Pratik Bhujade

01/11/2019

Question 1

1. Created the dataset, and cleaning it

```
# Importing the tidyR and zoo Library for analysis
library(tidyr)
library(zoo)
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
# Inserting the dataset
df <- read.csv('DiamondData.csv')
# Removing the spelling mistake in cuts.
df_cleaning <- df
df_cleaning$cut[df_cleaning$cut == "Very Geod"] <-"Very Good"
# Insert means rows in rows with na value
df_clean <- data.frame('cut'= df_cleaning$cut, 'color' = df_cleaning$color, 'clarity' = df_cleaning$cla
df_clean$carat <- na.aggregate(df_cleaning$carat)
df_clean$table <- na.aggregate(df_cleaning$table)
df_clean$x <- na.aggregate(df_cleaning$x)
df_clean$y <- na.aggregate(df_cleaning$y)
df_clean$z <- na.aggregate(df_cleaning$z)
df_cleaning$depthh <- (2*df_clean$z/(df_clean$x+df_clean$y))*100
df_clean$depth <- na.aggregate(df_cleaning$depthh)
df_clean$price <- na.aggregate(df_cleaning$price)
df_cleaned <- df_clean
#names(df_cleaned) <- c('cut','color','clarity','carat','table','x','y','z','depth')
summary(df_clean)
```

```
##         cut        color        clarity          carat
##  Fair     : 1480   D: 6264   SI1    :12120   Min.   : 0.200
##  Good     : 4559   E: 9066   VS2    :11406   1st Qu.: 0.400
##  Ideal    :19918   F: 8837   SI2    : 8486   Median : 0.700
##  Premium  :12826   G:10493   VS1    : 7563   Mean   : 0.907
##  Very Geod:    0   H: 7705   VVS2   : 4692   3rd Qu.: 1.050
##  Very Good:11217   I: 5028   VVS1   : 3377   Max.   :49.990
##                    J: 2607   (Other): 2356
##      table             x                y                z
##  Min.   :43.00   Min.   : 0.000   Min.   : 0.000   Min.   : 0.00
##  1st Qu.:56.00   1st Qu.: 4.710   1st Qu.: 4.720   1st Qu.: 2.91
```

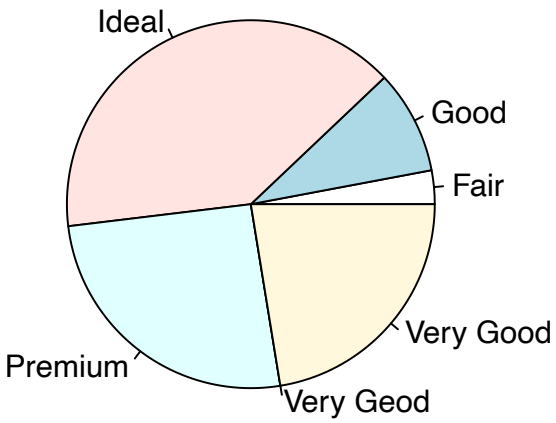
```
##  Median :57.00   Median : 5.700   Median : 5.720   Median : 3.53
##  Mean   :57.46   Mean   : 5.732   Mean   : 5.734   Mean   : 3.54
##  3rd Qu.:59.00   3rd Qu.: 6.540   3rd Qu.: 6.530   3rd Qu.: 4.03
##  Max.   :95.00   Max.   :10.230   Max.   :31.800   Max.   :31.80
##
##      depth            price
##  Min.   :  0.00   Min.   :  326
##  1st Qu.: 61.03   1st Qu.:  954
##  Median : 61.84   Median : 2426
##  Mean   : 61.77   Mean   : 3939
##  3rd Qu.: 62.54   3rd Qu.: 5315
##  Max.   :619.28   Max.   :18823
##
```

