DATA_MANIPULATION_1

JAYARUTHRA M V

2024-08-07

```r
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
```

```r
library(lattice)
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```r
#View(diamonds)
library(ggplot2)
#filter
filtered_data <- subset(diamonds, carat > 2 & price > 10000)
head(filtered_data)
```

```
## # 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  2.01 Very Good I     SI2      61.4    63 10009  8.19  7.96  4.96
## 2  2.09 Premium   I     SI2      60.1    59 10042  8.34  8.3   5
## 3  2.52 Fair      G     I1       66.9    57 10076  8.39  8.33  5.6
## 4  2.19 Premium   J     SI2      58.8    58 10179  8.57  8.53  5.03
## 5  2.02 Ideal     I     SI2      62.6    56 10181  8.05  8.01  5.03
## 6  2.09 Premium   H     SI2      61      60 10182  8.28  8.19  5.02
```

```r
diamonds$price_per_carat <- diamonds$price / diamonds$carat
head(diamonds)
```

```
## # A tibble: 6 x 11
##   carat cut     color clarity depth table price     x     y     z price_per_carat
##   <dbl> <ord>   <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>           <dbl>
## 1  0.23 Ideal   E     SI2      61.5    55   326  3.95  3.98  2.43           1417.
## 2  0.21 Premi~ E     SI1      59.8    61   326  3.89  3.84  2.31           1552.
## 3  0.23 Good   E     VS1      56.9    65   327  4.05  4.07  2.31           1422.
## 4  0.29 Premi~ I     VS2      62.4    58   334  4.2   4.23  2.63           1152.
## 5  0.31 Good   J     SI2      63.3    58   335  4.34  4.35  2.75           1081.
## 6  0.24 Very ~ J     VVS2     62.8    57   336  3.94  3.96  2.48           1400
```

```r
diamonds%>%filter(price > 18000)
```

```
## # A tibble: 312 x 11
##    carat cut   color clarity depth table price     x     y     z price_per_carat
##    <dbl> <ord> <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>           <dbl>
##  1  2.16 Ideal G     SI2      62.5  54.2 18001  8.23  8.27  5.16           8334.
##  2  2.09 Prem~ F     SI2      61.7  59   18002  8.23  8.21  5.07           8613.
##  3  2.18 Prem~ G     SI2      61.9  60   18003  8.29  8.24  5.12           8258.
##  4  2.06 Very~ G     SI2      62.3  59   18005  8.07  8.2   5.07           8740.
##  5  2.25 Prem~ D     SI2      60.4  59   18007  8.54  8.48  5.13           8003.
##  6  1.76 Very~ G     VS1      62.8  55.4 18014  7.7   7.74  4.85          10235.
##  7  2.05 Ideal G     SI2      61.6  56   18017  8.11  8.16  5.01           8789.
##  8  5.01 Fair  J     I1       65.5  59   18018 10.7  10.5   6.98           3596.
##  9  2.51 Prem~ J     VS2      62.2  58   18020  8.73  8.67  5.41           7179.
## 10  2    Good  H     VS2      63.8  59   18023  7.88  8.01  5.07           9012.
## # i 302 more rows
```

```r
diamonds%>%filter(price > 18810)
```

```
## # A tibble: 2 x 11
##   carat cut    color clarity depth table price     x     y     z price_per_carat
##   <dbl> <ord>  <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>           <dbl>
## 1  2    Very ~ G     SI1      63.5    56 18818   7.9  7.97  5.04           9409
## 2  2.29 Premi~ I     VS2      60.8    60 18823   8.5  8.47  5.16           8220.
```

```r
filter(diamonds,price==max(price))
```

