Block 3

Classification (Chapter 4).

Exercise 14, p. 194 (a), (b), (c), (f) only: predicting gas mileage based on the Auto data set.

```
library(ISLR)
library(MASS)
library(class)
Auto1 = read.csv("/Volumes/work/MTH522/data/auto1.csv")
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
```

```
# Here we are converting origin to classification colums and giving 3 names as Origin
1, Origin 2 and Origin 3.

Auto$origin[Auto$origin == 1] = "Origin 1"
Auto$origin[Auto$origin == 2] = "Origin 2"
Auto$origin[Auto$origin == 3] = "Origin 3"
Auto$origin = as.factor(Auto$origin)
head(Auto)
```

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

14(a). Create a binary variable, mpg01, that contains a 1 if mpg contains a value above its median, and a 0 if mpg contains a value below its median. You can compute the median using the median() function. Note you may find it helpful to use the data.frame() function to create a single data set containing both mpg01 and the other Auto variables.

```
mpg01 = rep(0, dim(Auto)[1])
mpg01[Auto$mpg > median(Auto$mpg)] = 1
Auto = data.frame(Auto, mpg01)
head(Auto)
```

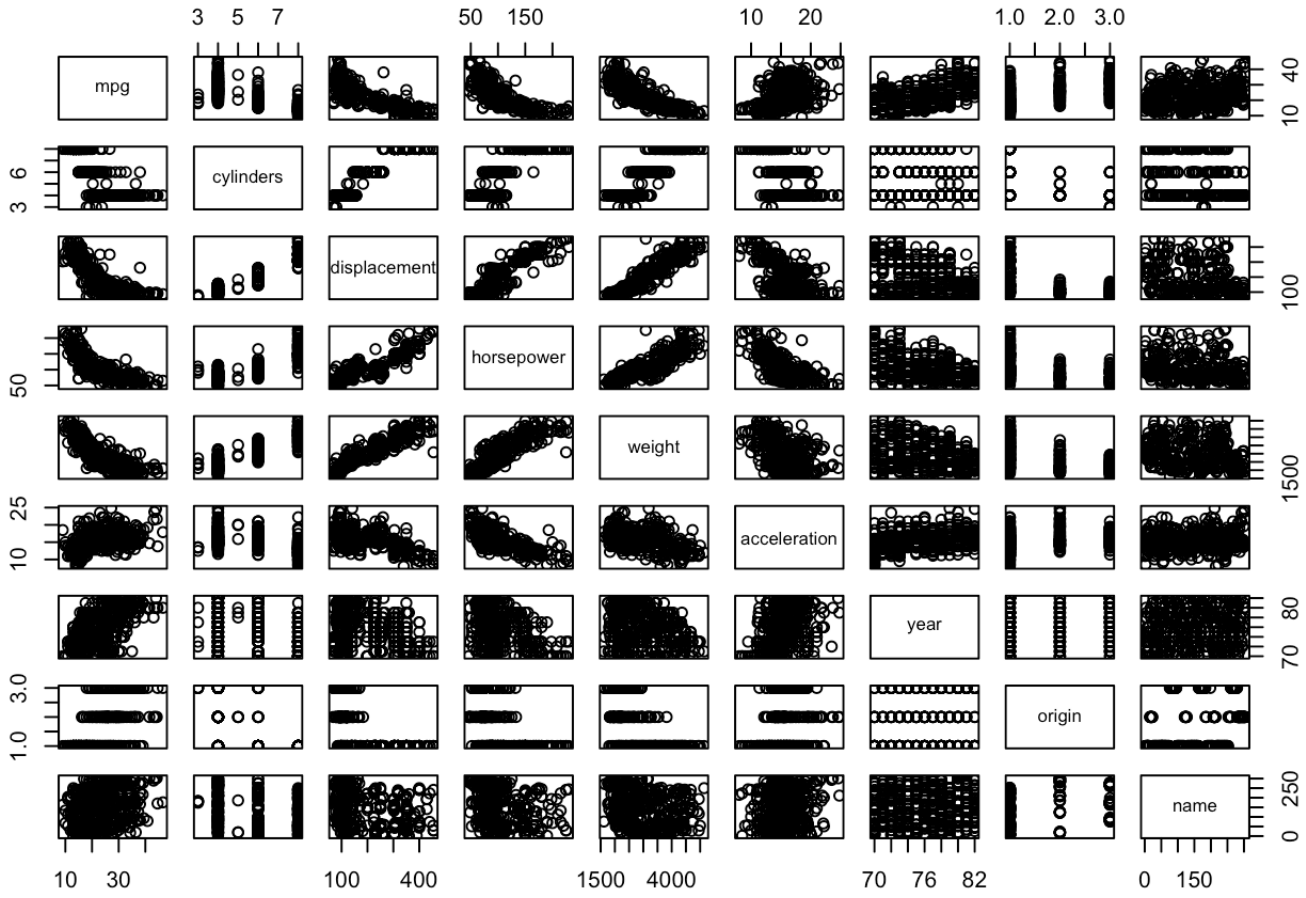
```
##   mpg cylinders displacement horsepower weight acceleration year   origin
## 1  18         8          307        130   3504         12.0   70 Origin 1
## 2  15         8          350        165   3693         11.5   70 Origin 1
## 3  18         8          318        150   3436         11.0   70 Origin 1
## 4  16         8          304        150   3433         12.0   70 Origin 1
## 5  17         8          302        140   3449         10.5   70 Origin 1
## 6  15         8          429        198   4341         10.0   70 Origin 1
##                       name mpg01
## 1 chevrolet chevelle malibu     0
## 2         buick skylark 320     0
## 3        plymouth satellite     0
## 4             amc rebel sst     0
## 5               ford torino     0
## 6          ford galaxie 500     0
```

```
colnames(Auto1)
```

