EDA_ProductAnalysis

```r
library(dplyr)
library(ggplot2)
library(ggplotify)

df <- read.csv("./../DataExtract/Data/processed.json_partial.csv")
df <- as.data.frame(df)
print(colnames(df))
```

```
##  [1] "infoType"     "title"        "link"         "stars"
##  [5] "totalRatings" "price"        "weight"       "rank"
##  [9] "description"  "brand"        "colors"
```

Data Correction

```r
df$totalRatingsNum <- as.numeric(gsub("," , "", as.character(df$totalRatings)))
df$rank <- as.numeric(gsub("," , "", as.character(df$rank)))
```

Correcting Brand Names

```r
table(df$brand)
```

```
##
##      Arctic         Fun     Garnier       Got2b        Hair        John
##           2           1           8           1           1           2
##     L'OrÃal    L'OrÃfal     L'oreal     L'Oreal     L'Oréal        Lime
##           2           1           3           4           6           5
##       Manic    MOFAJANG        Play     Pravana       Punky         RAW
##           2           1           1           1           1           1
##      Revlon Schwarzkopf      Silver     SoftSub       Vidal
##           7           7           1           1           1
```

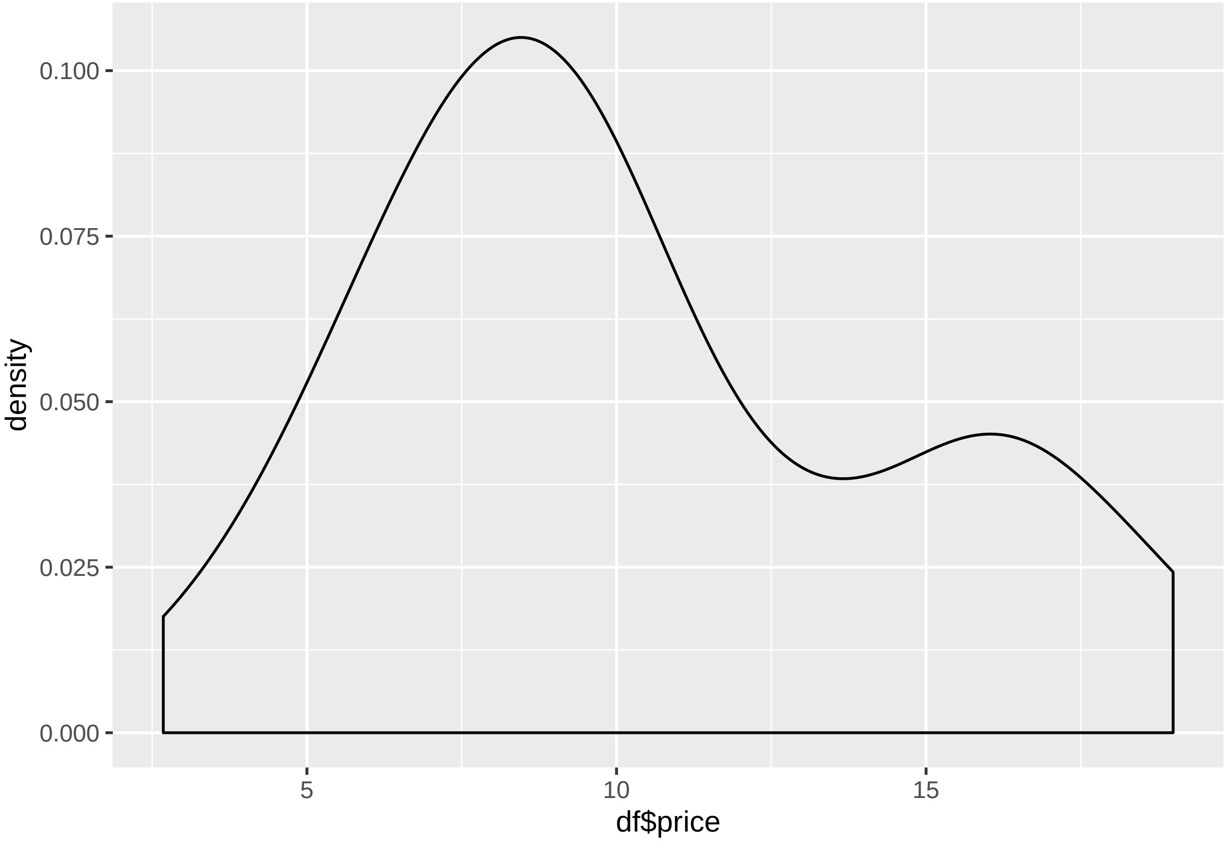
```r
# correcting l'oreal spellings
df[which(regexpr("L'" , df$brand) >= 0), ]$brand <- "L'Oreal"
```

