IDA LAB-1

SOLUTION FOR Practical-2, Practical-3, and Practical-4

Practical-2:

- This Practical relates to the **College** data set, which can be found in the file **College.csv**. It contains a number of variables for 777 different universities and colleges in the US. The variables are

  - Private : Public/private indicator
  - Apps : Number of applications received
  - Accept : Number of applicants accepted
  - Enroll : Number of new students enrolled
  - Top10perc : New students from top 10 % of high school class
  - Top25perc : New students from top 25 % of high school class
  - F.Undergrad : Number of full-time undergraduates
  - P.Undergrad : Number of part-time undergraduates
  - Outstate : Out-of-state tuition
  - Room.Board : Room and board costs
  - Books : Estimated book costs
  - Personal : Estimated personal spending
  - PhD : Percent of faculty with Ph.D.'s
  - Terminal : Percent of faculty with terminal degree
  - S.F.Ratio : Student/faculty ratio
  - perc.alumni : Percent of alumni who donate
  - Expend : Instructional expenditure per student
  - Grad.Rate : Graduation rate

Before reading the data into R, it can be viewed in Excel or a text editor.

(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.

```
college = read.csv("College.csv")
# attach(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. Try the following commands:

```
> rownames(college)=college[,1]
> fix(college)
```

You should see that there is now a row.names column with the name of each university recorded. This means that R has given each row a name corresponding to the appropriate university. R will not try to perform calculations on the row names. However, we still need to eliminate the first column in the data where the names are stored. Try

```
> rownames(college)=college[,-1]
> fix(college)
```

Now you should see that the first data column is Private. Note that another column labeled row.names now appears before the Private column. However, this is not a data column but rather

the name that R is giving to each row.

fix function
{r, results='hide'}

Add row names
rownames(college)=college[,1] ### Set rownames equal to first column

Remove first column
Remove column 1 because we assigned to rownames.
# college[,-1]
college=college[,-1] #

(c)

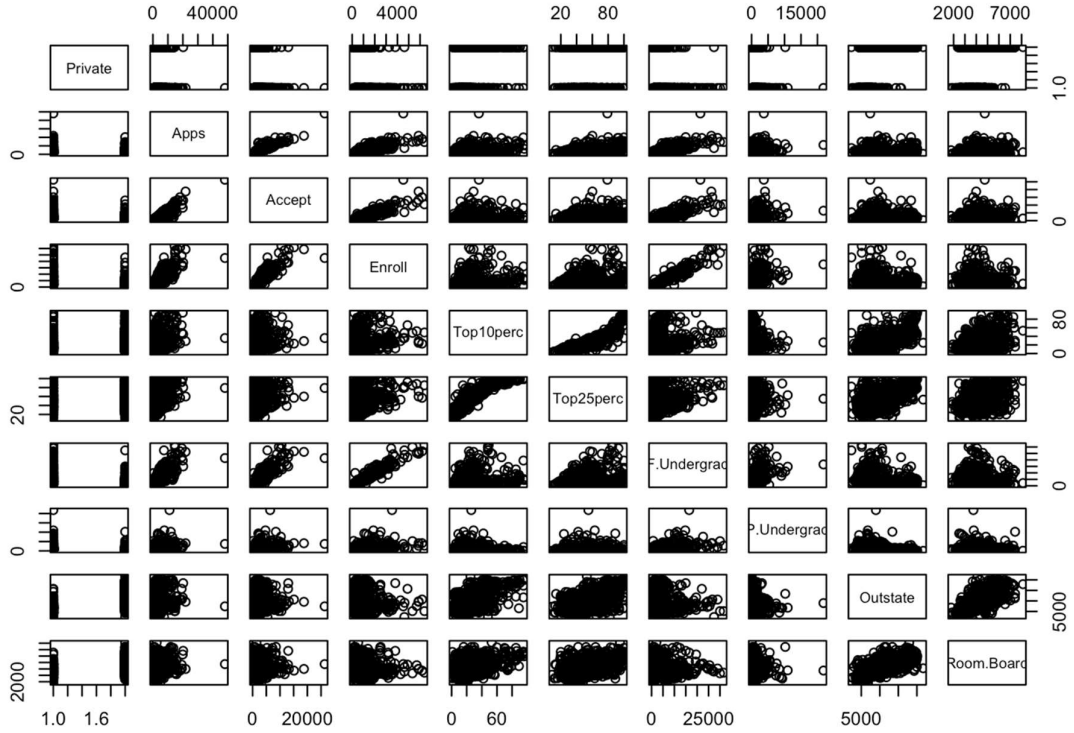
(i) Use the summary() function to produce a numerical summary of the variables in the data set.