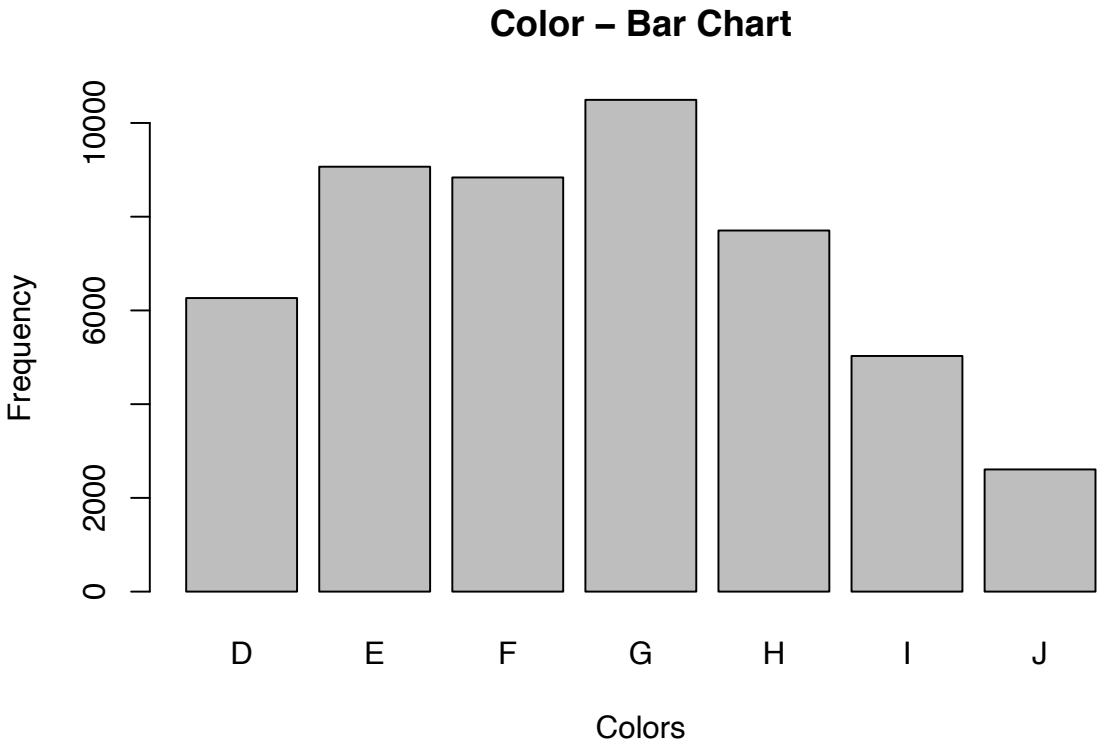
Question 2

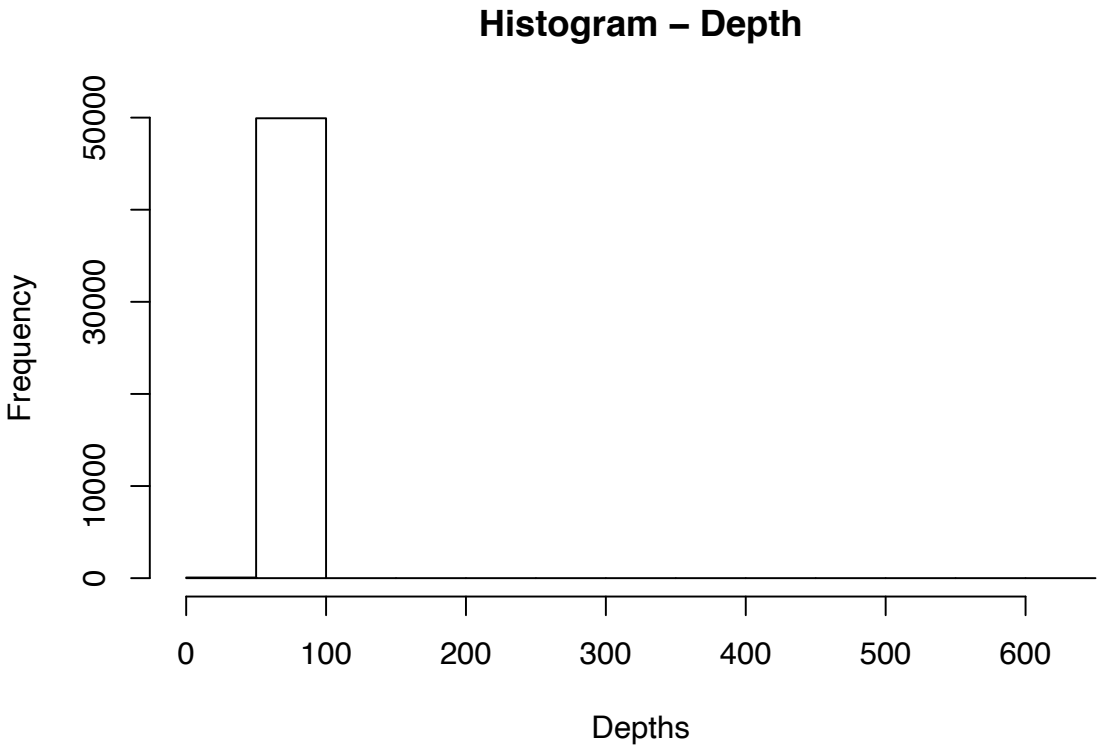
Summary and Plots

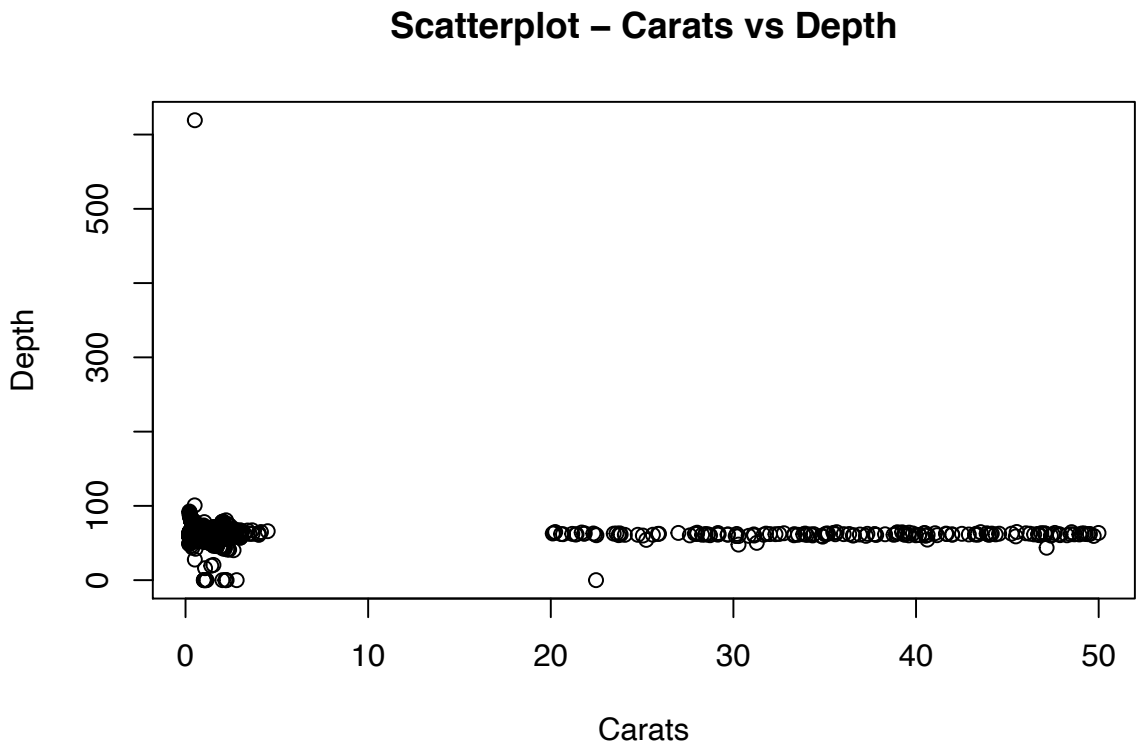
```
##         cut        color        clarity          carat
##  Fair     : 1480   D: 6264   SI1    :12120   Min.   : 0.200
##  Good     : 4559   E: 9066   VS2    :11406   1st Qu.: 0.400
##  Ideal    :19918   F: 8837   SI2    : 8486   Median : 0.700
##  Premium  :12826   G:10493   VS1    : 7563   Mean   : 0.907
##  Very Geod:    0   H: 7705   VVS2   : 4692   3rd Qu.: 1.050
##  Very Good:11217   I: 5028   VVS1   : 3377   Max.   :49.990
##                    J: 2607   (Other): 2356
##      table             x                y                z
##  Min.   :43.00   Min.   : 0.000   Min.   : 0.000   Min.   : 0.00
##  1st Qu.:56.00   1st Qu.: 4.710   1st Qu.: 4.720   1st Qu.: 2.91
##  Median :57.00   Median : 5.700   Median : 5.720   Median : 3.53
##  Mean   :57.46   Mean   : 5.732   Mean   : 5.734   Mean   : 3.54
##  3rd Qu.:59.00   3rd Qu.: 6.540   3rd Qu.: 6.530   3rd Qu.: 4.03
##  Max.   :95.00   Max.   :10.230   Max.   :31.800   Max.   :31.80
##
##      depth            price
##  Min.   :  0.00   Min.   :  326
##  1st Qu.: 61.03   1st Qu.:  954
##  Median : 61.84   Median : 2426
##  Mean   : 61.77   Mean   : 3939
##  3rd Qu.: 62.54   3rd Qu.: 5315
##  Max.   :619.28   Max.   :18823
##
```