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

14(b).Explore the data graphically in order to investigate the association between mpg01 and the other features. Which of the other features seem most likely to be useful in predicting mpg01? Scatterplots and boxplots may be useful tools to answer this question. Describe your findings.

```
pairs(Auto[,1:9])
```



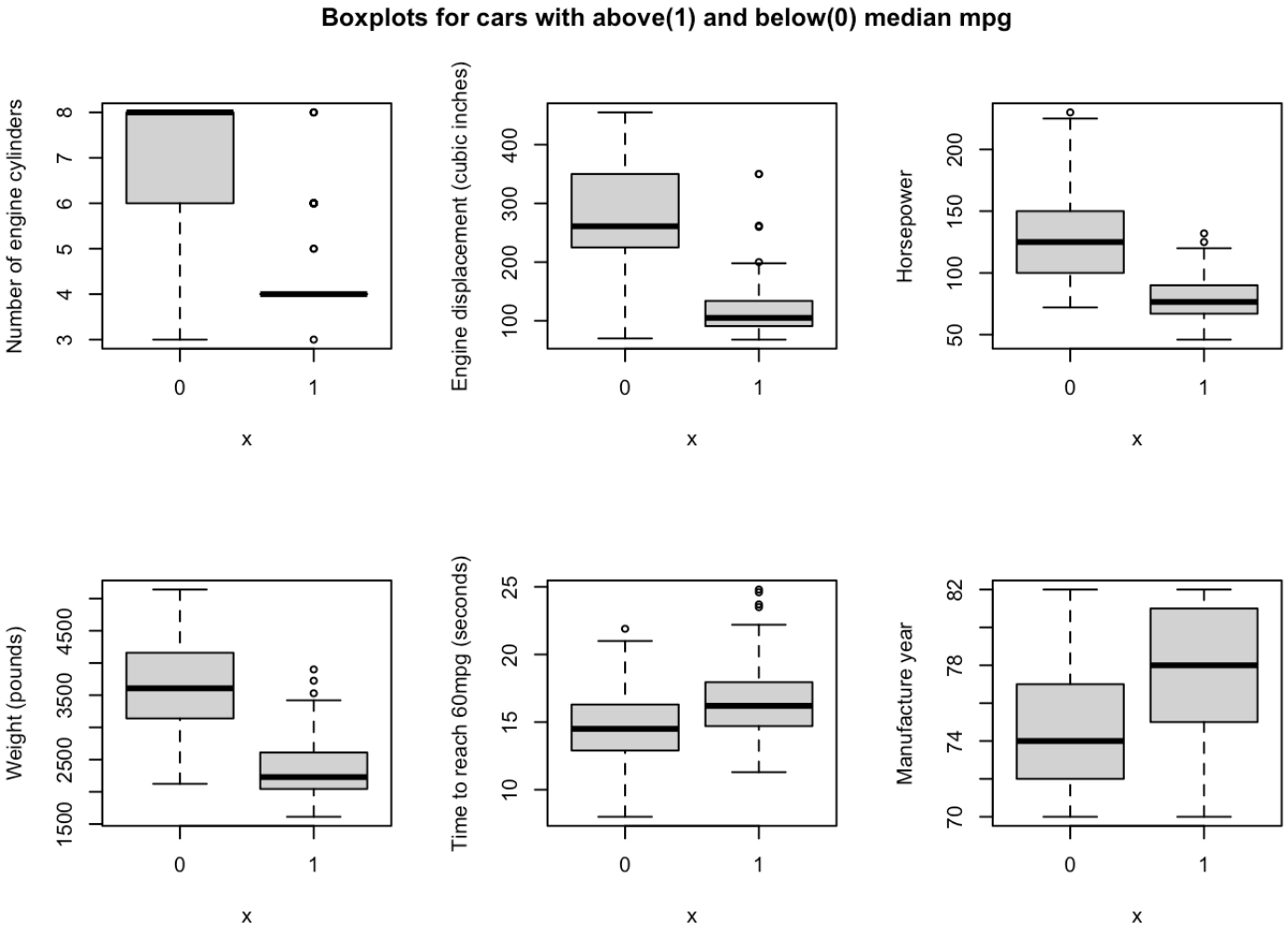
```
autocorr = cor(subset(Auto1,select = -c(name)))
autocorr
```

```
##                     mpg  cylinders displacement horsepower     weight
## mpg           1.0000000 -0.7762599   -0.8044430         NA -0.8317389
## cylinders    -0.7762599  1.0000000    0.9509199         NA  0.8970169
## displacement -0.8044430  0.9509199    1.0000000         NA  0.9331044
## horsepower           NA         NA           NA          1         NA
## weight       -0.8317389  0.8970169    0.9331044         NA  1.0000000
## acceleration  0.4222974 -0.5040606   -0.5441618         NA -0.4195023
## year          0.5814695 -0.3467172   -0.3698041         NA -0.3079004
## origin        0.5636979 -0.5649716   -0.6106643         NA -0.5812652
##              acceleration       year     origin
## mpg             0.4222974  0.5814695  0.5636979
## cylinders      -0.5040606 -0.3467172 -0.5649716
## displacement   -0.5441618 -0.3698041 -0.6106643
## horsepower             NA         NA         NA
## weight         -0.4195023 -0.3079004 -0.5812652
## acceleration    1.0000000  0.2829009  0.2100836
## year            0.2829009  1.0000000  0.1843141
## origin          0.2100836  0.1843141  1.0000000
```

