Q9

Rahul Atre

2023-10-03

Q9 [Data Visualization and Data Exploration]

This exercise involves the Auto.csv dataset. Make sure that the missing values have been removed from the data.

```
library(readr)
autoData = na.omit(read.csv("Auto.csv"))
```

1. Which of the predictors are quantitative, and which are qualitative?

Lets look at the initial values of each column

```
str(autoData)
```

```
## 'data.frame':    392 obs. of  9 variables:
##  $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
##  $ cylinders   : int  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : int  130 165 150 150 140 198 220 215 225 190 ...
##  $ weight      : int  3504 3693 3436 3433 3449 4341 4354 4312 4425 3850 ...
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ year        : int  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ name        : chr  "chevrolet chevelle malibu" "buick skylark 320" "plymouth satellite" "amc rebel
##  - attr(*, "na.action")= 'omit' Named int [1:5] 33 127 331 337 355
##   ..- attr(*, "names")= chr [1:5] "33" "127" "331" "337" ...
```

Based on the above data, the predictors that are quantitative are: mpg, cylinders, displacement, horsepower, weight, acceleration, year

The predictors that are qualitative are: origin, name

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

Since the last 3 columns are qualitative, we can ignore those and apply the following function:

```
sapply(autoData[, 1:7], range) #Apply the range function to all quant. columns
```

```
##       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
```

3. What is the mean and standard deviation of each quantitative predictor?

We can use the same function as above, replacing range with mean:

```
sapply(autoData[, 1:7], mean) #Apply the mean function to all quant. columns
```

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

```
sapply(autoData[, 1:7], sd) #Apply the standard deviation function to all quant. columns
```

```
##          mpg    cylinders displacement   horsepower       weight acceleration
##     7.805007     1.705783   104.644004    38.491160   849.402560     2.758864
##         year
##     3.683737
```

4. 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?

```
sapply(autoData[-(10:85), 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(autoData[-(10:85), 1:7], mean)
```

```
##          mpg    cylinders displacement   horsepower       weight acceleration
##    24.404430     5.373418   187.240506   100.721519  2935.971519    15.726899
##         year
##    77.145570
```

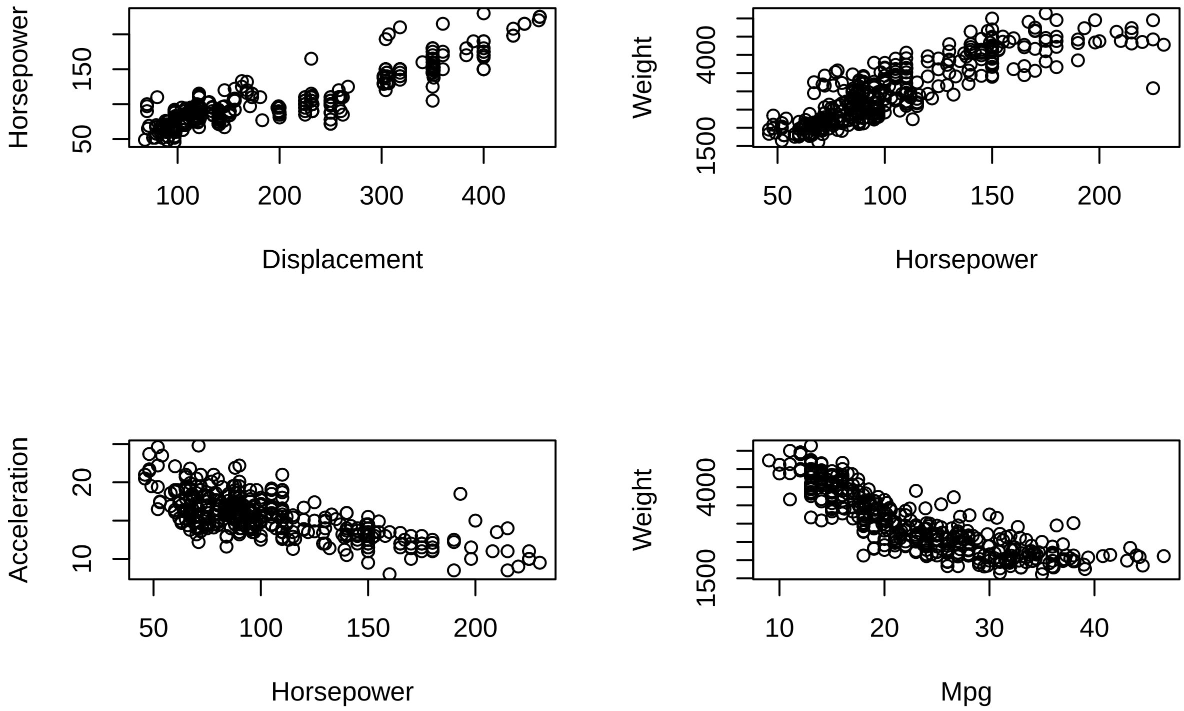
```
sapply(autoData[-(10:85), 1:7], sd)
```

```
##          mpg    cylinders displacement   horsepower       weight acceleration
##     7.867283     1.654179    99.678367    35.708853   811.300208     2.693721
##         year
##     3.106217
```