Cut – Pie Chart



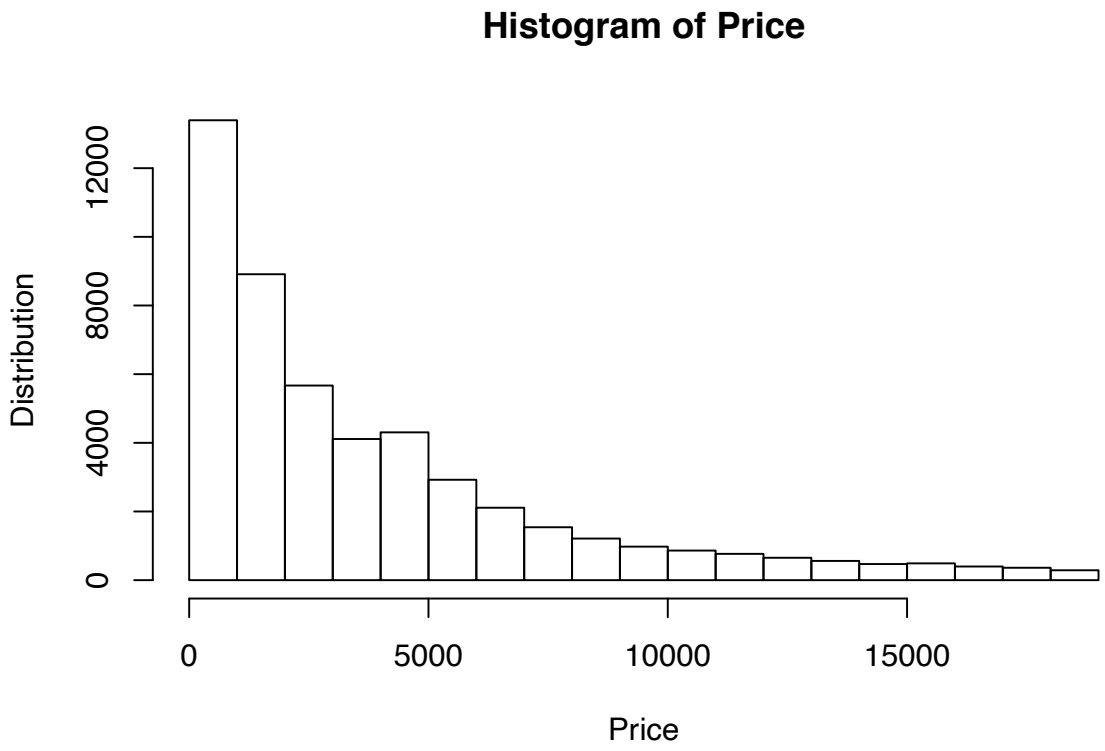

Color – Bar Chart
Frequency
0
2000
6000
10000
D
E
F
G
H
I
J
Colors


Histogram – Depth
Frequency
0
10000
30000
50000
0
100
200
300
400
500
600
Depths



Question 3 ### Part A Histogram of price along with the summary Statistics

```
# importing psych library for statistics
library(psych)
# histogram for prices
hist(df_cleaned$price, xlab = "Price", ylab = "Distribution", main = "Histogram of Price")
```




```
# Detailed summary statistics from psych library
describe(df_cleaned$price)
```

```
##    vars     n   mean      sd median trimmed     mad min   max range skew
## X1    1 50000 3938.8 3984.52   2426  3166.3 2501.15 326 18823 18497 1.62
##    kurtosis    se
## X1     2.18 17.82
```

The Histogram indicates that the distribution is positively skewed, and concentrated to the left. It also indicates that the price is not a normal distribtion, and the higest number of units lie in the distribution that is less than 5000 price. The summary statistics indicate that mean is 3938.64, and the media is 2401. Moreover the standard deviation is 3994.8.

Part B

```
# Low range 0-2401, first quartile
price_low <- 2401
# Medium Range from 2402-5345, Third quartile
price_med <- 5345
# High range greater than 5345
low <- subset(df_cleaned, price <= price_low)
med <- subset(df_cleaned, price <= price_med)
high <- subset(df_cleaned, price > price_med)
```

```r
# Summary statistics for low, medium and high
summary(low)
```

```
##         cut        color       clarity         carat
##  Fair     :  512   D:3595   VS2    :6005   Min.   : 0.200
##  Good     : 1979   E:5309   SI1    :5335   1st Qu.: 0.320
##  Ideal    :11636   F:4532   VS1    :4129   Median : 0.400
##  Premium  : 5538   G:5442   VVS2   :3057   Mean   : 0.537
##  Very Geod:    0   H:3214   VVS1   :2520   3rd Qu.: 0.520
##  Very Good: 5211   I:1905   SI2    :2352   Max.   :49.728
##                    J: 879   (Other):1478
##      table             x               y               z
##  Min.   :44.00   Min.   :0.000   Min.   : 0.00   Min.   : 0.000
##  1st Qu.:56.00   1st Qu.:4.390   1st Qu.: 4.40   1st Qu.: 2.710
##  Median :57.00   Median :4.710   Median : 4.73   Median : 2.910
##  Mean   :57.11   Mean   :4.801   Mean   : 4.81   Mean   : 2.969
##  3rd Qu.:58.00   3rd Qu.:5.180   3rd Qu.: 5.18   3rd Qu.: 3.200
##  Max.   :79.00   Max.   :6.860   Max.   :31.80   Max.   :31.800
##
##      depth            price
##  Min.   :  0.00   Min.   : 326
##  1st Qu.: 61.07   1st Qu.: 694
##  Median : 61.79   Median : 949
##  Mean   : 61.80   Mean   :1126
##  3rd Qu.: 62.42   3rd Qu.:1574
##  Max.   :619.28   Max.   :2401
##
```

```r
summary(med)
```

```
##         cut        color       clarity         carat
##  Fair     : 1118   D:5207   SI1    :9238   Min.   : 0.2000
##  Good     : 3509   E:7625   VS2    :8110   1st Qu.: 0.3400
##  Ideal    :15691   F:6902   SI2    :6157   Median : 0.5200
##  Premium  : 8840   G:7542   VS1    :5468   Mean   : 0.6955
##  Very Geod:    0   H:5397   VVS2   :3700   3rd Qu.: 0.7600
##  Very Good: 8420   I:3274   VVS1   :2961   Max.   :49.9904
##                    J:1631   (Other):1944
##      table             x               y                z
##  Min.   :43.00   Min.   :0.000   Min.   : 0.000   Min.   : 0.000
##  1st Qu.:56.00   1st Qu.:4.510   1st Qu.: 4.520   1st Qu.: 2.780
##  Median :57.00   Median :5.170   Median : 5.180   Median : 3.200
##  Mean   :57.36   Mean   :5.262   Mean   : 5.268   Mean   : 3.253
##  3rd Qu.:59.00   3rd Qu.:5.870   3rd Qu.: 5.870   3rd Qu.: 3.620
##  Max.   :79.00   Max.   :8.540   Max.   :31.800   Max.   :31.800
##
##      depth            price
##  Min.   :  0.00   Min.   : 326
##  1st Qu.: 61.05   1st Qu.: 812
##  Median : 61.83   Median :1588
##  Mean   : 61.80   Mean   :2030
##  3rd Qu.: 62.53   3rd Qu.:3053
##  Max.   :619.28   Max.   :5345
```

```
##
```