```
summary(college)
##    Private             Apps           Accept          Enroll
##  Length:777         Min.   :   81   Min.   :   72   Min.   :  35
##  Class :character   1st Qu.:  776   1st Qu.:  604   1st Qu.: 242
##  Mode  :character   Median : 1558   Median : 1110   Median : 434
##                     Mean   : 3002   Mean   : 2019   Mean   : 780
##                     3rd Qu.: 3624   3rd Qu.: 2424   3rd Qu.: 902
##                     Max.   :48094   Max.   :26330   Max.   :6392
##    Top10perc       Top25perc      F.Undergrad     P.Undergrad
##  Min.   : 1.00   Min.   :  9.0   Min.   :  139   Min.   :    1.0
##  1st Qu.:15.00   1st Qu.: 41.0   1st Qu.:  992   1st Qu.:   95.0
##  Median :23.00   Median : 54.0   Median : 1707   Median :  353.0
##  Mean   :27.56   Mean   : 55.8   Mean   : 3700   Mean   :  855.3
##  3rd Qu.:35.00   3rd Qu.: 69.0   3rd Qu.: 4005   3rd Qu.:  967.0
##  Max.   :96.00   Max.   :100.0   Max.   :31643   Max.   :21836.0
##     Outstate       Room.Board      Books           Personal
##  Min.   : 2340   Min.   :1780   Min.   :  96.0   Min.   : 250
##  1st Qu.: 7320   1st Qu.:3597   1st Qu.: 470.0   1st Qu.: 850
##  Median : 9990   Median :4200   Median : 500.0   Median :1200
##  Mean   :10441   Mean   :4358   Mean   : 549.4   Mean   :1341
##  3rd Qu.:12925   3rd Qu.:5050   3rd Qu.: 600.0   3rd Qu.:1700
##  Max.   :21700   Max.   :8124   Max.   :2340.0   Max.   :6800
##       PhD            Terminal       S.F.Ratio      perc.alumni
##  Min.   :  8.00   Min.   : 24.0   Min.   : 2.50   Min.   : 0.00
##  1st Qu.: 62.00   1st Qu.: 71.0   1st Qu.:11.50   1st Qu.:13.00
##  Median : 75.00   Median : 82.0   Median :13.60   Median :21.00
##  Mean   : 72.66   Mean   : 79.7   Mean   :14.09   Mean   :22.74
##  3rd Qu.: 85.00   3rd Qu.: 92.0   3rd Qu.:16.50   3rd Qu.:31.00
##  Max.   :103.00   Max.   :100.0   Max.   :39.80   Max.   :64.00
##      Expend        Grad.Rate
##  Min.   : 3186   Min.   : 10.00
##  1st Qu.: 6751   1st Qu.: 53.00
##  Median : 8377   Median : 65.00
##  Mean   : 9660   Mean   : 65.46
##  3rd Qu.:10830   3rd Qu.: 78.00
##  Max.   :56233   Max.   :118.00
```

(ii) Use the pairs() function to produce a scatterplot matrix of the first ten columns or variables of the data. Recall that you can reference the first ten columns of a matrix A using A[,1:10].

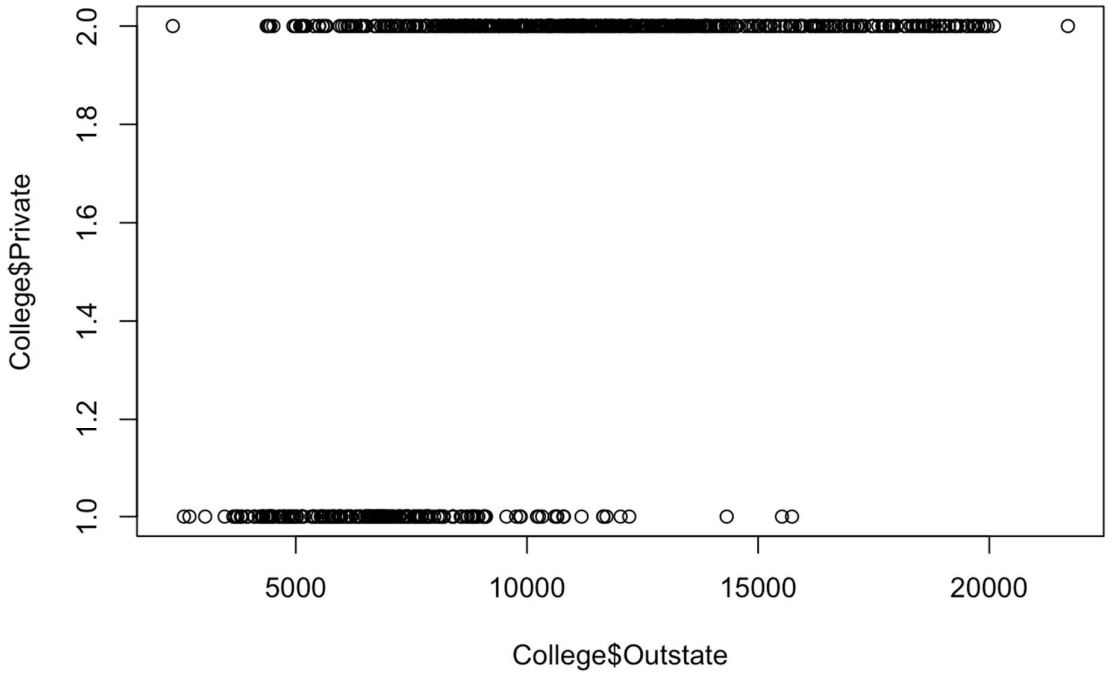
Correlation Pairs

