 suburbs in this data set bound the Charles river.

3. What is the median pupil-teacher ratio among the towns in this data set?

```
#insert r code here
summary(Boston$ptratio)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   12.60   17.40   19.05   18.46   20.20   22.00
```

The median pupil-teacher ration among the town in this data set is 19 puipls per teacher

4. In this data set, how many of the suburbs average more than seven rooms per dwelling?

```
#insert r code here
sum(Boston$rm>7)
```

```
## [1] 64
```

There are 64 suburbs average more than seven rooms per dewlling.

More than eight rooms per dwelling?

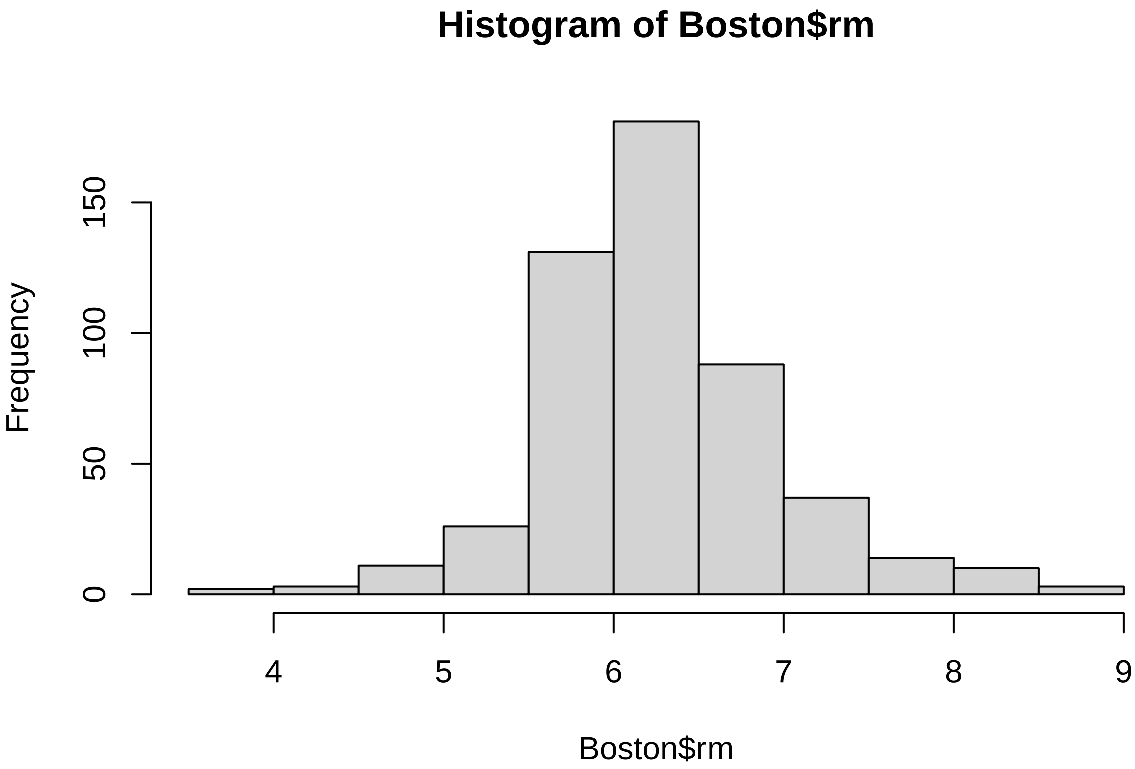
```
#insert r code here
sum(Boston$rm>8)
```

```
## [1] 13
```

```
summary(Boston$rm)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.561   5.886   6.208   6.285   6.623   8.780
```

```
hist(Boston$rm)
```



Comment on the suburbs that average more than eight rooms per dwelling.

There are 13 suburbs average more than eight rooms per dewlling. Most of the suburbs have average around 5.5 to 7 rooms per dlling.

Question 3 (based on JWHT Chapter 4, Problem 10)

This question should be answered using the Weekly data set, which is part of the ISLR package. This data contains 1,089 weekly returns for 21 years, from the beginning of 1990 to the end of 2010.

1. What does the data represent?

```
#insert r code here
library(’ISLR’)
attach(Weekly)
?Weekly
```

The data represents Weekly percentage returns for the S&P 500 stock index between 1990 and 2010.