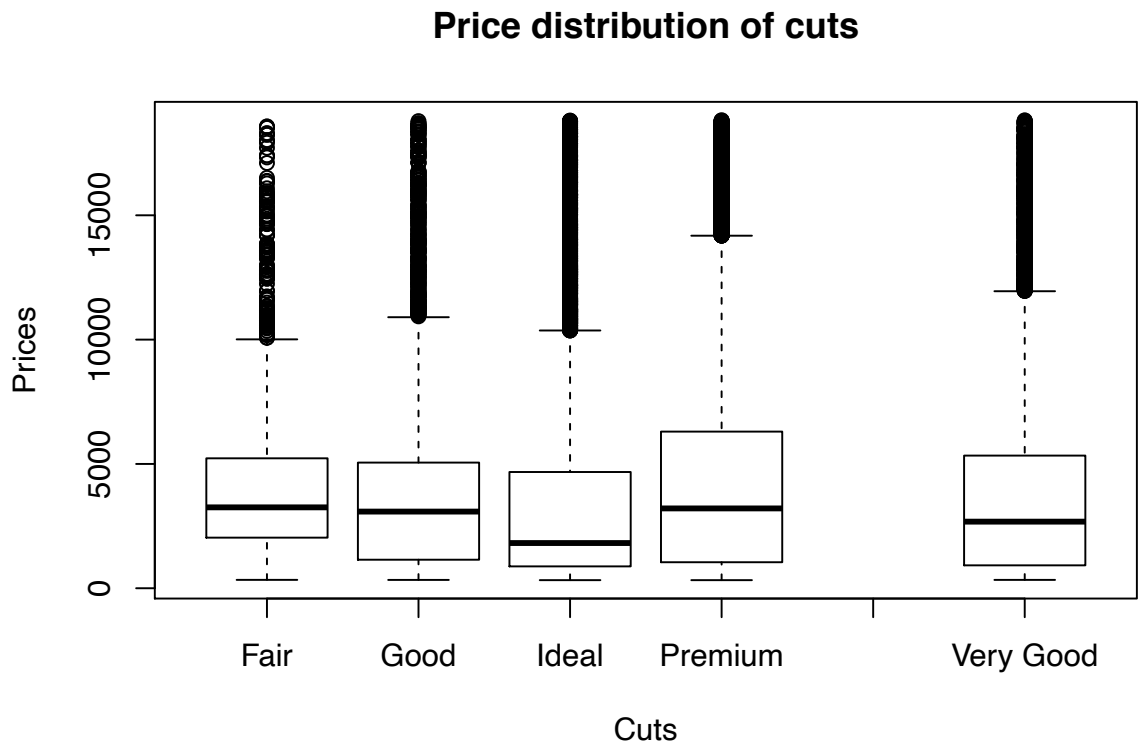
```
summary(high)
```

```
##         cut       color       clarity         carat            table
##  Fair     : 362   D:1057   VS2    :3296   Min.   : 0.630   Min.   :50.00
##  Good     :1050   E:1441   SI1    :2882   1st Qu.: 1.090   1st Qu.:56.00
##  Ideal    :4227   F:1935   SI2    :2329   Median : 1.330   Median :58.00
##  Premium  :3986   G:2951   VS1    :2095   Mean   : 1.547   Mean   :57.79
##  Very Geod:   0   H:2308   VVS2   : 992   3rd Qu.: 1.600   3rd Qu.:59.00
##  Very Good:2797   I:1754   VVS1   : 416   Max.   :49.521   Max.   :95.00
##                   J: 976   (Other): 412
##        x                y                z             depth
##  Min.   : 0.000   Min.   : 0.000   Min.   :0.000   Min.   : 0.00
##  1st Qu.: 6.630   1st Qu.: 6.630   1st Qu.:4.080   1st Qu.:60.95
##  Median : 7.050   Median : 7.050   Median :4.340   Median :61.88
##  Mean   : 7.154   Mean   : 7.143   Mean   :4.407   Mean   :61.68
##  3rd Qu.: 7.530   3rd Qu.: 7.520   3rd Qu.:4.630   3rd Qu.:62.58
##  Max.   :10.230   Max.   :10.160   Max.   :6.720   Max.   :80.57
##
##      price
##  Min.   : 5346
##  1st Qu.: 6609
##  Median : 8696
##  Mean   : 9713
##  3rd Qu.:12183
##  Max.   :18823
##
```

Part C

Box plot for different cuts

```
# boxplot for prices, based on different cuts
boxplot(df_cleaned$price~df_cleaned$cut, xlab = "Cuts", ylab = "Prices",main = "Price distribution of cu
```



Part D

```
# creating the dataframe for numerical variables
n_df <- data.frame(df_cleaned$carat,df_cleaned$price,df_cleaned$depth,df_cleaned$table,df_cleaned$x,df_
# Correlation for numerical variables
cor(n_df)
```

```
##                  df_cleaned.carat df_cleaned.price df_cleaned.depth
## df_cleaned.carat      1.000000000        0.2132887     -0.009724909
## df_cleaned.price      0.213288669        1.0000000     -0.020770698
## df_cleaned.depth     -0.009724909       -0.0207707      1.000000000
## df_cleaned.table      0.039761303        0.1270487     -0.132510964
## df_cleaned.x          0.223011294        0.8801776     -0.038237138
## df_cleaned.y          0.221418092        0.8756857     -0.050007328
## df_cleaned.z          0.214743207        0.8541778      0.220431188
##                  df_cleaned.table df_cleaned.x df_cleaned.y df_cleaned.z
## df_cleaned.carat        0.0397613   0.22301129   0.22141809    0.2147432
## df_cleaned.price        0.1270487   0.88017761   0.87568573    0.8541778
## df_cleaned.depth       -0.1325110  -0.03823714  -0.05000733    0.2204312
## df_cleaned.table        1.0000000   0.19389482   0.18654068    0.1495847
## df_cleaned.x            0.1938948   1.00000000   0.98714681    0.9629605
## df_cleaned.y            0.1865407   0.98714681   1.00000000    0.9570021
## df_cleaned.z            0.1495847   0.96296045   0.95700213    1.0000000
```

Question 4

Frequencies and Scatterplots

```
# Importing ggplot for visualization
library(ggplot2)
```

```
##
## Attaching package: 'ggplot2'
```

```
## The following objects are masked from 'package:psych':
##
##     %+%, alpha
```

