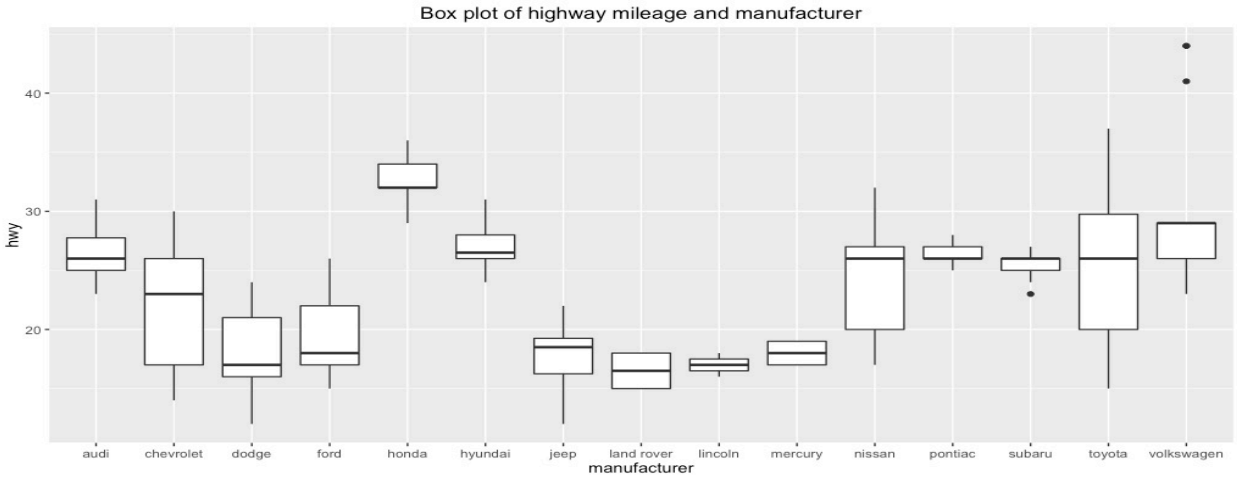
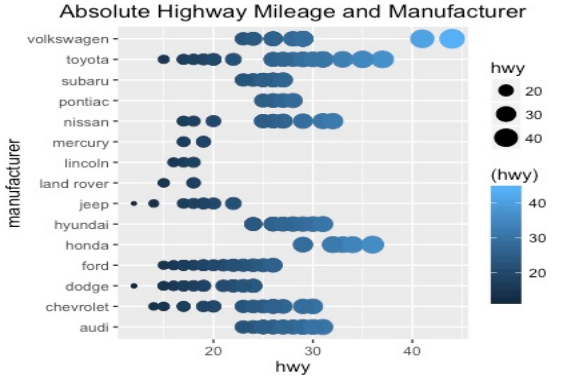
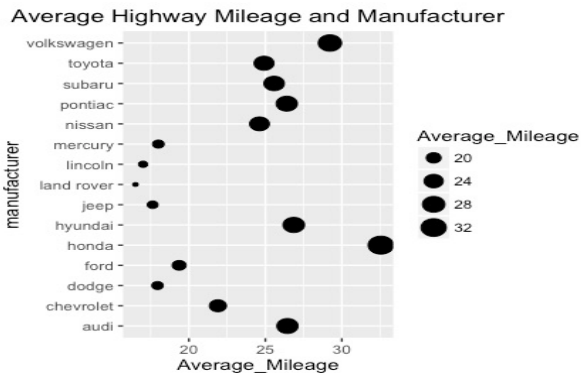
Assignment 2

Q1)

```
library(ggplot2)
library(reshape)
library(plyr)
Plot<-function(n)
{
  g_1<-ggplot(mpg, aes(manufacturer, hwy)) + geom_boxplot() + ggtitle("Box plot of highway mileage and manufacturer")
  print(g_1)

  g_1<-ggplot(mpg, aes(hwy, manufacturer)) + geom_point(aes(color=(hwy), size=hwy)) + ggtitle("Absolute Highway Mileage and Manufacturer")
  print(g_1)

  df<-(ddply(mpg, .(manufacturer), summarize,  Average_Mileage=mean(hwy)))
  g_2<-ggplot(df, aes(Average_Mileage,manufacturer)) + geom_point(aes(size = Average_Mileage)) +  ggtitle("Average Highway Mileage and Manufacturer")
  print(g_2)
}
```



