Graphs and plots HW

Hamed

1/31/2020

1. Use the diamonds dataset that comes with R

```
library(ggplot2)
data("diamonds")
attach(diamonds)
head(diamonds)
```

```
## # A tibble: 6 x 10
##   carat cut       color clarity depth table price     x     y     z
##   <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
## 1 0.23  Ideal     E     SI2      61.5    55   326  3.95  3.98  2.43
## 2 0.21  Premium   E     SI1      59.8    61   326  3.89  3.84  2.31
## 3 0.23  Good      E     VS1      56.9    65   327  4.05  4.07  2.31
## 4 0.290 Premium   I     VS2      62.4    58   334  4.2   4.23  2.63
## 5 0.31  Good      J     SI2      63.3    58   335  4.34  4.35  2.75
## 6 0.24  Very Good J     VVS2     62.8    57   336  3.94  3.96  2.48
```

2. Count the number of rows by clarity

```
table(diamonds$clarity)
```

```
##
##    I1   SI2   SI1   VS2   VS1  VVS2  VVS1    IF
##   741  9194 13065 12258  8171  5066  3655  1790
```

3. What are the unique values for cut?

```
unique(cut)
```

```
## [1] Ideal     Premium   Good      Very Good Fair
## Levels: Fair < Good < Very Good < Premium < Ideal
```

4. Tabulate the frequency (no of diamonds) by cut

```
table(diamonds$cut)
```

```
##
##      Fair      Good Very Good   Premium     Ideal
##      1610      4906     12082     13791     21551
```

5. Find the mean carat size by color

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
diamonds %>%
  group_by(color) %>%
  summarise_at(vars(carat), funs(mean(., na.rm=TRUE)))
```

```
## Warning: funs() is soft deprecated as of dplyr 0.8.0
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## This warning is displayed once per session.
```

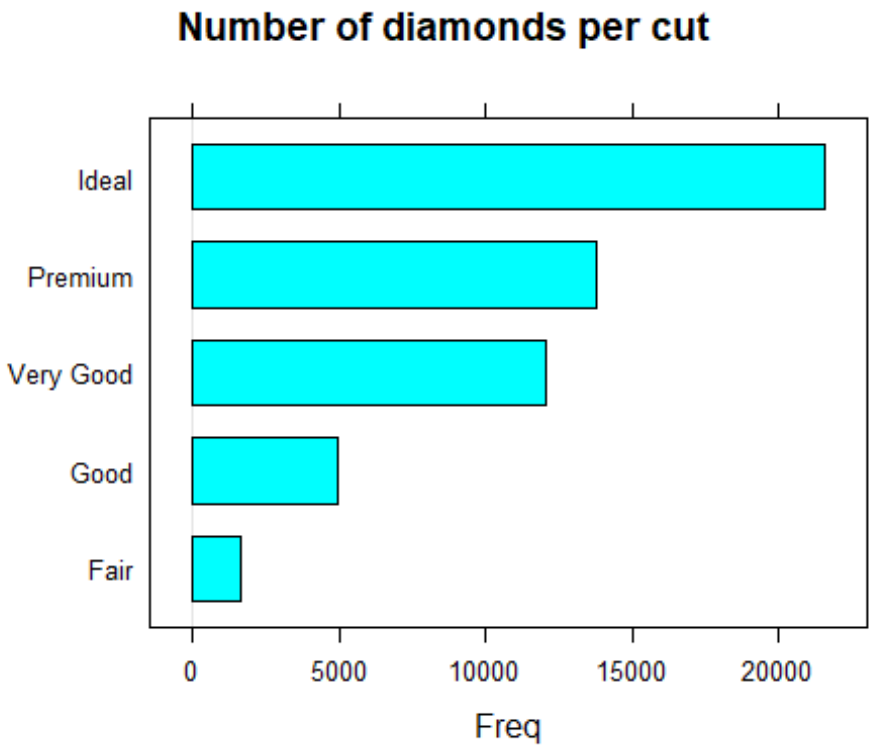
```
## # A tibble: 7 x 2
##   color carat
##   <ord> <dbl>
## 1 D     0.658
## 2 E     0.658
## 3 F     0.737
## 4 G     0.771
## 5 H     0.912
## 6 I     1.03
## 7 J     1.16
```

6. Use a bar chart to find which cut has the most amount of diamonds