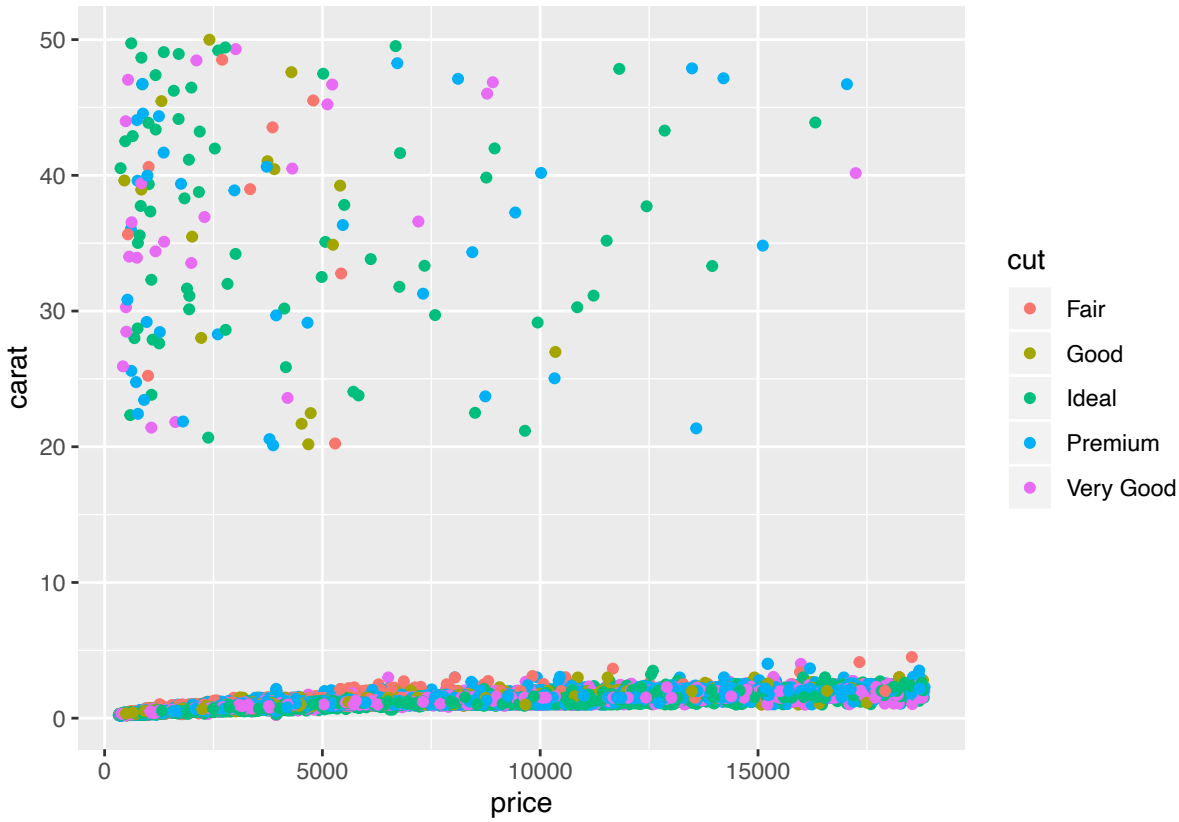
```
# SUmmary for different cuts
summary(df_cleaned$cut)
```

```
##      Fair      Good     Ideal   Premium Very Geod Very Good
##      1480      4559     19918     12826         0     11217
```

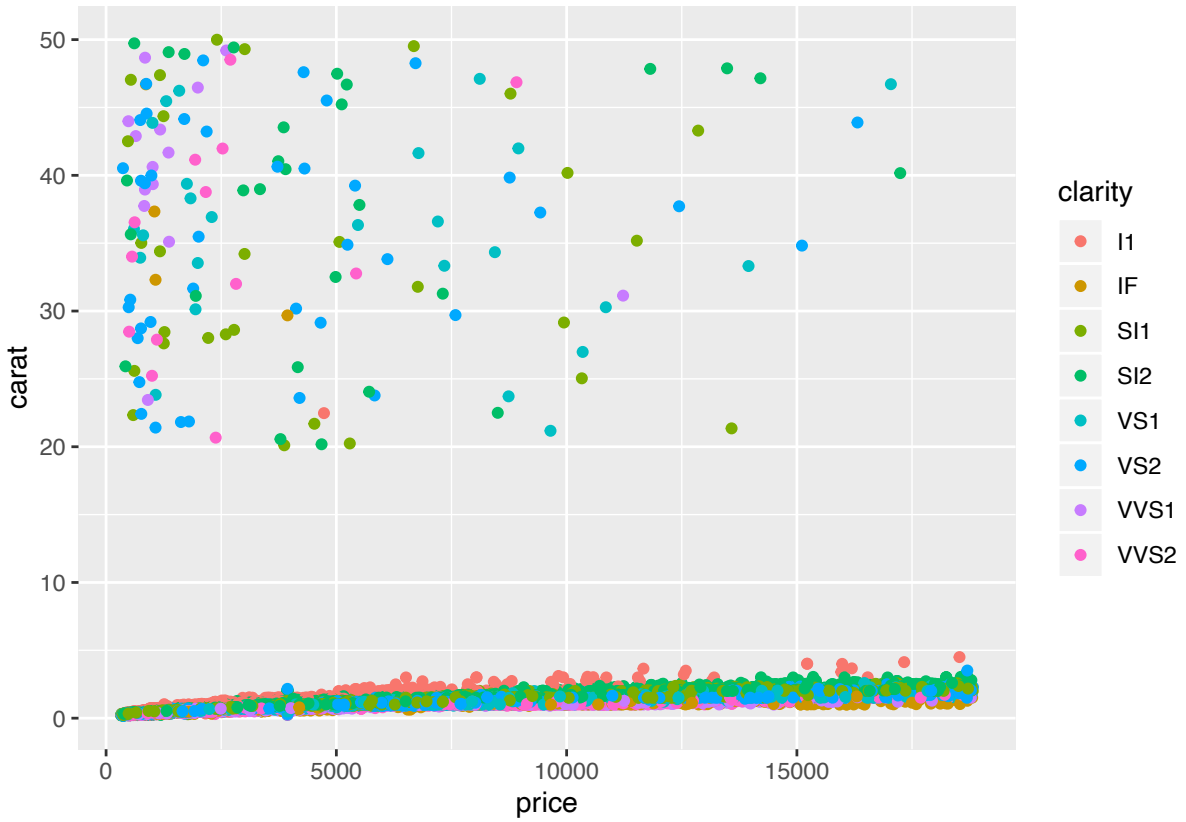
```
# Summary for different clarities
summary(df_cleaned$clarity)
```

```
##    I1    IF   SI1   SI2   VS1   VS2  VVS1  VVS2
##   690  1666 12120  8486  7563 11406  3377  4692
```

```
# Relationship between carat and price on the basis of cuts
ggplot(df_cleaned, aes(price, carat, color = cut)) + geom_point()
```