Observations:

1. According to the data above we can conclude that “mpg” and “mpg01” have a strong positive correlation between them. We Also have a few strong negative correlations like “cylinders”, “displacement”, and “weight”.

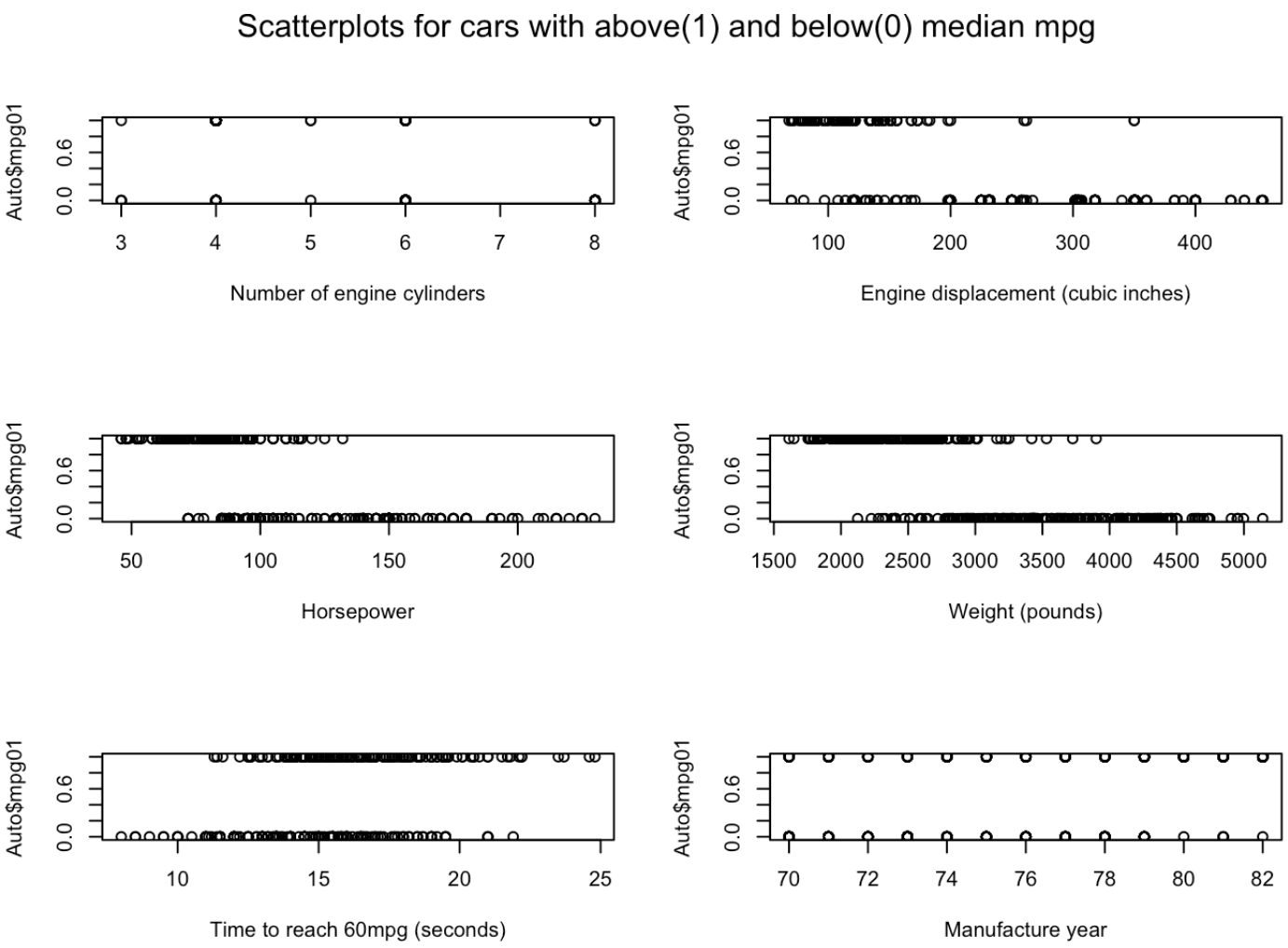
```
par(mfrow = c(2, 3))
plot(factor(Auto$mpg01), Auto$cylinders, ylab = "Number of engine cylinders")
plot(factor(Auto$mpg01), Auto$displacement, ylab = "Engine displacement (cubic inches
)")
plot(factor(Auto$mpg01), Auto$horsepower, ylab = "Horsepower")
plot(factor(Auto$mpg01), Auto$weight, ylab = "Weight (pounds)")
plot(factor(Auto$mpg01), Auto$acceleration, ylab = "Time to reach 60mpg (seconds)")
plot(factor(Auto$mpg01), Auto$year, ylab = "Manufacture year")
title("Boxplots for cars with above(1) and below(0) median mpg", outer = TRUE, line =
-1)
```



Observations:

1. From the above box plot we can see that, majority of cars with above-median mpg have four-cylinder engines (upper-left figure).
2. At least 75% of the cars with above-median mpg have smaller engines than 75% of the cars with below-median mpg (upper-middle figure). This is also true for horsepower (upper-right pair of boxplots) and weight (lower-left pair of box plots).
3. There is a lot more overlap in both time to reach 60mpg and manufacture year between the two categories of cars, whereas for the other predictors there is almost no overlap for the boxplots between the two categories. This suggests that `cylinders`, `displacement`, `horsepower`, and `weight` will be the most useful in predicting `mpg01`.