```
library(ISLR)
pairs(College[,1:10]) # [,1:10] college dataframe didn't work
```

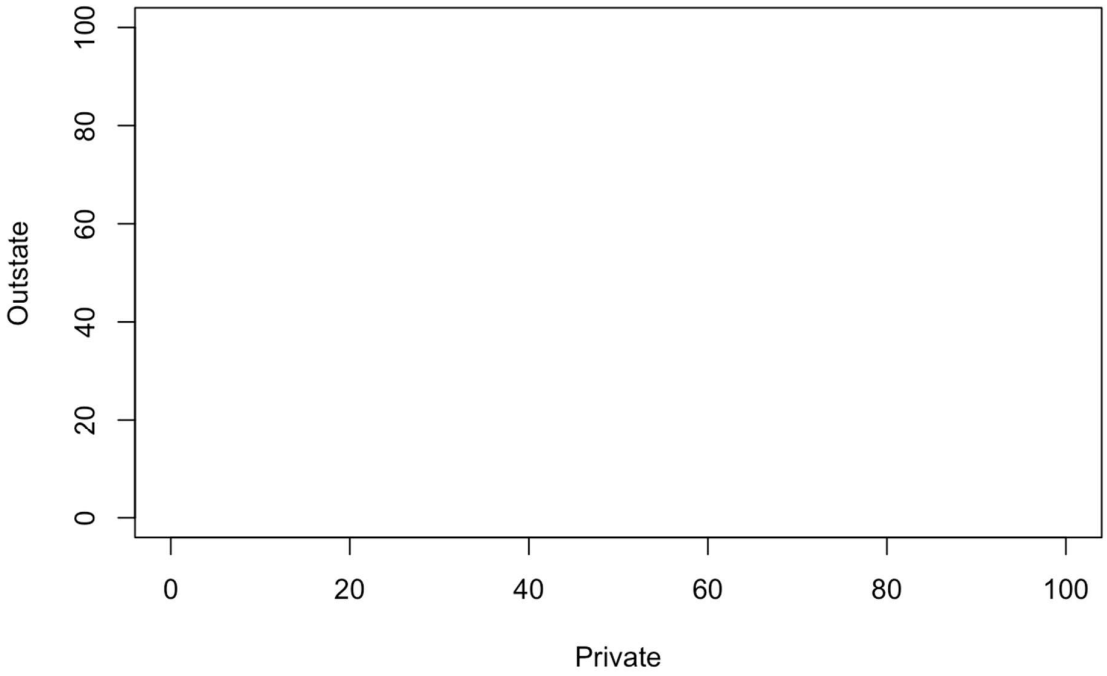


(iii) Use the plot() function to produce side-by-side boxplots of Outstate versus Private.