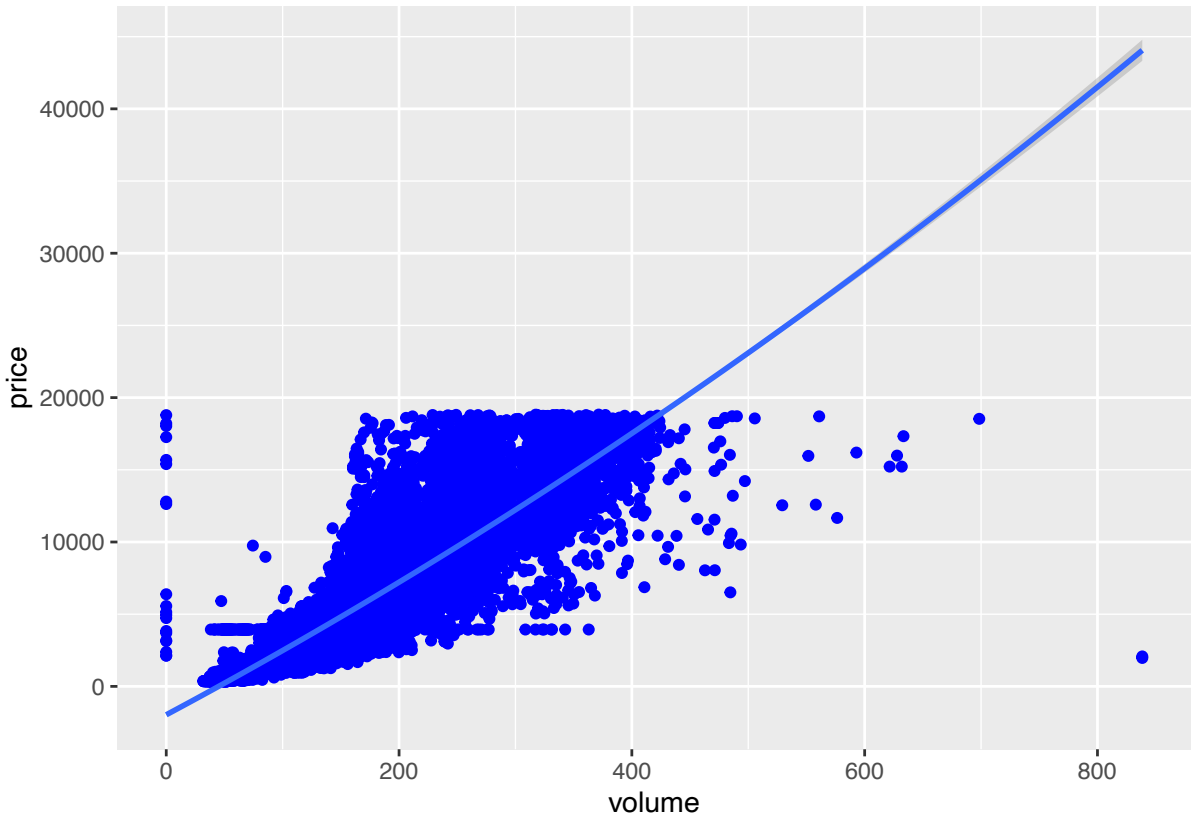



```
# Relationship between carat and price on the basis of clarities
ggplot(df_cleaned, aes(price, carat, color = clarity)) + geom_point()
```



Question 5 ### Part A

```
# creating volume dataframe
volume =data.frame(df_cleaned$x, df_cleaned$y, df_cleaned$z)
# putting names in it
names(volume) <- c('x','y','z')
# calculating volume
volume$volume <- volume$x*volume$y*volume$z
# adding price to the mix
volume$price <- df_cleaned$price
# ploting volume and price
ggplot(volume, aes(volume, price)) + geom_point(color='blue') + geom_smooth(method = 'lm', formula = y
```

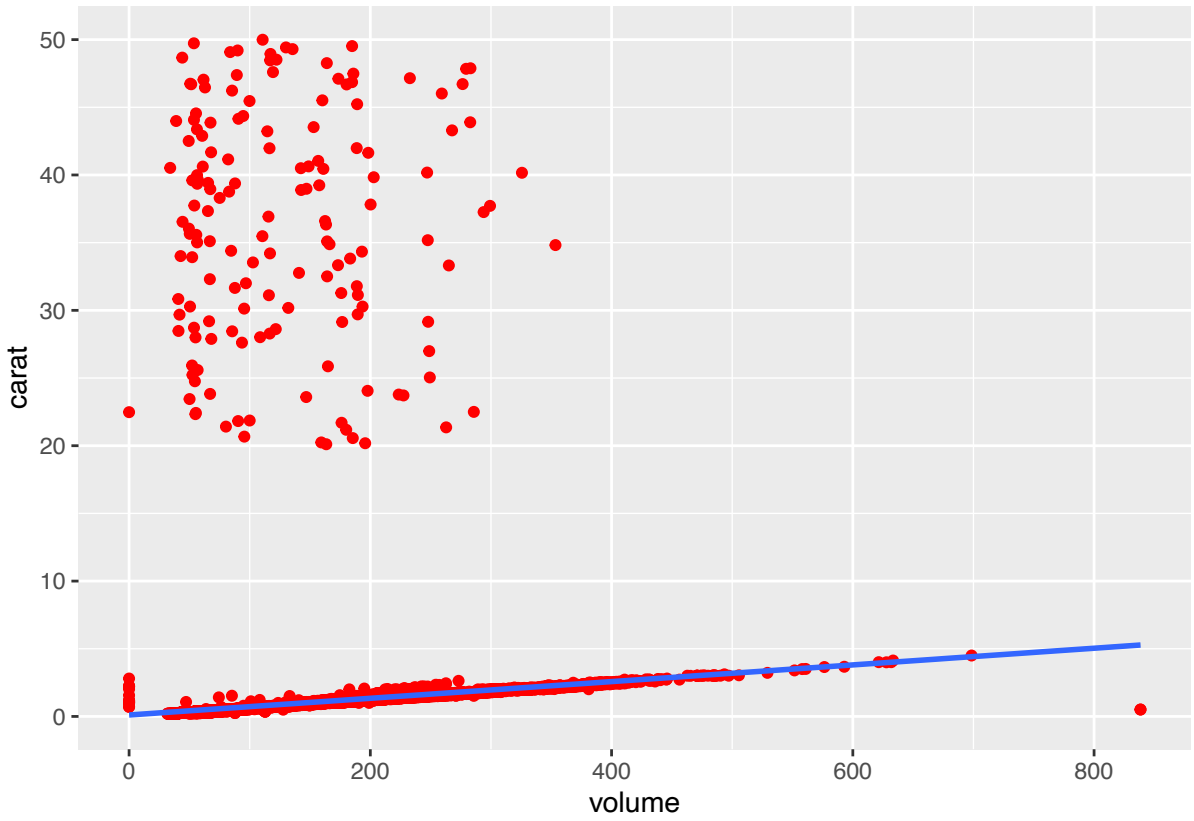


This indicates a polynomial regression, and it indicates that the regression is increasing in magnitude along with being positive.

Part B

Relationship between carat and volume

```
# adding carats to the mix
volume$carat <- df_cleaned$carat
# plotting volume and carat
ggplot(volume, aes(volume, carat)) + geom_point(color='red') + geom_smooth(method = 'lm', formula = y ~
```