Price Analysis

```r
## Estimated bands
df %>%
  group_by(brand) %>%
  summarise(meanprice = mean(price))
```

```
## # A tibble: 19 x 2
##    brand       meanprice
##    <fct>           <dbl>
##  1 Arctic          19.0
##  2 Fun             13.0
##  3 Garnier          8.50
##  4 Got2b            9.97
##  5 Hair             2.99
##  6 John            12.2
##  7 L'Oreal          9.20
##  8 Lime             16
##  9 Manic             8.06
## 10 MOFAJANG          6.93
## 11 Play             19.0
## 12 Pravana          10.5
## 13 Punky             7.46
## 14 RAW              13.0
## 15 Revlon            7.51
## 16 Schwarzkopf       9.97
## 17 Silver           13.8
## 18 SoftSub          16.0
## 19 Vidal             9.95
```

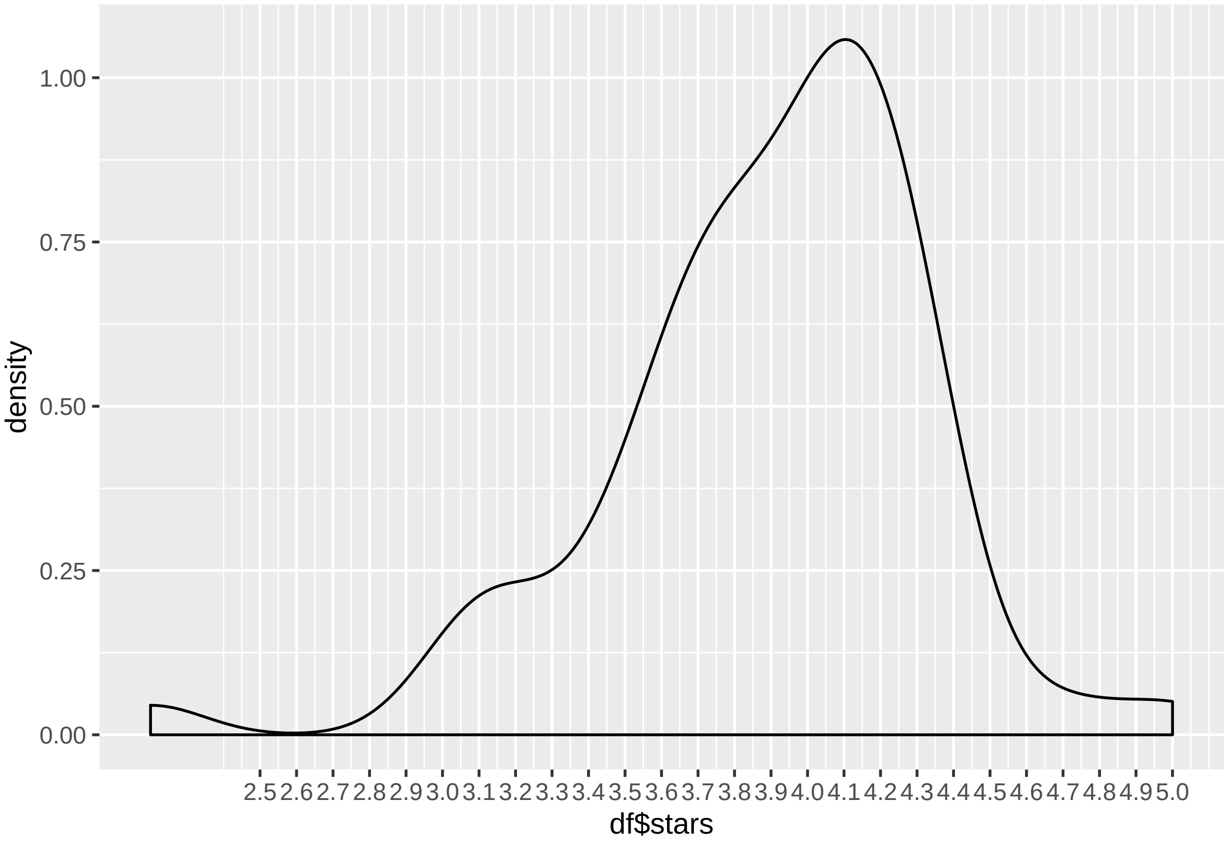
```
## Price
ggplot(data = df , aes(x = df$price)) + geom_density()
```



```
df$priceband <- case_when( df$price < 7.5 ~ "Low"
                          , df$price < 12.5 ~ "Medium"
                          ,TRUE ~ "High")
```