```
plot(College$Outstate, College$Private)
```



```
#rm(college)
detach("package:ISLR", unload=TRUE)

attach(college)
plot(Private, Outstate, xlim=c(0,100), ylim=c(0,100))
## Warning in xy.coords(x, y, xlabel, ylabel, log): NAs introduced by coercion
```

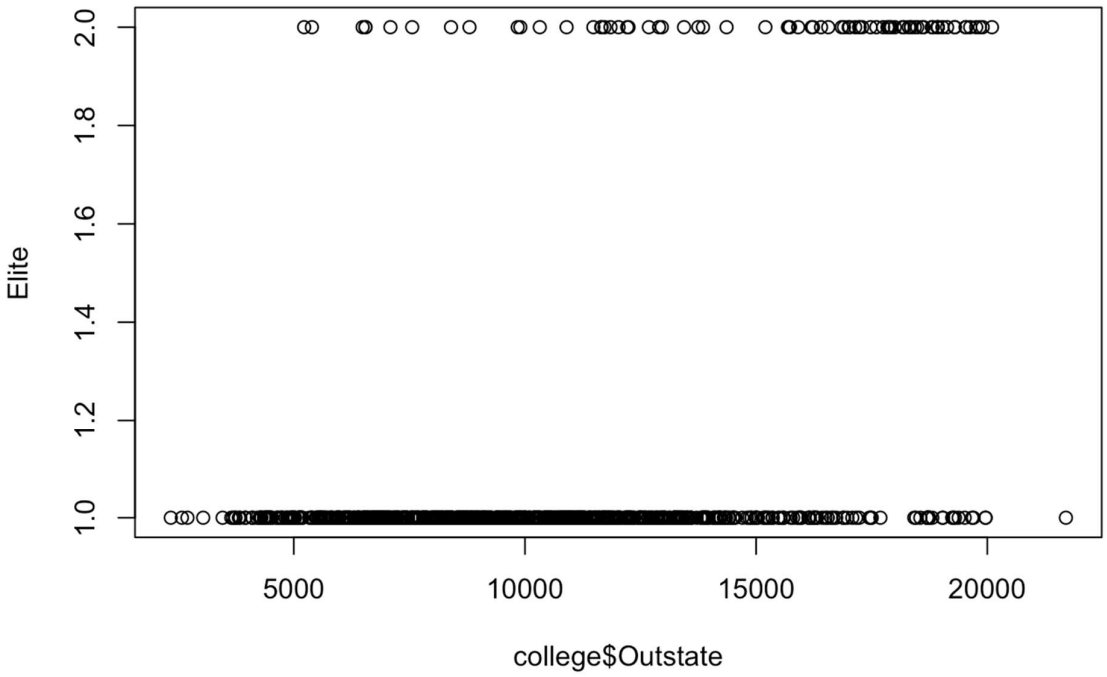


(iv) Create new qualitative variable, called Elite, by *binning* the Top10perc variable. We are going to divide universities into two groups based on