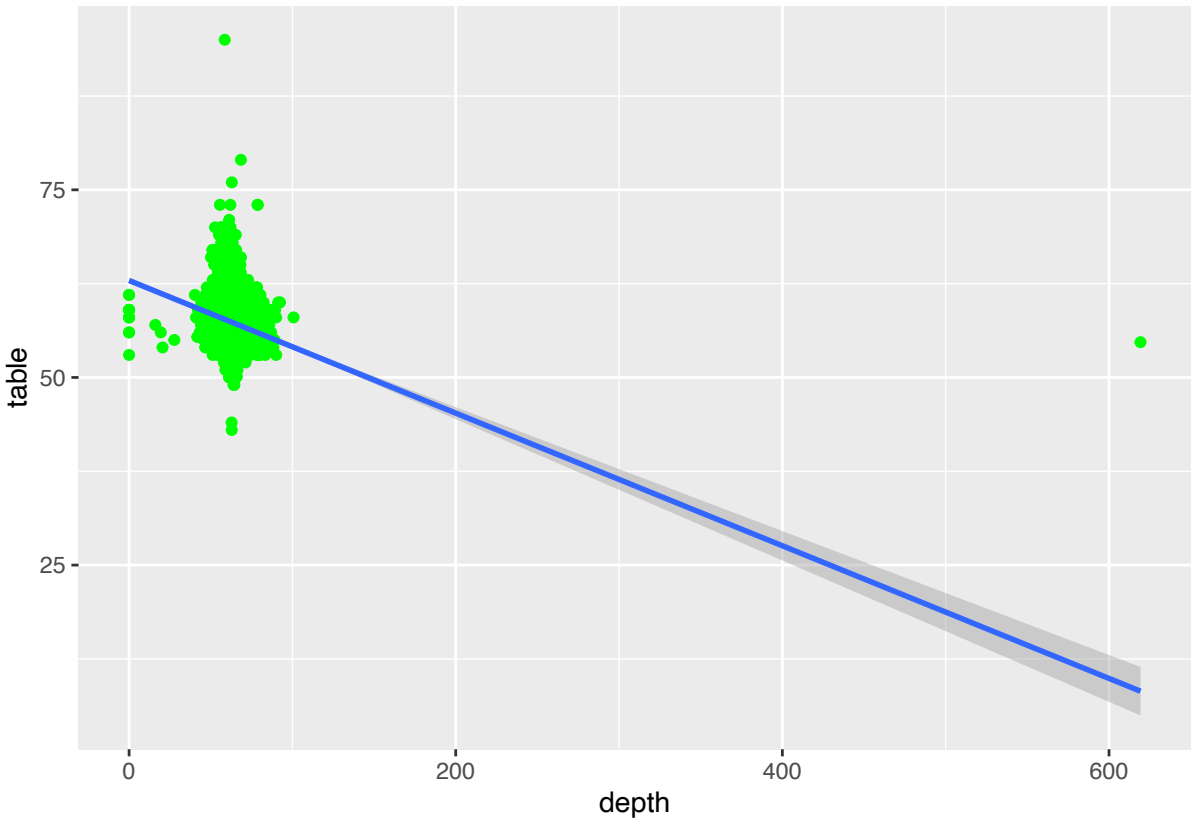


No linear relationship between volum and carats.

Part C

Relationship between table and depth

```
# Adding table and depth to the mix
volume$table <- df_cleaned$table
volume$depth <- df_cleaned$depth
# ploting table and depth
ggplot(volume, aes(depth, table)) + geom_point(color='green') + geom_smooth(method = 'lm', formula = y
```

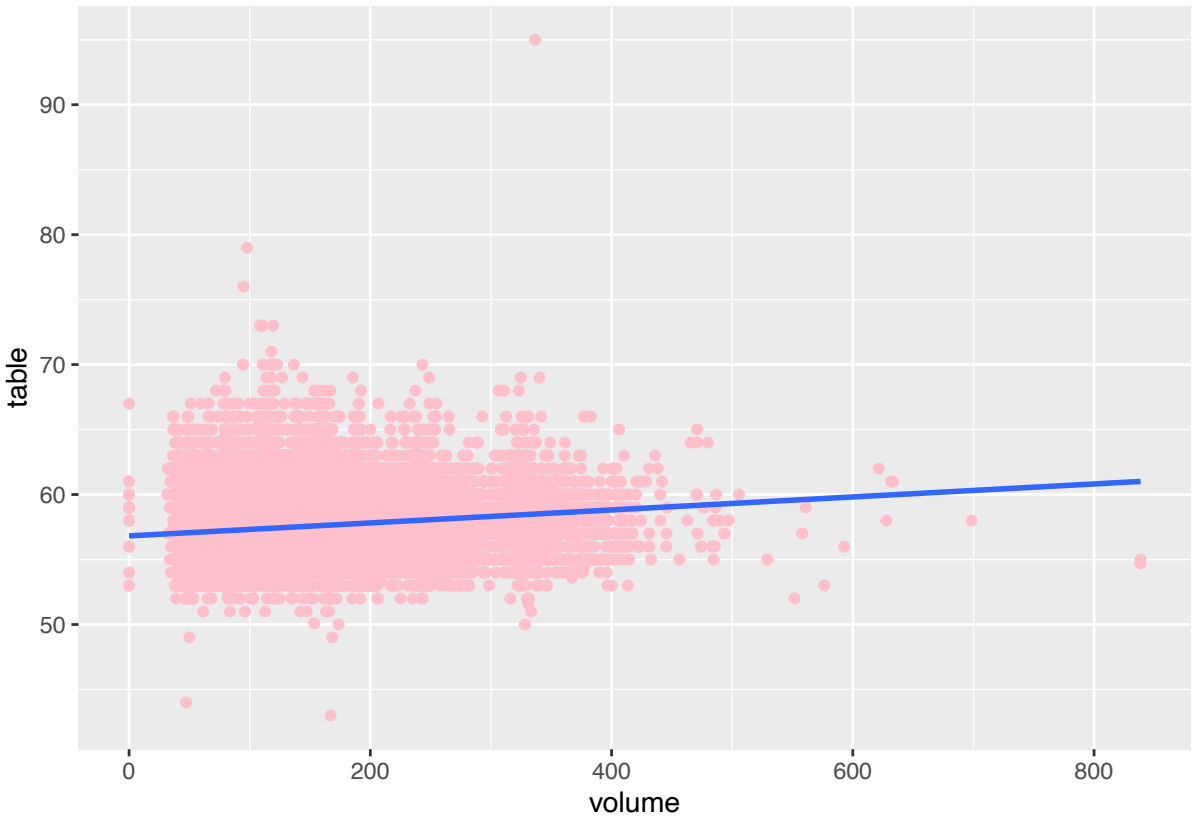


Negative relationship between depth and table ### Part D Relationship of table with all the variables