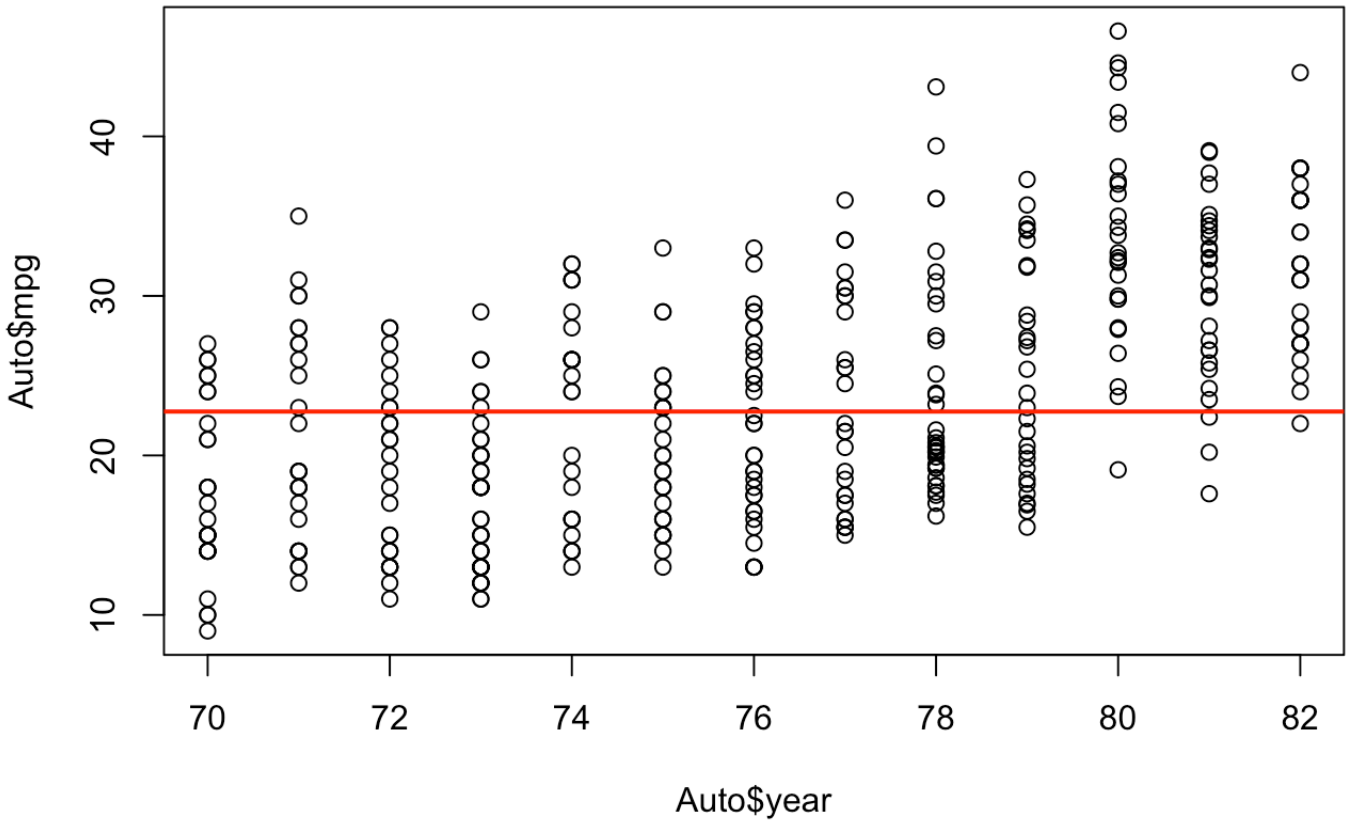
```
par(mfrow = c(3, 2))
plot(Auto$cylinders, Auto$mpg01, xlab = "Number of engine cylinders")
plot(Auto$displacement, Auto$mpg01, xlab = "Engine displacement (cubic inches)")
plot(Auto$horsepower, Auto$mpg01, xlab = "Horsepower")
plot(Auto$weight, Auto$mpg01, xlab = "Weight (pounds)")
plot(Auto$acceleration, Auto$mpg01, xlab = "Time to reach 60mpg (seconds)")
plot(Auto$year, Auto$mpg01, xlab = "Manufacture year")
mtext("Scatterplots for cars with above(1) and below(0) median mpg", outer = TRUE, li
ne = -1.1)
```



Observations:

1. Looking at scatterplots with `mpg01` on the y-axis and the various quantitative variables on the x-axes provides further evidence to suggest that `horsepower` and `weight` will be useful in predicting `mpg01`
2. For engine displacement, the overlap mainly comes from a decent number of cars with below-median mpg and engine displacements in the bottom 25%.