Ratings Analysis

```
## Ratings
ggplot(data = df , aes(x = df$stars)) +
  geom_density() +
  scale_x_continuous(breaks = seq(2.5,  5 , 0.1))
```



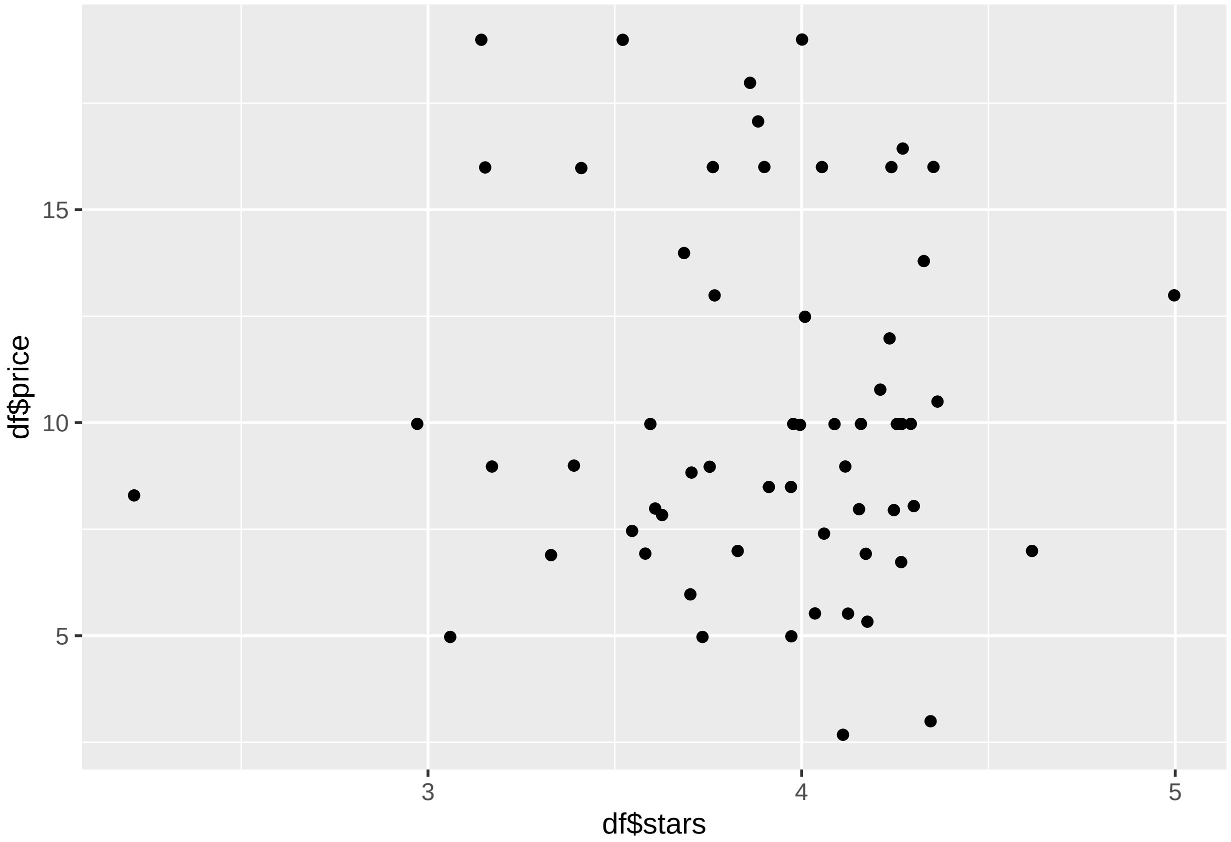
```
df$starsBand <- case_when( df$stars < 3.9 ~ "Poor"
                           ,df$stars < 4.3 ~ "Good"
                           ,TRUE ~ "Excellent")
```