whether or not the proportion of students coming from the top 10 % of their high school classes exceeds 50 %.

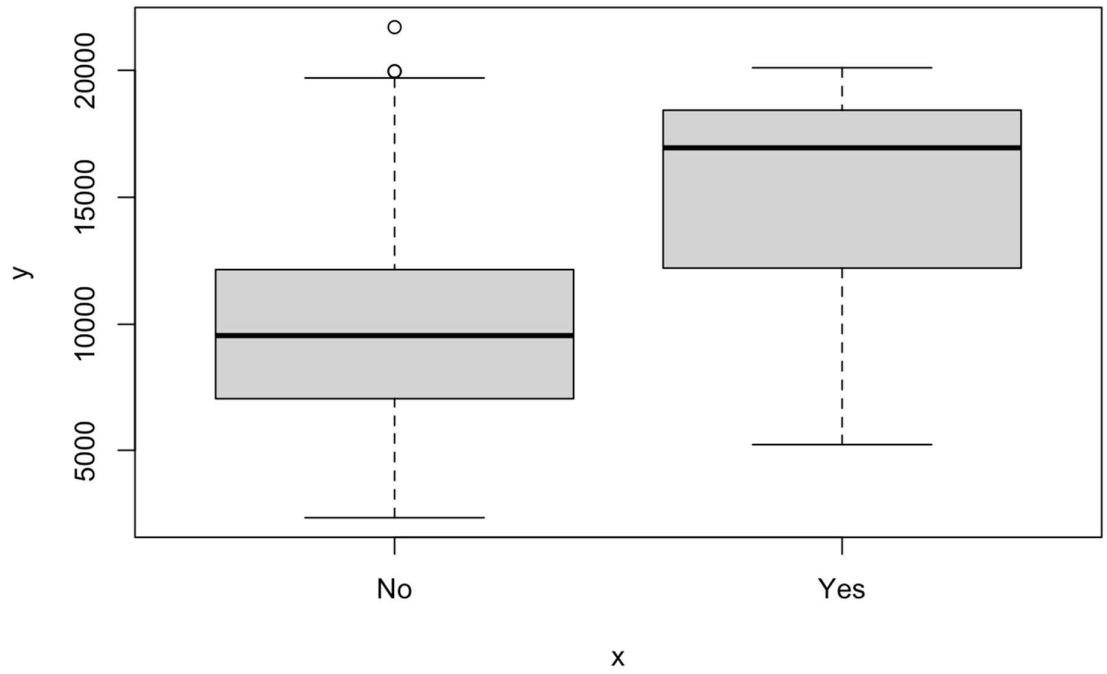
```
> Elite=rep("No",nrow(college))
> Elite[college$Top10perc>50]="Yes"
> Elite=as.factor(Elite)
> college=data.frame(college,Elite)
```

Use the summary() function to see how many elite universities there are. Now use the plot() function to produce side-by-side boxplots of Outstate versus Elite.