```
## # A tibble: 1 x 11
##   carat cut    color clarity depth table price     x     y     z price_per_carat
##   <dbl> <ord>  <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>           <dbl>
## 1  2.29 Premi~ I     VS2      60.8    60 18823   8.5  8.47  5.16           8220.
```

```r
#select
selected_data <- diamonds[, c("carat", "cut", "price")]
head(selected_data)
```

```
## # A tibble: 6 x 3
##   carat cut       price
##   <dbl> <ord>     <int>
## 1  0.23 Ideal       326
## 2  0.21 Premium     326
## 3  0.23 Good        327
## 4  0.29 Premium     334
## 5  0.31 Good        335
## 6  0.24 Very Good   336
```

```r
diamonds %>%dplyr::select(clarity)
```

```
## # A tibble: 53,940 x 1
##    clarity
##    <ord>
##  1 SI2
##  2 SI1
##  3 VS1
##  4 VS2
##  5 SI2
##  6 VVS2
##  7 VVS1
##  8 SI1
##  9 VS2
## 10 VS1
## # i 53,930 more rows
```

```
diamonds%>%filter(color=='I')%>%dplyr::select(clarity,price)
```

```
## # A tibble: 5,422 x 2
##    clarity price
##    <ord>   <int>
##  1 VS2       334
##  2 VVS1      336
##  3 SI2       348
##  4 SI2       351
##  5 VS1       355
##  6 SI2       403
##  7 SI2       403
##  8 SI1       404
##  9 SI2       405
## 10 SI1       405
## # i 5,412 more rows
```

```
#mutate
diamonds%>%mutate(grade = if_else(carat <0.7, "A", "B"))
```

```
## # A tibble: 53,940 x 12
##    carat cut   color clarity depth table price     x     y     z price_per_carat
##    <dbl> <ord> <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>           <dbl>
##  1  0.23 Ideal E     SI2      61.5    55   326  3.95  3.98  2.43           1417.
##  2  0.21 Prem~ E     SI1      59.8    61   326  3.89  3.84  2.31           1552.
##  3  0.23 Good  E     VS1      56.9    65   327  4.05  4.07  2.31           1422.
##  4  0.29 Prem~ I     VS2      62.4    58   334  4.2   4.23  2.63           1152.
##  5  0.31 Good  J     SI2      63.3    58   335  4.34  4.35  2.75           1081.
##  6  0.24 Very~ J     VVS2     62.8    57   336  3.94  3.96  2.48           1400
##  7  0.24 Very~ I     VVS1     62.3    57   336  3.95  3.98  2.47           1400
##  8  0.26 Very~ H     SI1      61.9    55   337  4.07  4.11  2.53           1296.
##  9  0.22 Fair  E     VS2      65.1    61   337  3.87  3.78  2.49           1532.
## 10  0.23 Very~ H     VS1      59.4    61   338  4     4.05  2.39           1470.
## # i 53,930 more rows
## # i 1 more variable: grade <chr>
```

```
#arrange
diamonds%>%arrange('carat')
```

```
## # A tibble: 53,940 x 11
##    carat cut   color clarity depth table price     x     y     z price_per_carat
##    <dbl> <ord> <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>           <dbl>
##  1  0.23 Ideal E     SI2      61.5    55   326  3.95  3.98  2.43           1417.
##  2  0.21 Prem~ E     SI1      59.8    61   326  3.89  3.84  2.31           1552.
##  3  0.23 Good  E     VS1      56.9    65   327  4.05  4.07  2.31           1422.
##  4  0.29 Prem~ I     VS2      62.4    58    334  4.2   4.23  2.63           1152.
##  5  0.31 Good  J     SI2      63.3    58    335  4.34  4.35  2.75           1081.
##  6  0.24 Very~ J     VVS2     62.8    57    336  3.94  3.96  2.48           1400
##  7  0.24 Very~ I     VVS1     62.3    57    336  3.95  3.98  2.47           1400
##  8  0.26 Very~ H     SI1      61.9    55    337  4.07  4.11  2.53           1296.
##  9  0.22 Fair  E     VS2      65.1    61    337  3.87  3.78  2.49           1532.
## 10  0.23 Very~ H     VS1      59.4    61    338  4     4.05  2.39           1470.
## # i 53,930 more rows
```

```r
sorted_data <- diamonds[order(diamonds$carat, decreasing = TRUE), ]
head(sorted_data)
```

