GDAA 1000 Assignment 1

Juan C. Reyes - W0465200@campus.nscc.ca

GDAA 1000 Fundamentals of Geospatial Data Analytics

Let's begin by importing the dataset 'mtcars'.

Reference: 1974 US Magazine Motor Trend Car Road Tests

```
data("mtcars")
#Assigning the data to a variable
carsdf <- mtcars
```

Let's investigate the class of the data set:

```
class(carsdf)
```

```
## [1] "data.frame"
```

As we can see, the "mtcars" data set is stored as a data frame object in memory.

Let's take a quick first glance of the data set:

```
head(carsdf)
```

```
##                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

From this first glance of the data frame we can see that there are 32 unique rows (cars) and 11 distinct columns (variables/attributes).

Let's take a quick look at each of our unique variables:

```
names(carsdf)
```

```
##  [1] "mpg"  "cyl"  "disp" "hp"   "drat" "wt"   "qsec" "vs"   "am"   "gear"
## [11] "carb"
```

We have 11 unique variables which each describe individual properties to each vehicle (row).

It is important to get an understanding of the structure of this data set. Let's take a close look at the data type each variable represents:

```
str(carsdf)
```

```
## 'data.frame':    32 obs. of  11 variables:
##  $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
##  $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
##  $ disp: num  160 160 108 258 360 ...
##  $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
##  $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
##  $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
##  $ qsec: num  16.5 17 18.6 19.4 17 ...
##  $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
##  $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
##  $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
##  $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

As we can see, every variable is a numeric data type!

Let's see what the average horsepower of the vehicles tested is:

```
mean(carsdf$hp)
```

```
## [1] 146.6875
```

Furthermore, the mean displacement in the first five vehicles of the data set is given by:

```
mean(carsdf[1:5,]$disp)
```

```
## [1] 209.2
```

Now we investigate filtering our data such that we only display the vehicles that have more than 4 cylinders and less than 21 mpg.

We can use the filter function in `dplyr` to determine this:

```
dplyr::filter(carsdf, cyl > 4 & mpg < 21)
```

```
##                    mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## Hornet Sportabout 18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
## Duster 360        14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
## Merc 280          19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
## Merc 280C         17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
## Merc 450SE        16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
## Merc 450SL        17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
## Merc 450SLC       15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
## Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
## Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
## Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
## Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
## AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
## Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
## Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
## Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
## Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
## Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
```

Similarly, we could use `dplyr` to apply its pipe operator `%>%`:

```
library(dplyr)
carsdf %>% filter(cyl > 4, mpg < 21 )
```

```
##                    mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## Hornet Sportabout 18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
## Duster 360        14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
## Merc 280          19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
## Merc 280C         17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
## Merc 450SE        16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
## Merc 450SL        17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
## Merc 450SLC       15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
## Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
## Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
## Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
## Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
## AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
## Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
## Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
## Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
## Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
## Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
```

Note the use of the pipe operator `%>%` which forwards a value (or in our case, a data frame) into the following function. It also serves as a way of'decreasing development time and improve readability and maintainability of code.' (click for source)

Now we can determine the number of levels in the cylinders variable by using `as.factor()`:

```
as.factor(carsdf$cyl)
```

```
##  [1] 6 6 4 6 8 6 8 4 4 6 6 8 8 8 8 8 8 4 4 4 4 8 8 8 8 4 4 4 8 6 8 4
## Levels: 4 6 8
```

The unique factors for the cylinders variable are 4,6, and 8.

Lastly, we create a scatterplot matrix to display the relationship between the miles per gallon (mpg), cylinders(cyl), displacement (disp), and horsepower (hp) attribute data for just the first 20 cars of the dataset. (Click here for the pairs() function documentation.