2. Use the full data set to perform a logistic regression with Direction as the response and the five lag variables plus Volume as predictors. Use the summary function to print the results. Do any of the predictors appear to be statistically significant? If so, which ones?

```
#insert r code here
glm.fit=glm(Direction~Lag1+Lag2+Lag3+Lag4+Lag5+Volume,family=binomial,data=Weekly)
summary(glm.fit)
```

```
##
## Call:
## glm(formula = Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 +
##     Volume, family = binomial, data = Weekly)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6949  -1.2565   0.9913   1.0849   1.4579
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.26686    0.08593   3.106   0.0019 **
## Lag1        -0.04127    0.02641  -1.563   0.1181
## Lag2         0.05844    0.02686   2.175   0.0296 *
## Lag3        -0.01606    0.02666  -0.602   0.5469
## Lag4        -0.02779    0.02646  -1.050   0.2937
## Lag5        -0.01447    0.02638  -0.549   0.5833
## Volume      -0.02274    0.03690  -0.616   0.5377
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1496.2  on 1088  degrees of freedom
## Residual deviance: 1486.4  on 1082  degrees of freedom
## AIC: 1500.4
##
## Number of Fisher Scoring iterations: 4
```

lag2 appears to be statistically significant.

3. Fit a logistic regression model using a training data period from 1990 to 2008, with Lag2 as the only predictor. Compute the confusion matrix and the overall fraction of correct predictions for the held out data (that is, the data from 2009 and 2010).

```r
#insert r code here
train=(Weekly$Year<2009)
Weekly.2009=Weekly[!train,]
contrasts(Weekly$Direction)
```

```
##      Up
## Down  0
## Up    1
```

```r
dim(Weekly.2009)
```

```
## [1] 104   9
```

```r
glm.fit=glm(Direction~Lag2,data=Weekly,family = binomial,subset =train)
glm.probs=(predict(glm.fit,Weekly.2009,type="response"))
glm.pred=rep("Down",104)
glm.pred[glm.probs > 0.5]="Up"
table(glm.pred,Weekly.2009$Direction)
```

```
##
## glm.pred Down Up
##     Down    9  5
##     Up     34 56
```

```r
Logistic=mean(glm.pred==Weekly.2009$Direction)
Logistic
```

```
## [1] 0.625
```

4. Repeat Part 3 using LDA.

```r
#insert r code here
library(MASS)
lda.fit=lda(Direction~Lag2,data=Weekly,subset=train)
lda.pred=predict(lda.fit,Weekly.2009)
names(lda.pred)
```

```
## [1] "class"     "posterior" "x"
```

```r
lda.class=lda.pred$class
table(lda.class,Weekly.2009$Direction)
```

```
##
## lda.class Down Up
##      Down    9  5
##      Up     34 56
```

```
LDA=mean(lda.class==Weekly.2009$Direction)
LDA
```

```
## [1] 0.625
```

5. Repeat Part 3 using QDA.

```
#insert r code here
qda.fit=qda(Direction~Lag2,data=Weekly,subset=train)
qda.class=predict(qda.fit,Weekly.2009)$class
table(qda.class,Weekly.2009$Direction)
```

```
##
## qda.class Down Up
##      Down    0  0
##      Up     43 61
```

```
QDA=mean(qda.class==Weekly.2009$Direction)
QDA
```

```
## [1] 0.5865385
```

6. Repeat Part 3 using KNN with K = 1, 2, 3.

```
#insert r code here
library(class)
train.X=Weekly[train,"Lag2",drop=FALSE]
test.X=Weekly[!train,"Lag2",drop=FALSE]
train.Direction=Weekly[train,"Direction",drop=TRUE]
test.Direction=Weekly[!train,"Direction",drop=TRUE]
set.seed(1)
knn.pred=knn(train.X,test.X,train.Direction,k=1)
table(knn.pred,test.Direction)
```

```
##         test.Direction
## knn.pred Down Up
##     Down   21 30
##     Up     22 31
```

```
k1=mean(knn.pred==test.Direction)
k1
```

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

```r
knn.pred=knn(train.X,test.X,train.Direction,k=2)
table(knn.pred,test.Direction)
```

```
##         test.Direction
## knn.pred Down Up
##     Down   18 25
##     Up     25 36
```

```r
k2=mean(knn.pred==test.Direction)
k2
```

```
## [1] 0.5192308
```

```r
knn.pred=knn(train.X,test.X,train.Direction,k=3)
table(knn.pred,test.Direction)
```

```
##         test.Direction
## knn.pred Down Up
##     Down   16 20
##     Up     27 41
```

```r
k3=mean(knn.pred==test.Direction)
k3
```

```
## [1] 0.5480769
```

7. Which of these methods in Parts 3, 4, 5, and 6 appears to provide the best results on this data?

```r
#insert r code here
bestmethod=cbind(Logistic,LDA,QDA,k1,k2,k3)
bestmethod
```

```
##      Logistic   LDA       QDA  k1        k2        k3
## [1,]    0.625 0.625 0.5865385 0.5 0.5192308 0.5480769
```

Logistic Regression and LDA appears to provide the best results on this data