```
## # A tibble: 6 x 11
##   carat cut    color clarity depth table price     x     y     z price_per_carat
##   <dbl> <ord>  <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>           <dbl>
## 1  5.01 Fair   J     I1       65.5    59 18018  10.7 10.5   6.98           3596.
## 2  4.5  Fair   J     I1       65.8    58 18531  10.2 10.2   6.72           4118
## 3  4.13 Fair   H     I1       64.8    61 17329  10    9.85  6.43           4196.
## 4  4.01 Premi~ I     I1       61      61 15223  10.1 10.1   6.17           3796.
## 5  4.01 Premi~ J     I1       62.5    62 15223  10.0  9.94  6.24           3796.
## 6  4    Very ~ I     I1       63.3    58 15984  10.0  9.94  6.31           3996
```

```r
#groupby
diamonds%>%group_by(carat)%>%mutate(price_carat=price/carat)
```

```
## # A tibble: 53,940 x 12
## # Groups:   carat [273]
##    carat cut   color clarity depth table price     x     y     z price_per_carat
##    <dbl> <ord> <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>           <dbl>
##  1  0.23 Ideal E     SI2      61.5    55    326  3.95  3.98  2.43           1417.
##  2  0.21 Prem~ E     SI1      59.8    61    326  3.89  3.84  2.31           1552.
##  3  0.23 Good  E     VS1      56.9    65    327  4.05  4.07  2.31           1422.
##  4  0.29 Prem~ I     VS2      62.4    58    334  4.2   4.23  2.63           1152.
##  5  0.31 Good  J     SI2      63.3    58    335  4.34  4.35  2.75           1081.
##  6  0.24 Very~ J     VVS2     62.8    57    336  3.94  3.96  2.48           1400
##  7  0.24 Very~ I     VVS1     62.3    57    336  3.95  3.98  2.47           1400
##  8  0.26 Very~ H     SI1      61.9    55    337  4.07  4.11  2.53           1296.
##  9  0.22 Fair  E     VS2      65.1    61    337  3.87  3.78  2.49           1532.