```
Elite=rep("No", nrow(college))
Elite[college$Top10perc > 50] = "Yes"
Elite = as.factor(Elite)
summary(Elite)

##  No Yes
## 699  78

plot(college$Outstate, Elite)
```




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

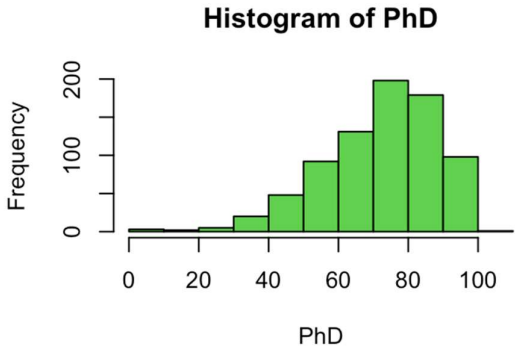


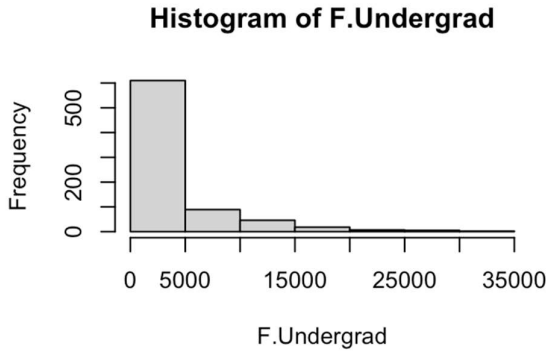
(v) Use the hist() function to produce some histograms with differing numbers of bins for a few of the quantitative variables. You may find the command par(mfrow=c(2,2)) useful: it will divide the print window into four regions so that four plots can be made simultaneously. Modifying the arguments to this function will divide the screen in other ways.

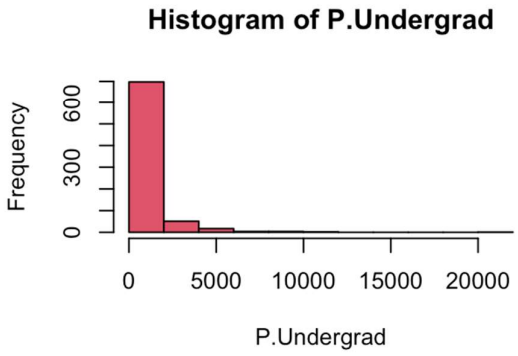
```
par(mfrow=c(2,2)) # 4 plots per picture
attach(college)

## The following objects are masked from college (pos = 3):
```

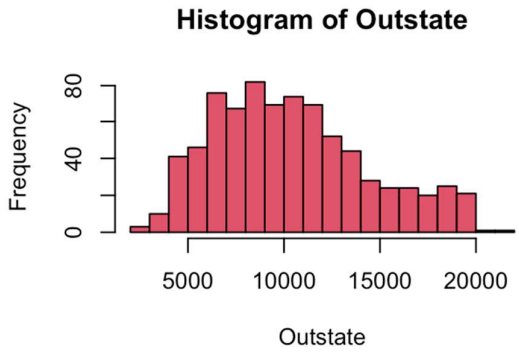
```
##
##    Accept, Apps, Books, Enroll, Expend, F.Undergrad, Grad.Rate,
##    Outstate, P.Undergrad, perc.alumni, Personal, PhD, Private,
##    Room.Board, S.F.Ratio, Terminal, Top10perc, Top25perc
```

```
hist(PhD, breaks = 10, col = 3)
hist(F.Undergrad, breaks = 10)
hist(P.Undergrad, col=2, breaks = 15)
hist(Outstate, col=2, breaks = 15)
```









Practical-3:

- This Practical involves the Auto data set. Make sure that the missing values have been removed from the data.

```
Auto = read.csv("../data/Auto.csv", header=T, na.strings="?")
Auto = na.omit(Auto)
dim(Auto)
summary(Auto)
```

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

```
# quantitative: mpg, cylinders, displacement, horsepower, weight,
# acceleration, year
# qualitative: name, origin
```

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

```
# apply the range function to the first seven columns of Auto
sapply(Auto[, 1:7], 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
```

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

```
sapply(Auto[, 1:7], mean)
#       mpg    cylinders displacement   horsepower      weight acceleration
#   23.445918    5.471939   194.411990   104.469388  2977.584184    15.541327
#      year
#   75.979592

sapply(Auto[, 1:7], sd)
#       mpg    cylinders displacement   horsepower      weight acceleration
#    7.805007    1.705783   104.644004    38.491160   849.402560    2.758864
#      year
#    3.683737
```

(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?