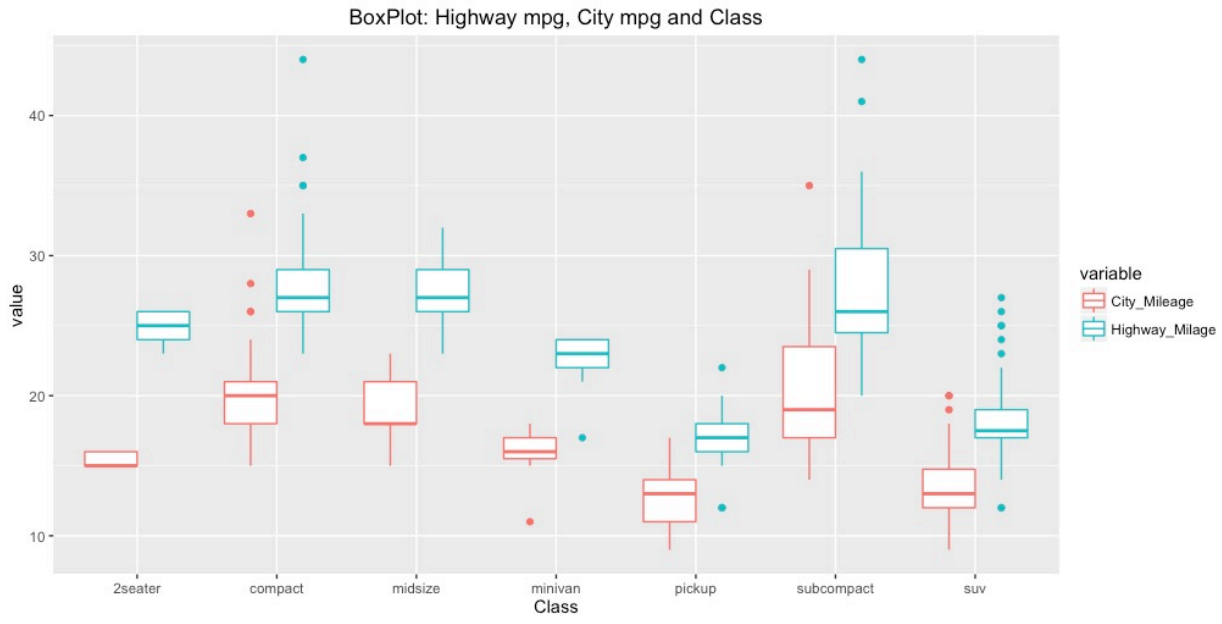




Based on the above plots, it is clear that Honda offers the highest average mileage whereas Land Rover offers the lowest average mileage on highway. Interestingly enough, if we look at Volkswagen's mileage, a couple of their cars perform extremely well and they could be considered outliers.

Q2)

```
library(ggplot2)
library(reshape)
library(plyr)
Plot<-function(n)
{
 df <- data.frame(Class = mpg$class,
                  City_Mileage = mpg$cty,
                  Highway_Milage = mpg$hwy)

 df <- melt(df ,  id.vars = 'Class')

 g_1<-ggplot(df, aes(Class,value)) + geom_boxplot(aes(colour=variable)) +  ggtitle("BoxPlot:
Highway mpg, City mpg and Class")
 print(g_1)
}
```



From the box plot above, it is apparent that the highway mileage is always better than the city mileage for any model class. It can also be observed that sub-compact and compact cars seem to deliver the highest mileage in both city and highway. On the other hand, SUVs and pickups offer the worst performance in terms of mileage.

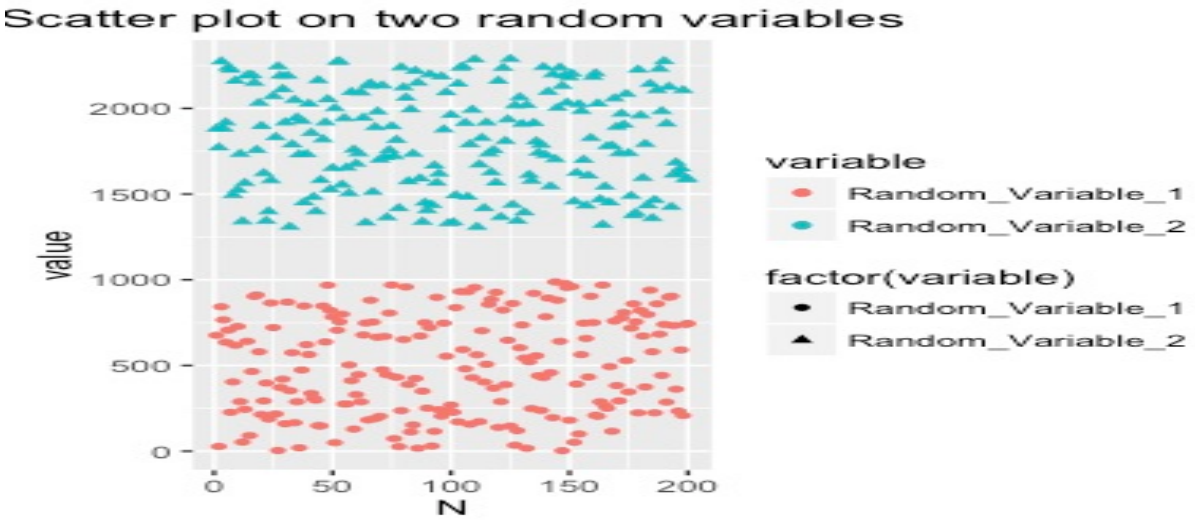
Q3)
A histogram is highly useful when wide variances or very little variance that exist among the observed frequencies for a particular data set.