## 10  0.23 Very~ H     VS1      59.4    61    338  4     4.05  2.39           1470.
## # i 53,930 more rows
## # i 1 more variable: price_carat <dbl>
```

```r
group_by(diamonds,cut=="Premium")
```

```
## # A tibble: 53,940 x 12
## # Groups:   cut == "Premium" [2]
##    carat cut   color clarity depth table price     x     y     z price_per_carat
##    <dbl> <ord> <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>           <dbl>
##  1  0.23 Ideal E     SI2      61.5    55    326  3.95  3.98  2.43           1417.
##  2  0.21 Prem~ E     SI1      59.8    61    326  3.89  3.84  2.31           1552.
##  3  0.23 Good  E     VS1      56.9    65    327  4.05  4.07  2.31           1422.
##  4  0.29 Prem~ I     VS2      62.4    58    334  4.2   4.23  2.63           1152.
##  5  0.31 Good  J     SI2      63.3    58    335  4.34  4.35  2.75           1081.
##  6  0.24 Very~ J     VVS2     62.8    57    336  3.94  3.96  2.48           1400
##  7  0.24 Very~ I     VVS1     62.3    57    336  3.95  3.98  2.47           1400
##  8  0.26 Very~ H     SI1      61.9    55    337  4.07  4.11  2.53           1296.
##  9  0.22 Fair  E     VS2      65.1    61   337  3.87  3.78  2.49           1532.
## 10  0.23 Very~ H     VS1      59.4    61   338  4     4.05  2.39           1470.
## # i 53,930 more rows
## # i 1 more variable: `cut == "Premium"` <lgl>
```

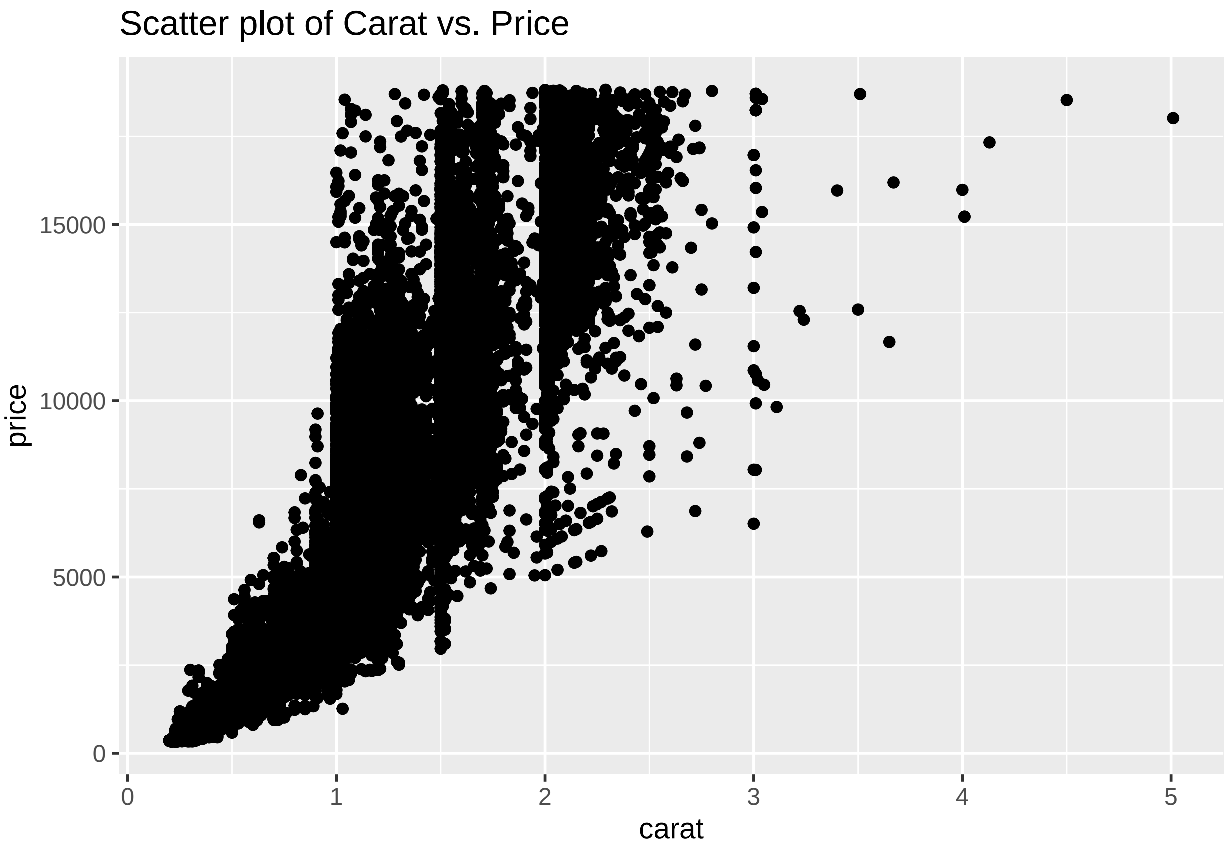
```
#summarise
diamonds %>%summarise(mean = mean(price),median =median(price),min=min(price),,max=max(price),sd=sd(pri
```

```
## # A tibble: 1 x 5
##    mean median   min   max    sd
##   <dbl>  <dbl> <int> <int> <dbl>
## 1 3933.   2401   326 18823 3989.
```

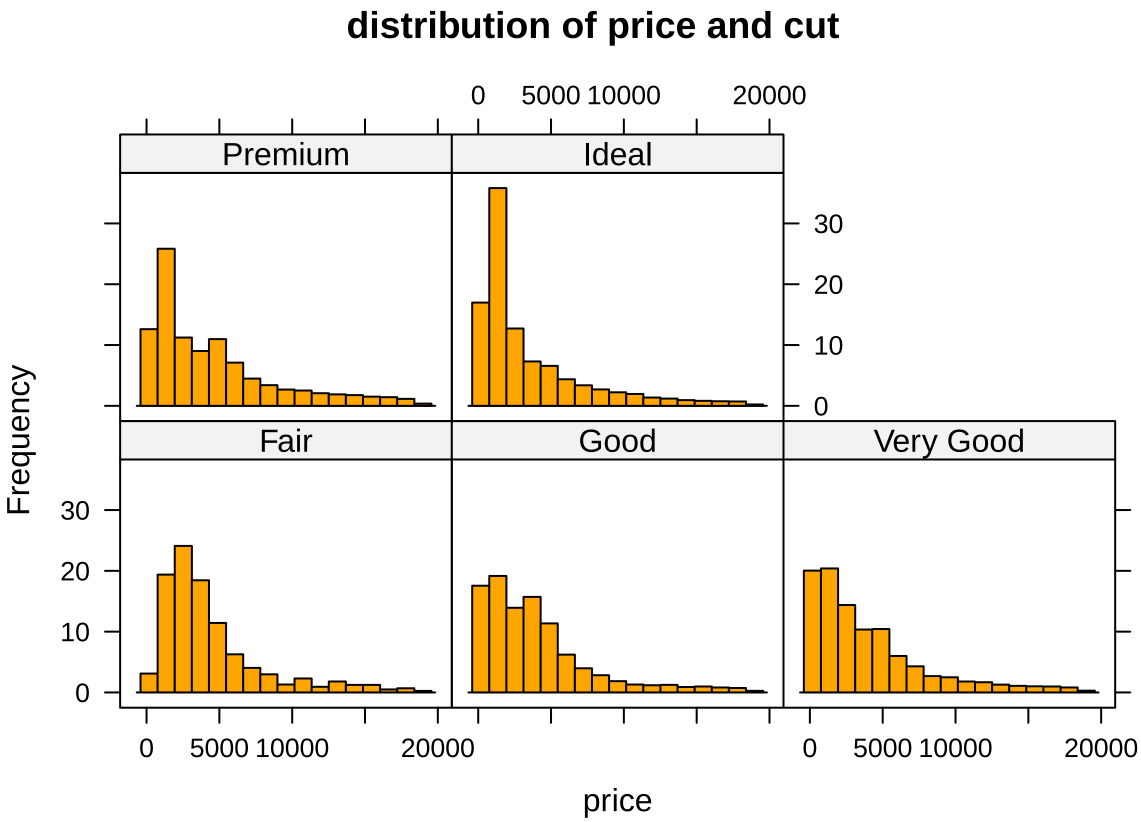
```
diamonds_summary <- diamonds %>%
  group_by(cut) %>%
  summarise(avg_price = mean(price))
diamonds_summary