```
require(lattice)
```

```
## Loading required package: lattice
```

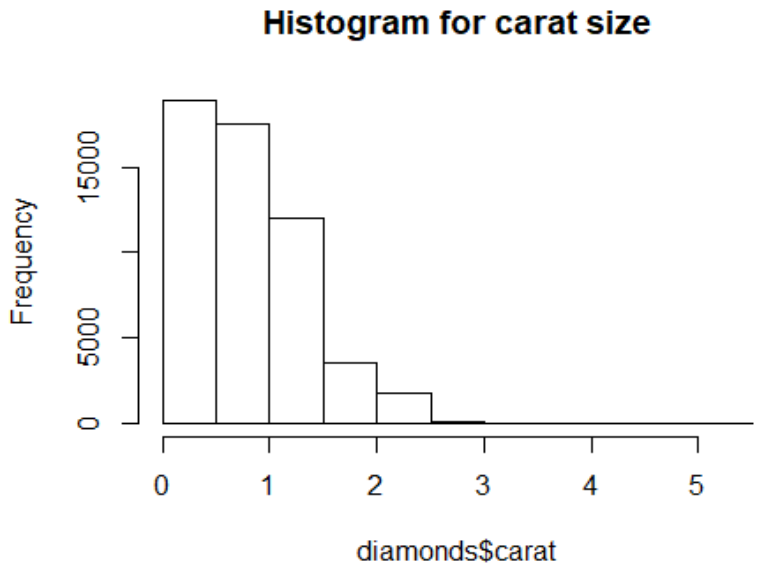
```
barchart(table(diamonds$cut),main="Number of diamonds per cut")
```



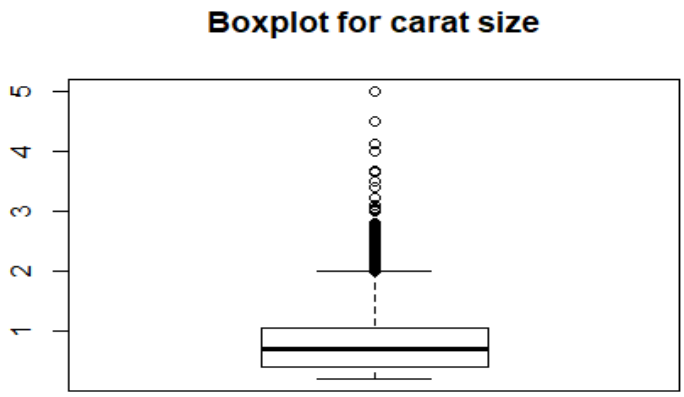
7. Check the distribution of the carat size. How can you describe the distribution?

- According to the plots the data is mostly found around the mean which is 0.79 but has a lot of outliers and thus the data is rightly skewed.

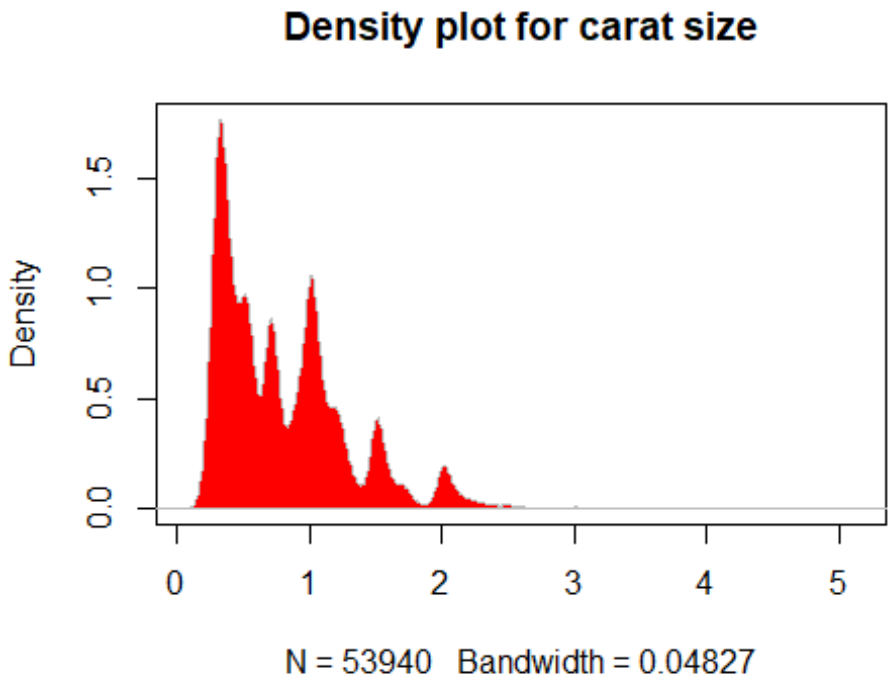
```
# Histogram plot
hist(diamonds$carat,main = "Histogram for carat size")
```