A box plot is useful when there is moderate variation among the observed frequencies, which causes the histogram to look ragged and non-symmetrical due to the way the data is grouped. This may lead one to assume the data is slightly skewed. However, when a box plot is used to graph the same data points, the chart indicates a perfect normal distribution.

Reference - http://www.brighthubpm.com/six-sigma/58254-box-plots-vs-histograms-in-project-management/

Q4)

```
library(ggplot2)
library(reshape)
library(plyr)
Plot<-function(n)
{
  n = 200 #Tried changing this number to 2000 to check the size of files
  df <- data.frame(N = 1:n,
                   Random_Variable_1 = runif(n, 1, 1000),
                   Random_Variable_2 = runif(n, 1300, 2300))

  df <- melt(df ,  id.vars = 'N')

  g_1<-ggplot(df, aes(N,value)) + geom_point(aes(colour=variable, shape = factor(variable))) +
ggtitle("Scatter plot using runif() on two random variables")
  print(g_1)
}
```



N= 200

Size in ps format: 25 KB
Size in pdf format: 25 KB
Size in jpeg format: 49 KB
Size in png format: 61 KB

N= 2000
Size in ps format: 156 KB
Size in pdf format: 156 KB
Size in jpeg format: 47 KB
Size in png format: 74 KB

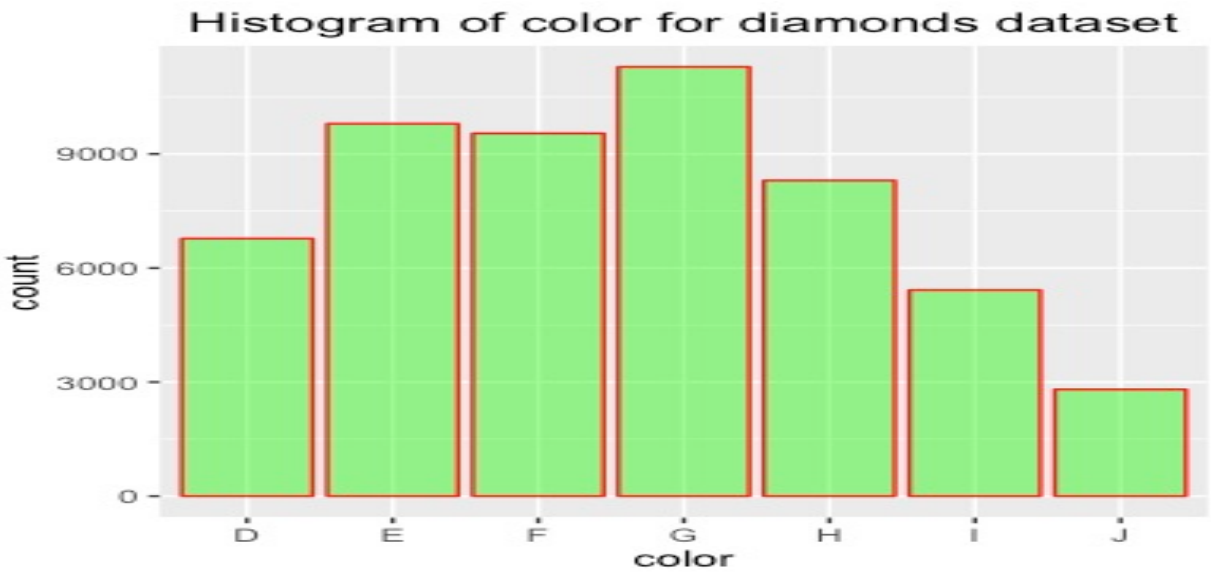
Clearly, as value of N was increased 10 times, the size of eps and pdf formats increased six times whereas png and jpeg formats size increased very little.