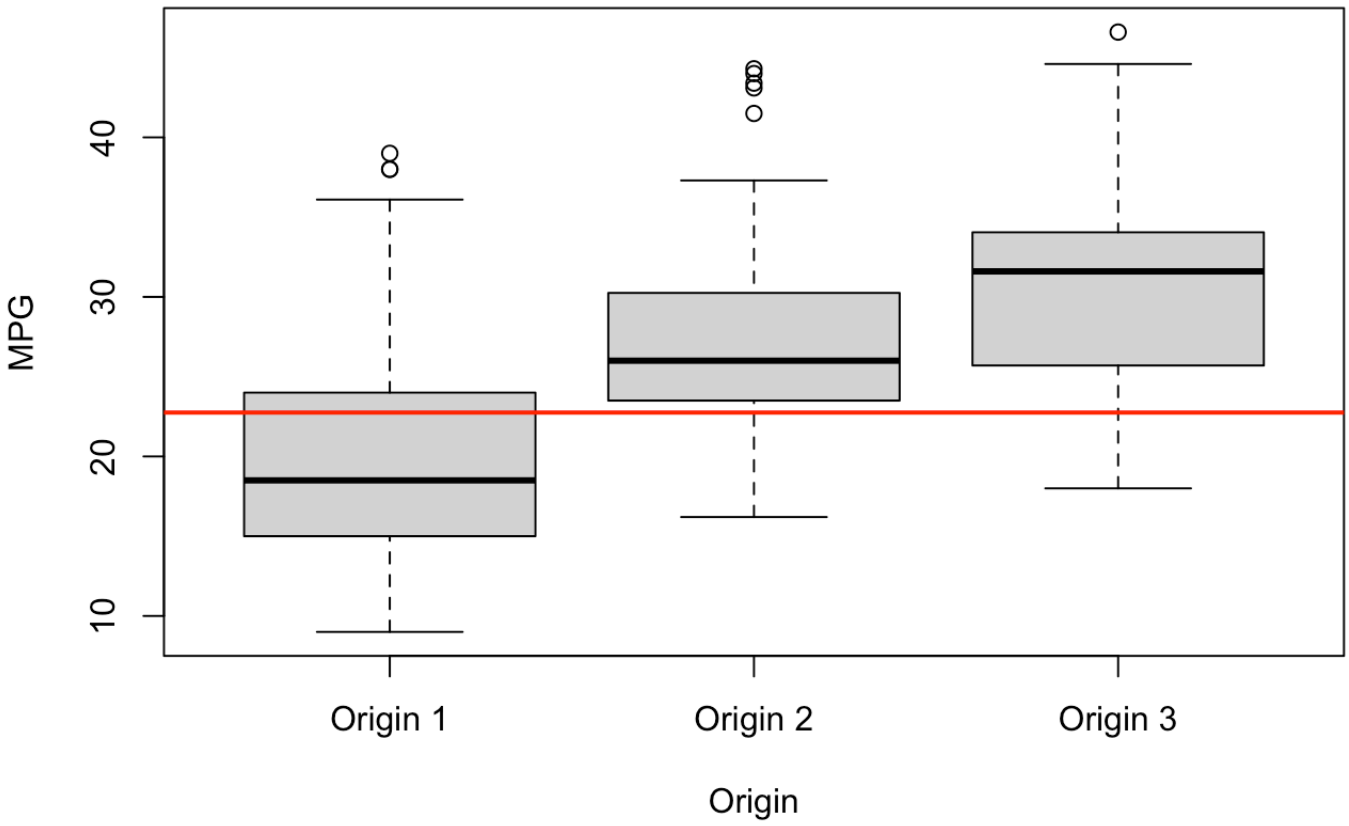
```
plot(Auto$year, Auto$mpg)
abline(h = median(Auto$mpg), lwd = 2, col = "red")
```



Observations:

1. The above scatterplot of `mpg` vs `year` also shows that the newer cars in the data set tend to be more fuel efficient.

```
plot(Auto$origin, Auto$mpg, xlab = "Origin", ylab = "MPG")
abline(h = median(Auto$mpg), lwd = 2, col = "red")
```



Observations:

1. We see that there is a clear difference between Origin 1 cars, which tend to have below-median fuel efficiency, and Origin 2 and Origin3 cars, which tend to have above-median fuel efficiency. Thus, it seems that `origin` will also be useful in predicting `mpg01` .

14(c).Split the data into a training set and a test set.

```
set.seed(1)
train = sample(dim(Auto)[1], size = 0.75*dim(Auto)[1])
```

14(f).Perform logistic regression on the training data in order to predict mpg01 using the variables that seemed most associated with mpg01 in (b). What is the test error of the model obtained.

```
lda.fit = lda(mpg01 ~ cylinders + displacement + horsepower + weight + year + origin,
data = Auto, subset = train)
```

```
lda.pred = predict(lda.fit, Auto[-train, ])
table(lda.pred$class, Auto[-train, "mpg01"], dnn = c("Predicted", "Actual"))
```

```
##           Actual
## Predicted  0  1
##         0 41  0
##         1 12 45
```

```
1 - mean(lda.pred$class == Auto[-train, "mpg01"])
```

```
## [1] 0.122449
```

Observations:

1. When using linear discriminant analysis to predict `mpg01` using `cylinders`, `displacement`, `horsepower`, `weight`, `year`, and `origin`, we had an overall test error of 12.24%.