```r
# Boxplot
boxplot(diamonds$carat,main="Boxplot for carat size")
```



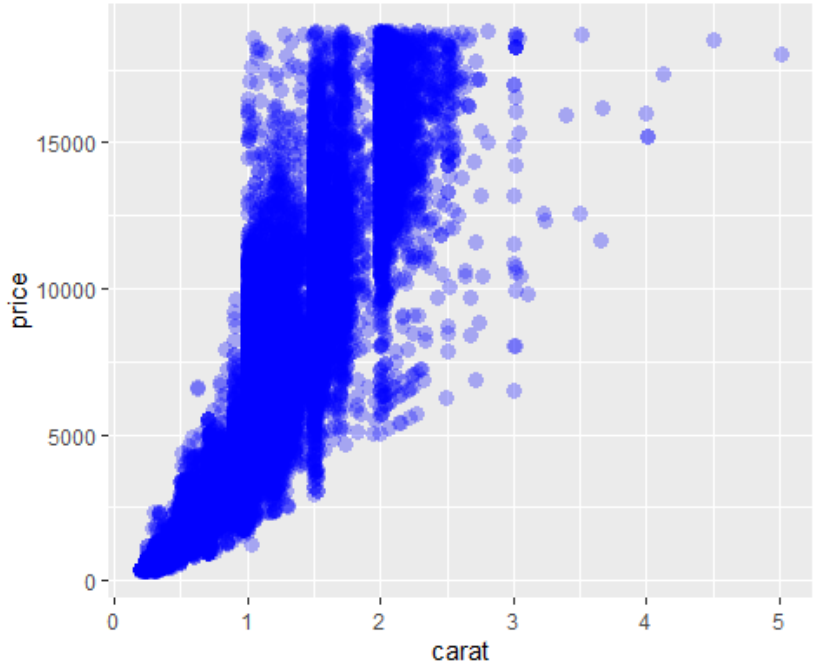

```r
#Density plot
d <- density(diamonds$carat)
plot(d, type="n", main="Density plot for carat size")
polygon(d, col="red", border="gray")
```



8. Is there a relationship between carat size and price? Check using a graph.

- From the graph the there is a positive slope showing carat size has a positive relationship with price.

```
ggplot(diamonds, aes(carat,price)) + geom_point(alpha=0.3,col="#0000ff22",
pch=16,cex=3)
```



9. Which color has the maximum variability in the price? Use a graph to find out.

-From the boxplots the color E shows higher variability in price.

```
qplot(color,price,data=diamonds, geom = "boxplot")
```