```

```
## # A tibble: 5 x 2
##   cut       avg_price
##   <ord>         <dbl>
## 1 Fair          4359.
## 2 Good          3929.
## 3 Very Good     3982.
## 4 Premium       4584.
## 5 Ideal         3458.
```

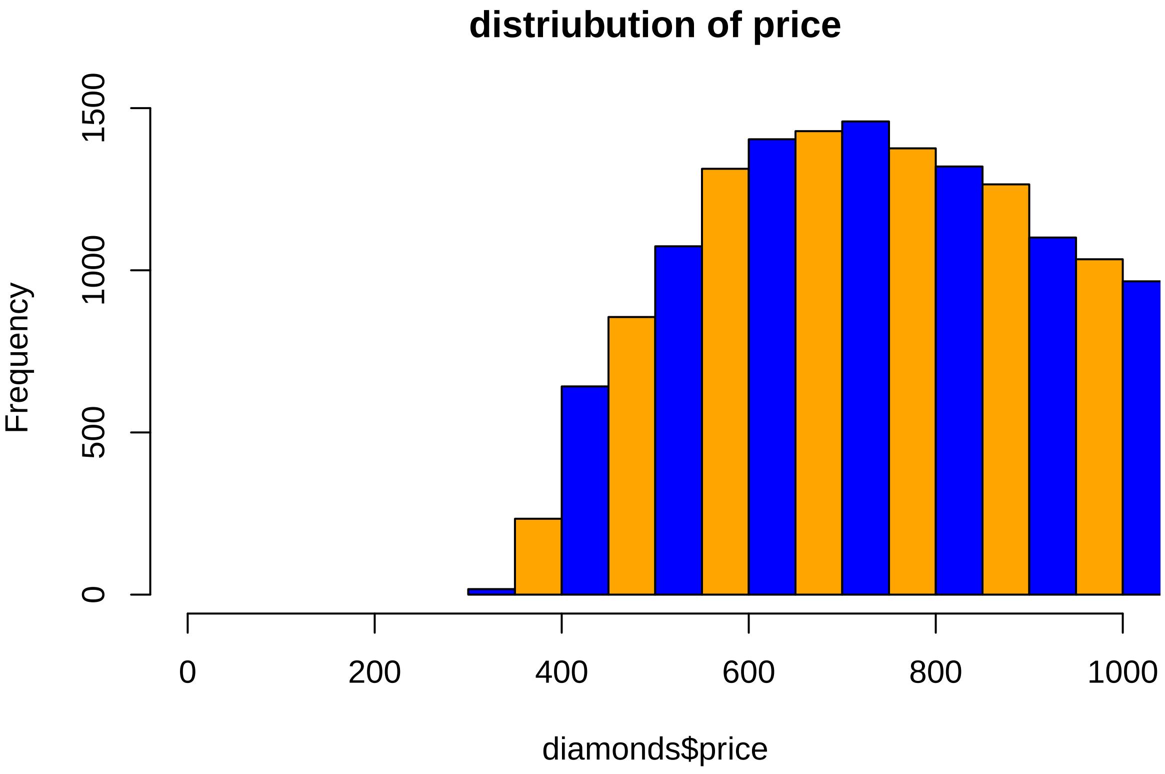
```
#scatter plot
ggplot(diamonds, aes(x = carat, y = price)) +
  geom_point() +
  labs(title = "Scatter plot of Carat vs. Price")
```



```
#Histogram
histogram(~price|cut,data=diamonds,col='orange',xlab = 'price',ylab = 'Frequency',main='distribution of
```

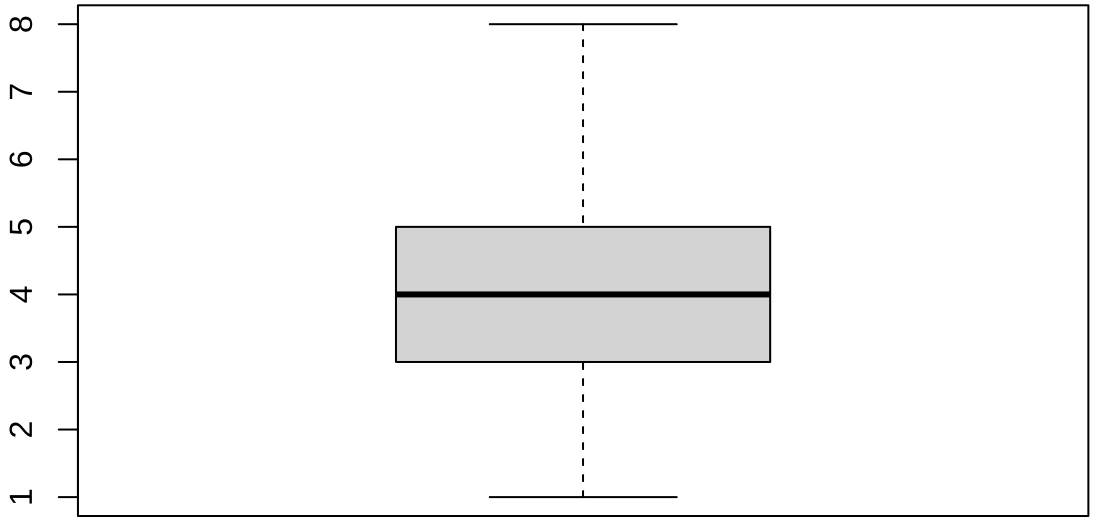


```
hist(diamonds$price ,breaks=500,xlim=c(0,1000),main="distriubution of price",col=c('blue','orange'))
```

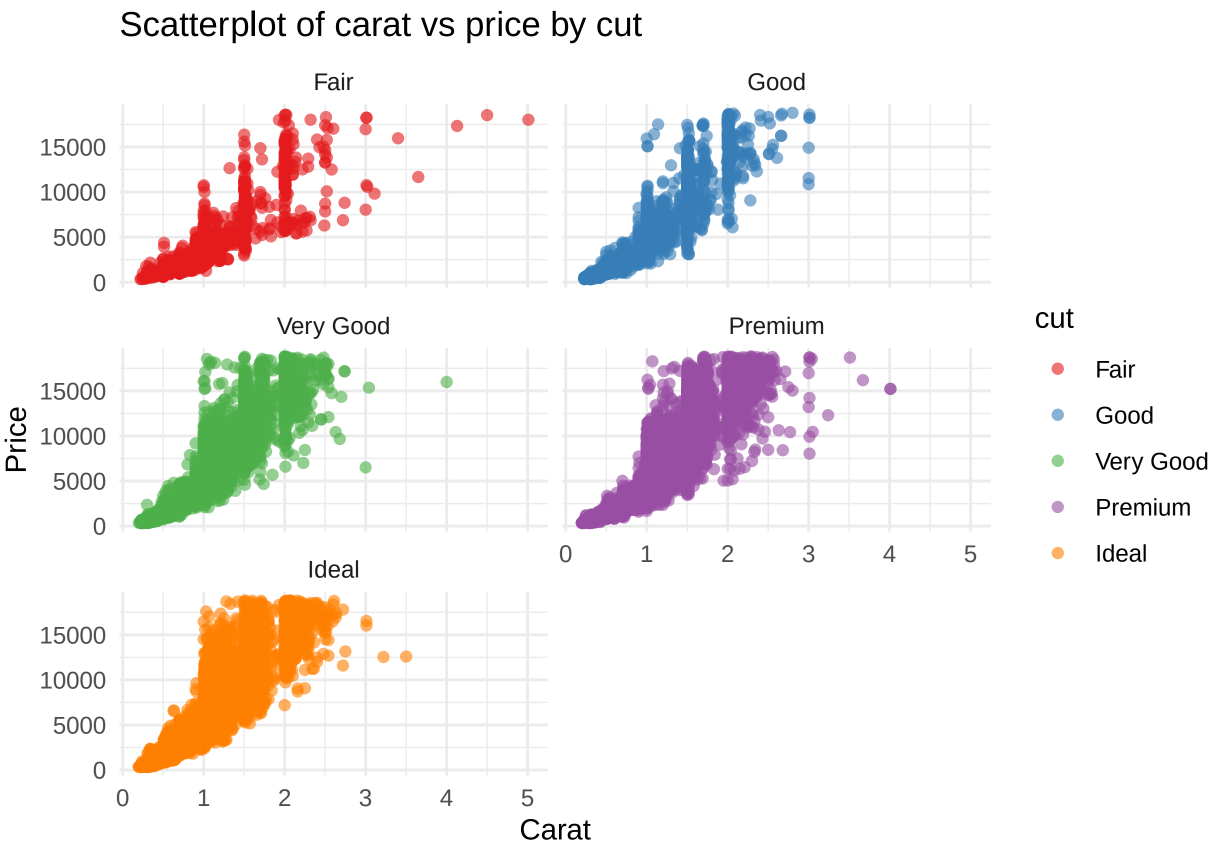


```
attach(diamonds)
#boxplot
```

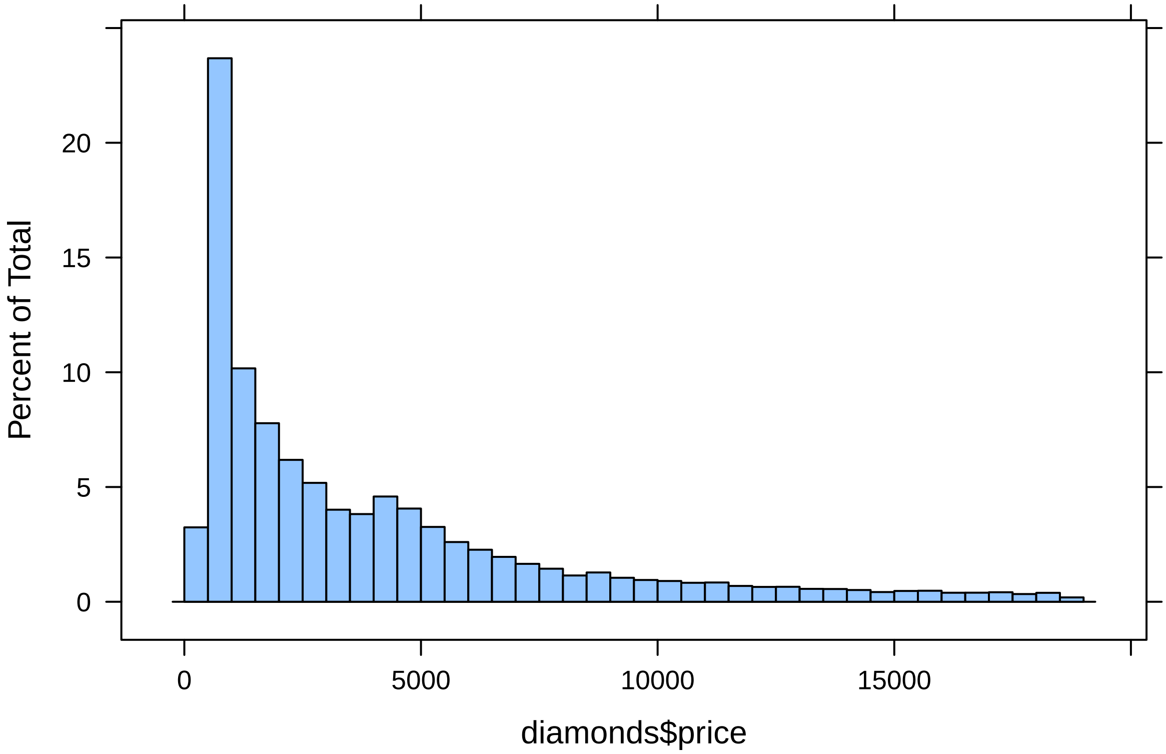
```
boxplot(diamonds$clarity)
```



```
#scatterplot
ggplot(diamonds,aes(x=carat,y=price,color=cut))+geom_point(alpha=0.6)+facet_wrap(~cut,ncol=2)+labs(x='C