```
newAuto = Auto[-(10:85),]
dim(newAuto) == dim(Auto)- c(76,0)
newAuto[9,] == Auto[9,]
newAuto[10,] == Auto[86,]

sapply(newAuto[, 1:7], range)
#      mpg cylinders displacement horsepower weight acceleration year
# [1,] 11.0       3         68        46   1649        8.5   70
# [2,] 46.6       8        455       230   4997       24.8   82
sapply(newAuto[, 1:7], mean)
#       mpg    cylinders displacement   horsepower      weight acceleration
#   24.404430    5.373418   187.240506   100.721519  2935.971519    15.726899
```

```
#       year
#   77.145570
sapply(newAuto[, 1:7], sd)
#        mpg    cylinders displacement   horsepower       weight acceleration
#    7.867283     1.654179    99.678367    35.708853   811.300208     2.693721
#       year
#    3.106217
```

(e)

Using the full data set, investigate the predictors graphically, using scatterplots or other tools of your choice. Create some plots highlighting the relationships among the predictors. Comment on your findings.

```
pairs(Auto)
plot(Auto$mpg, Auto$weight)
# Heavier weight correlates with lower mpg.
plot(Auto$mpg, Auto$cylinders)
# More cylinders, less mpg.
plot(Auto$mpg, Auto$year)
# Cars become more efficient over time.
```

(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.

```
pairs(Auto)
# See descriptions of plots in (e).
# All of the predictors show some correlation with mpg. The name predictor has
# too little observations per name though, so using this as a predictor is
# likely to result in overfitting the data and will not generalize well.
```

Practical-4:

- This Practical involves the Boston housing data set.

(a)

To begin, load in the Boston data set. The Boston data set is part of the MASS library in R.

> library(MASS)

Now the data set is contained in the object Boston.

> Boston

Read about the data set:

> ?Boston

How many rows are in this data set? How many columns? What do the rows and columns represent?

```
library(MASS)
?Boston
dim(Boston)
# 506 rows, 14 columns
# 14 features, 506 housing values in Boston suburbs
```

(b)

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

```
pairs(Boston)
# X correlates with: a, b, c
# crim: age, dis, rad, tax, ptratio
# zn: indus, nox, age, lstat
# indus: age, dis
# nox: age, dis
# dis: lstat
# lstat: medv
```

(c)

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

```
plot(Boston$age, Boston$crim)
# Older homes, more crime
plot(Boston$dis, Boston$crim)
# Closer to work-area, more crime
plot(Boston$rad, Boston$crim)
# Higher index of accessibility to radial highways, more crime
plot(Boston$tax, Boston$crim)
# Higher tax rate, more crime
plot(Boston$ptratio, Boston$crim)
# Higher pupil:teacher ratio, more crime
```

(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.

```
par(mfrow=c(1,3))
hist(Boston$crim[Boston$crim>1], breaks=25)
# most cities have low crime rates, but there is a long tail: 18 suburbs appear
# to have a crime rate > 20, reaching to above 80
hist(Boston$tax, breaks=25)
# there is a large divide between suburbs with low tax rates and a peak at 660-680
hist(Boston$ptratio, breaks=25)
```

```
# a skew towards high ratios, but no particularly high ratios
```

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

```
dim(subset(Boston, chas == 1))
# 35 suburbs
```

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

```
median(Boston$ptratio)
# 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.

```
> t(subset(Boston, medv == min(Boston$medv)))
#             399      406
# crim     38.3518  67.9208 above 3rd quartile
# zn        0.0000   0.0000 at min
# indus    18.1000  18.1000 at 3rd quartile
# chas      0.0000   0.0000 not bounded by river
# nox       0.6930   0.6930 above 3rd quartile
# rm        5.4530   5.6830 below 1st quartile
# age     100.0000 100.0000 at max
# dis       1.4896   1.4254 below 1st quartile
# rad      24.0000  24.0000 at max
# tax     666.0000 666.0000 at 3rd quartile
# ptratio  20.2000  20.2000 at 3rd quartile
# black   396.9000 384.9700 at max; above 1st quartile
# lstat    30.5900  22.9800 above 3rd quartile
# medv      5.0000   5.0000 at min
summary(Boston)
# Not the best place to live, but certainly not the worst.
```

(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.

```
dim(subset(Boston, rm > 7))
# 64
dim(subset(Boston, rm > 8))
# 13
summary(subset(Boston, rm > 8))
summary(Boston)
# relatively lower crime (comparing range), lower lstat (comparing range)
```