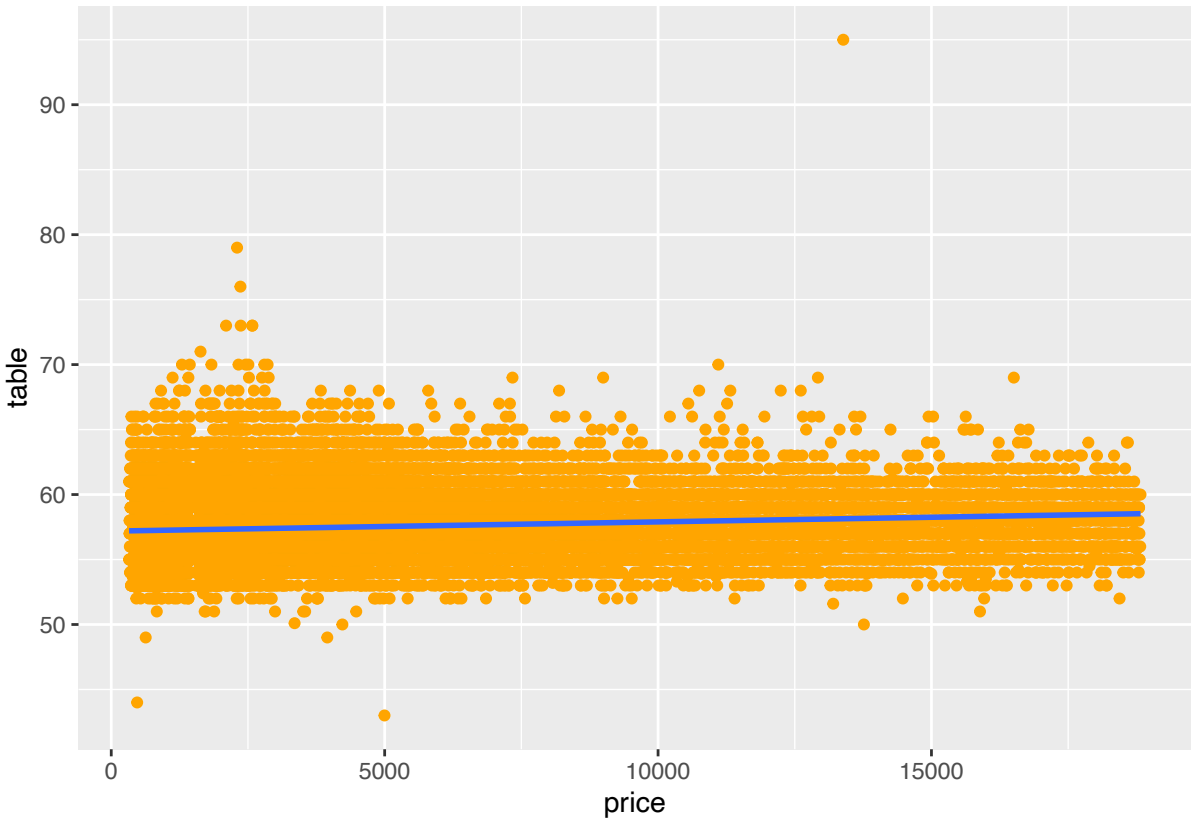
```
# finding the correlation between all the added variables
cor(volume)
```

```
##                  x           y         z     volume      price
## x       1.00000000  0.98714681 0.9629605 0.97326375  0.8801776
## y       0.98714681  1.00000000 0.9570021 0.97176259  0.8756857
## z       0.96296045  0.95700213 1.0000000 0.96191265  0.8541778
## volume  0.97326375  0.97176259 0.9619127 1.00000000  0.9162446
## price   0.88017761  0.87568573 0.8541778 0.91624455  1.0000000
## carat   0.22301129  0.22141809 0.2147432 0.22671334  0.2132887
## table   0.19389482  0.18654068 0.1495847 0.17033926  0.1270487
## depth  -0.03823714 -0.05000733 0.2204312 0.03456058 -0.0207707
##                carat      table        depth
## x       0.223011294  0.1938948 -0.038237138
## y       0.221418092  0.1865407 -0.050007328
## z       0.214743207  0.1495847  0.220431188
## volume  0.226713342  0.1703393  0.034560579
## price   0.213288669  0.1270487 -0.020770698
## carat   1.000000000  0.0397613 -0.009724909
## table   0.039761303  1.0000000 -0.132510964
## depth  -0.009724909 -0.1325110  1.000000000
```

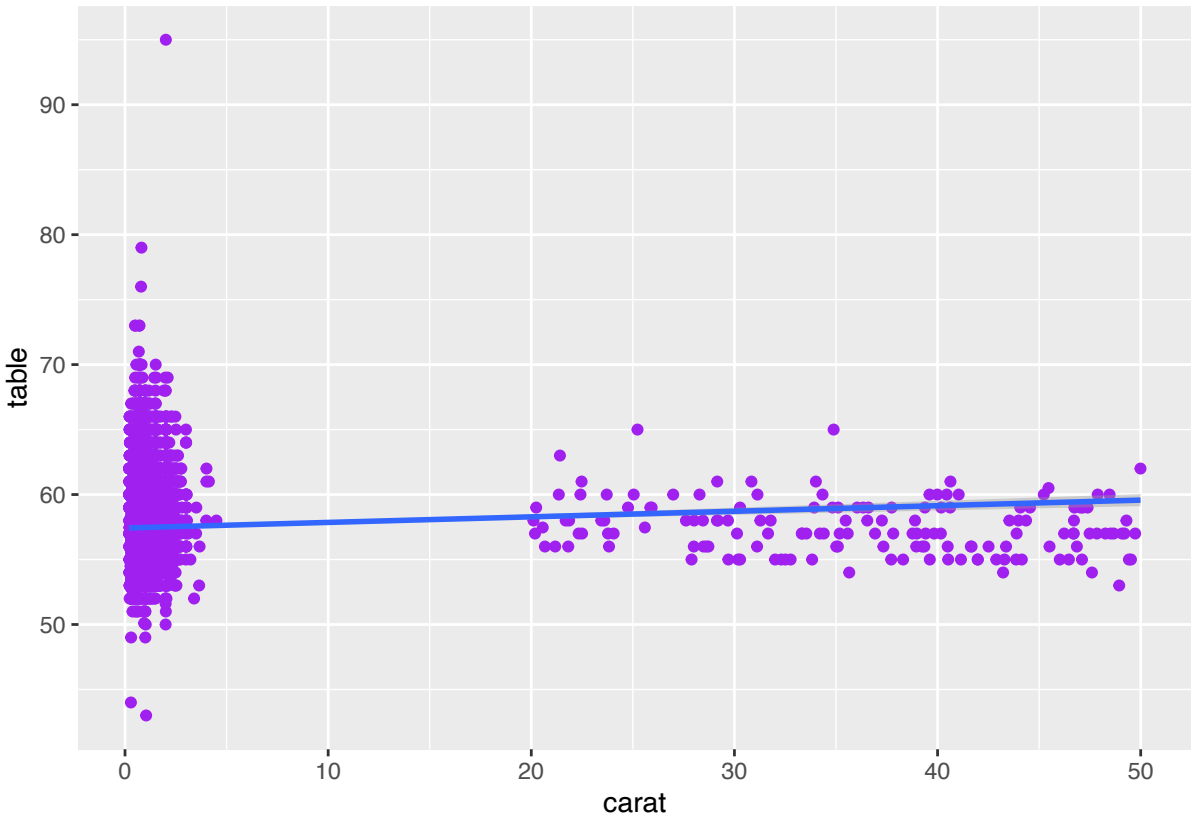
```
# ploting volume, price and carats with table
ggplot(volume, aes(volume, table)) + geom_point(color='pink') + geom_smooth(method = 'lm', formula = y
```




```
ggplot(volume, aes(price, table)) + geom_point(color='orange') + geom_smooth(method = 'lm', formula = y
```




```
ggplot(volume, aes(carat, table)) + geom_point(color='purple') + geom_smooth(method = 'lm', formula = y
```


90
80
70
60
50
table
0
10
20
30
40
50
carat