5. Using the full dataset, 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.

```
par(mfrow=c(2,2)) #Create a plot matrix

plot(autoData$displacement, autoData$horsepower, xlab = "Displacement", ylab = "Horsepower")
plot(autoData$horsepower, autoData$weight, xlab = "Horsepower", ylab = "Weight")
plot(autoData$horsepower, autoData$acceleration, xlab = "Horsepower", ylab = "Acceleration")
plot(autoData$mpg, autoData$weight, xlab = "Mpg", ylab = "Weight")
```



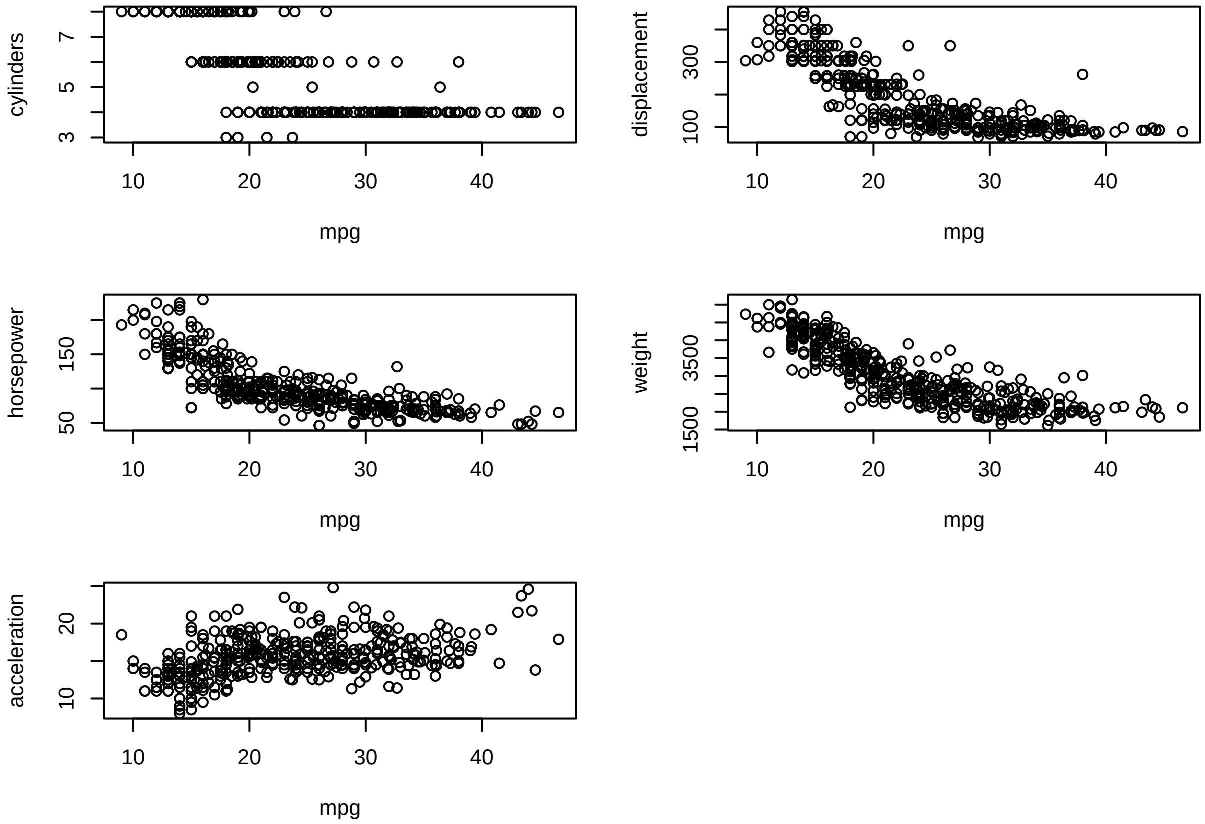
Based on the above four scatterplots, we can make four conclusions regarding the correlation between the auto data's variables:

- As displacement **increases**, horsepower **increases**
- As horsepower **increases**, weight **increases**
- As horsepower **increases**, the acceleration **decreases**
- As mpg **increases**, the weight **decreases**

All relationships can be estimated using the least squares approach for a linear line of best fit.

6. 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.

```r
par(mfrow=c(3,2), mar = c(5, 4, 1, 2)) #Create matrix of plots w/ set graphical parameters
plot(autoData$cylinders ~ autoData$mpg, xlab = 'mpg', ylab = 'cylinders')
plot(autoData$displacement ~ autoData$mpg, xlab = 'mpg', ylab = 'displacement')
plot(autoData$horsepower ~ autoData$mpg, xlab = 'mpg', ylab = 'horsepower')
plot(autoData$weight ~ autoData$mpg, xlab = 'mpg', ylab = 'weight')
plot(autoData$acceleration ~ autoData$mpg, xlab = 'mpg', ylab = 'acceleration')
```



The above scatterplots do suggest that the other variables would be useful in predicting mpg. We can say that, for any positive correlation, the other variable would require a relatively higher value, whereas for a negative correlation, the other variable would need to have a lower value. So, mpg increases when the number of cylinders are low, displacement is low, horsepower is low, weight is low, and acceleration is high.