```



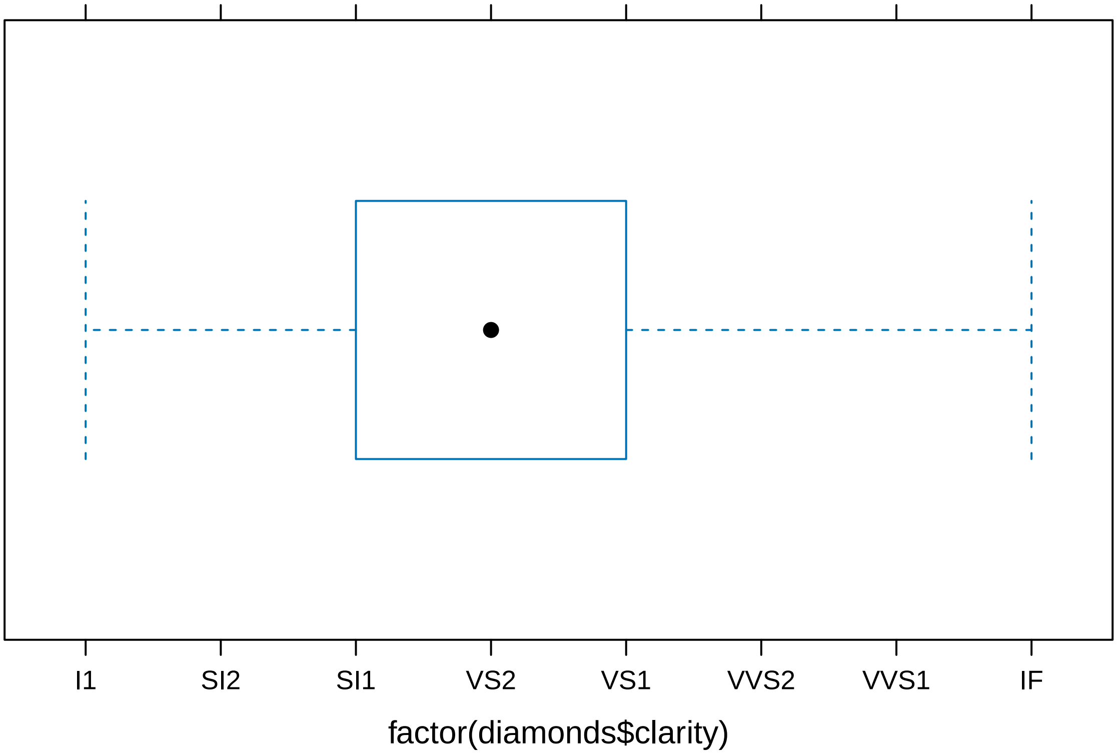
```
histogram(diamonds$price,breaks=50)
```



```
attach(diamonds)
```

```
## The following objects are masked from diamonds (pos = 3):
##
##     carat, clarity, color, cut, depth, price, price_per_carat, table,
##     x, y, z
```

```
#bwplot
bwplot(~factor(diamonds$clarity))
```



```
#Quantiles
claritys_Holder=diamonds[,4]
claritys_Holder
```

```
## # A tibble: 53,940 x 1
##    clarity
##    <ord>
##  1 SI2
##  2 SI1
##  3 VS1
##  4 VS2
##  5 SI2
##  6 VVS2
##  7 VVS1
##  8 SI1
##  9 VS2
## 10 VS1
## # i 53,930 more rows
```

```
Q1=quantile(diamonds$price,0.25)
Q3=quantile(diamonds$price,0.75)
IQR=Q3-Q1
QRL=Q1-1.5*IQR
QRU=Q3-1.5*IQR
data_no_outlier=subset(claritys_Holder,claritys_Holder>Q1&claritys_Holder<Q3)
length(data_no_outlier)
```

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