Q5)

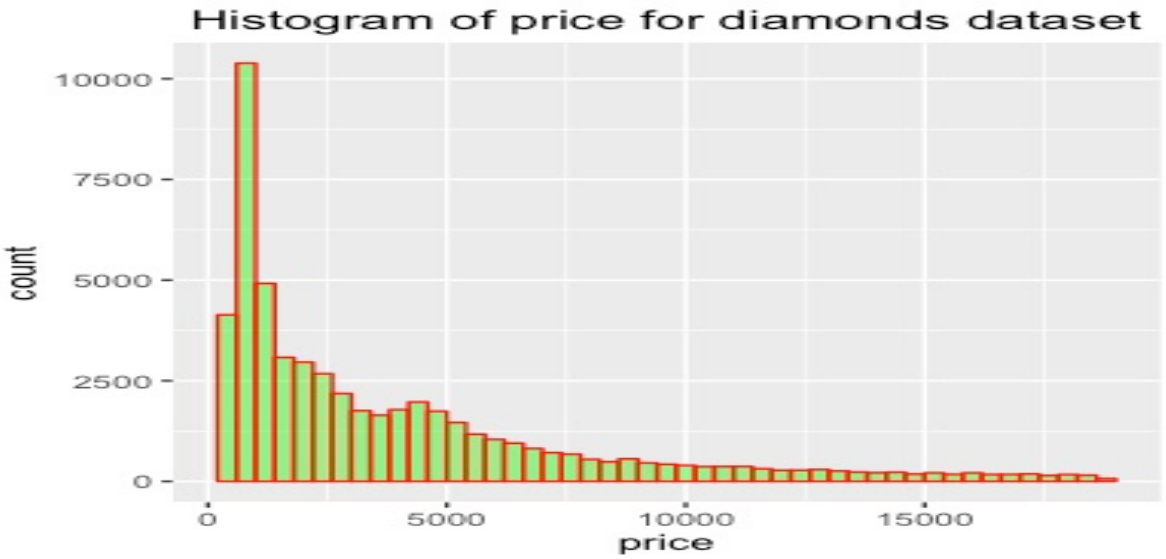
```
library(ggplot2)
library(reshape)
library(plyr)
Plot<-function()
{
 g_1<-ggplot(diamonds, aes(color)) + geom_bar(col="red",
                                fill="green",
                                alpha = .5)  + ggtitle("Histogram of color for diamonds dataset")
 print(g_1)

 g_1<-ggplot(diamonds, aes(carat)) + geom_histogram(col="red",
                                    fill="green",
                                    alpha = .5,
                                    binwidth =.2)  + ggtitle("Histogram of carat for diamonds dataset")
 print(g_1)

  g_1<-ggplot(diamonds, aes(price)) + geom_histogram(col="red",
                                     fill="green",
                                     alpha = .5,
                                     binwidth =400) + ggtitle("Histogram of price for diamonds
dataset")
  print(g_1)

  dsmall <- diamonds[sample(nrow(diamonds), 100), ]
  g_1<-ggplot(dsmall, aes(carat, price)) + geom_point(aes(shape=cut, colour = cut)) +
geom_smooth(span=0.2)
  print(g_1)
```

```
  g_1<-ggplot(diamonds, aes(carat, price)) + geom_point(aes(shape=cut)) +
geom_smooth(span=0.2)
  print(g_1)
}
```