It is helpful to recall that the first twenty elements of this dataset should be unique, and thus should be classified as factors. We double check:

```
first_twenty <- head(carsdf,20)

num_unique_cars <- nrow(unique(first_twenty))

each_car <- factor(rownames(first_twenty))

print(num_unique_cars)
```

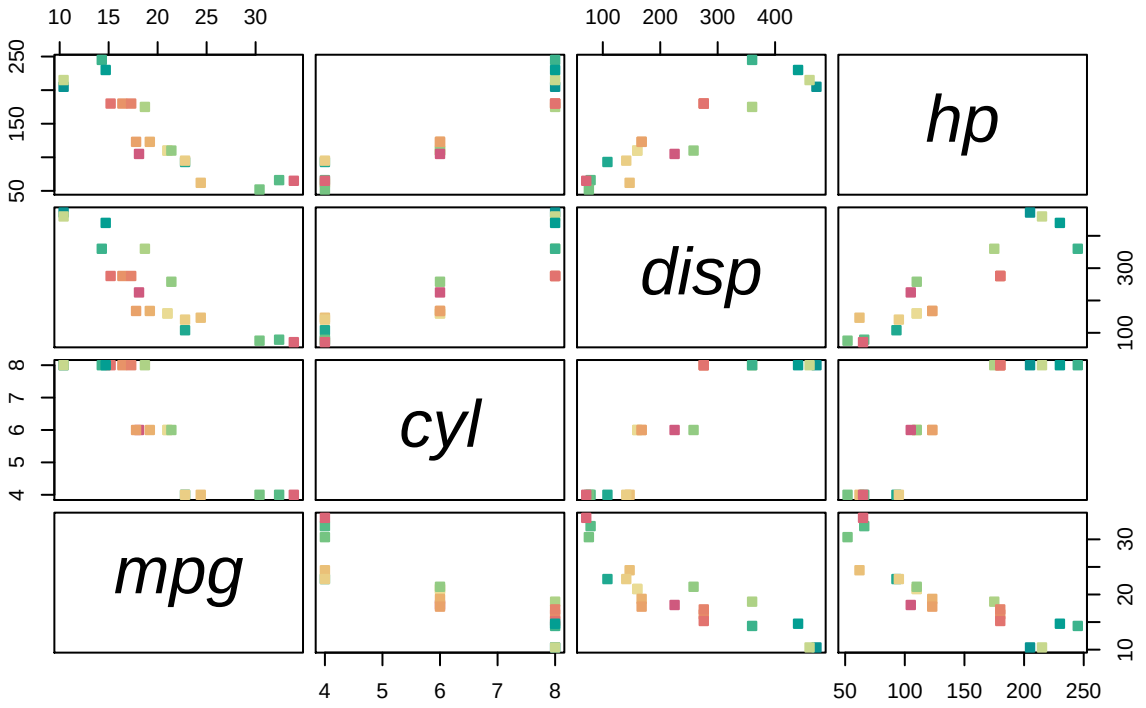
```
## [1] 20
```

As we can see the number of unique cars is 20 and the class structure is set to factor. Let's look at the cars we are about to compare!

```
each_car
```

```
##  [1] Mazda RX4           Mazda RX4 Wag       Datsun 710
##  [4] Hornet 4 Drive      Hornet Sportabout   Valiant
##  [7] Duster 360          Merc 240D           Merc 230
## [10] Merc 280            Merc 280C           Merc 450SE
## [13] Merc 450SL          Merc 450SLC         Cadillac Fleetwood
## [16] Lincoln Continental Chrysler Imperial   Fiat 128
## [19] Honda Civic         Toyota Corolla
## 20 Levels: Cadillac Fleetwood Chrysler Imperial Datsun 710 ... Valiant
```

We have 20 unique levels (as expected). We can use these to display each vehicle as a unique colour on the graph like in the example shown *here!*



This is an interesting graph which immediately displays various interesting aspects of our data set. There appears to be several linear correlations that we can infer from regression tools. For instance, there appears to be a negative correlation between vehicle horsepower and the number of cylinders to that of the miles per gallon obtainable by the vehicle. It seems reasonable to believe that the more horsepower or cylinders a vehicle possesses, the less miles per gallon it is able to attain. This would be an interesting analysis for further study of this data set.