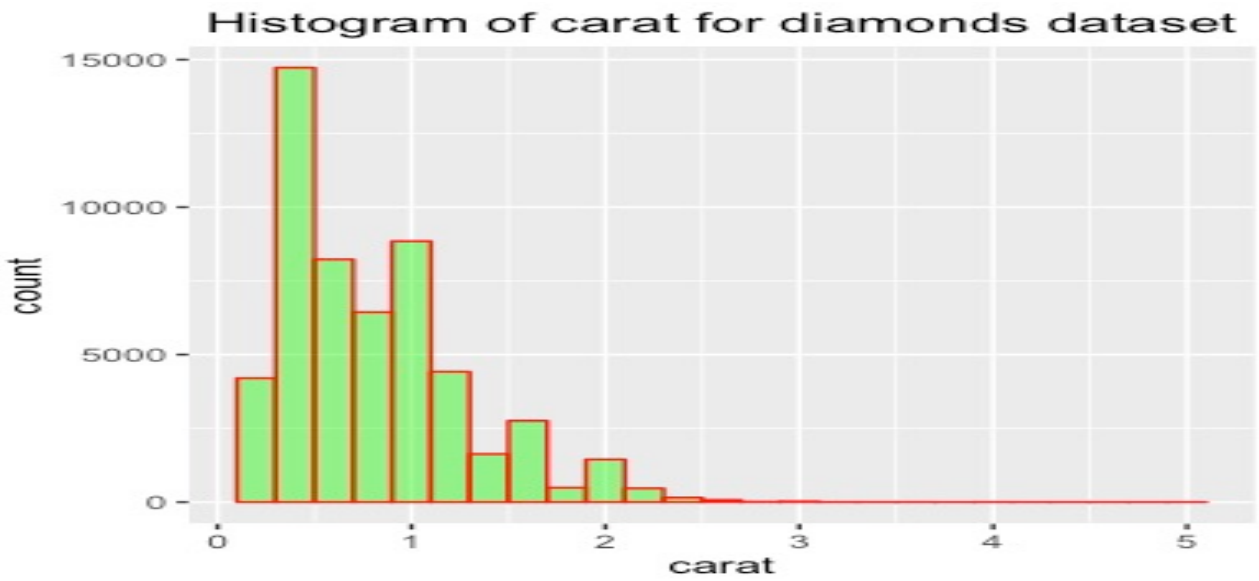
The below three plots show the histogram of color, carat and price of the diamonds dataset.



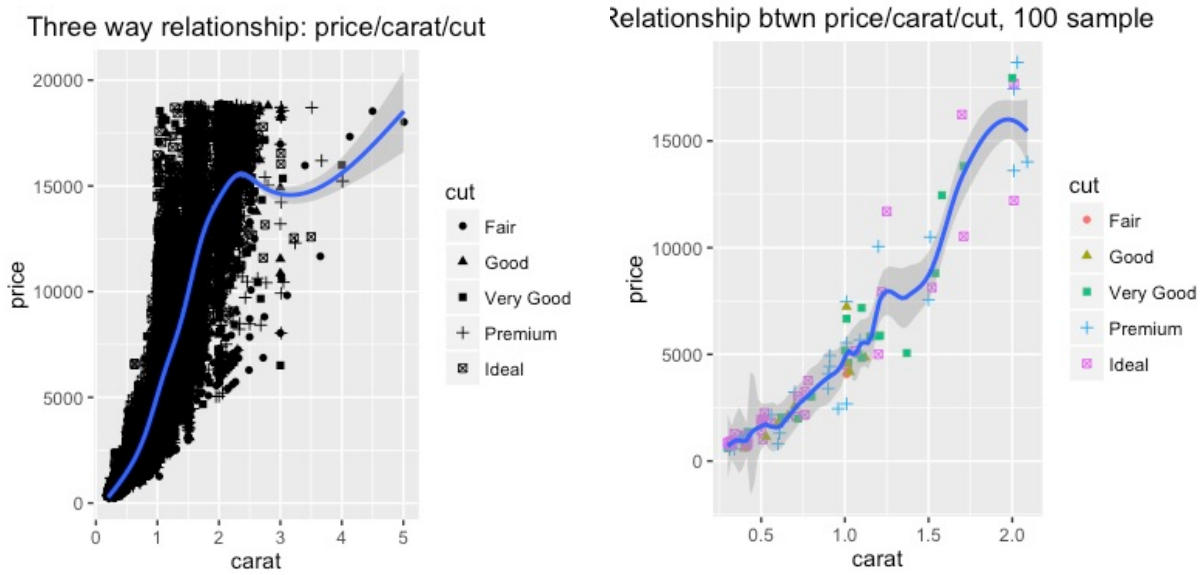
As can be seen above, the G color happens to be present the most number of times whereas the J color is most infrequent.



As can be seen above, most diamonds (>10000) are priced in the 1000-2000 dollar range and as the price of diamonds increases, their count also decreases.



The above carat histogram is very similar to the price histogram. Most diamonds are in the range of 0.2 carats and as the carat # increases, their count decreases.



The above two plots display the relationship between carat, price and cut. The plot on the left samples the entire dataset (53,940 diamonds) whereas the dataset on the right is a sample representation of only 100 diamonds from the dataset. This was necessary in order to avoid the dense population of cut data in first graph. Based on these plots, it is clear that as the carat count increases, the price of the diamond increases exponentially. Diamonds at the costly side tend to be either premium or ideal in cut whereas cheaper diamonds tend to contain all spectrums of cut.