Price Vs Ratings

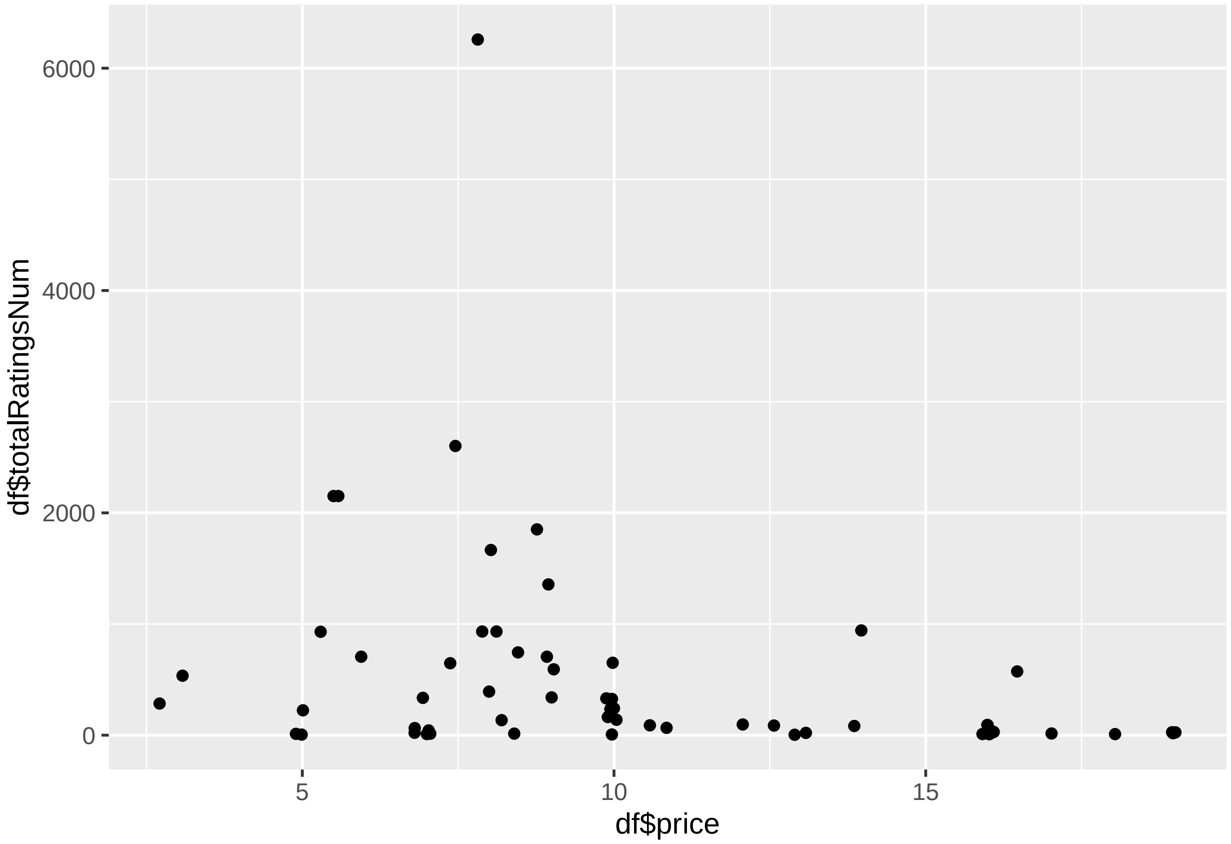
```
table(df$starsBand , df$priceband)
```

```
##
##              High Low Medium
##   Excellent     3   3      5
##   Good          5   7     12
##   Poor          9   7      9
```

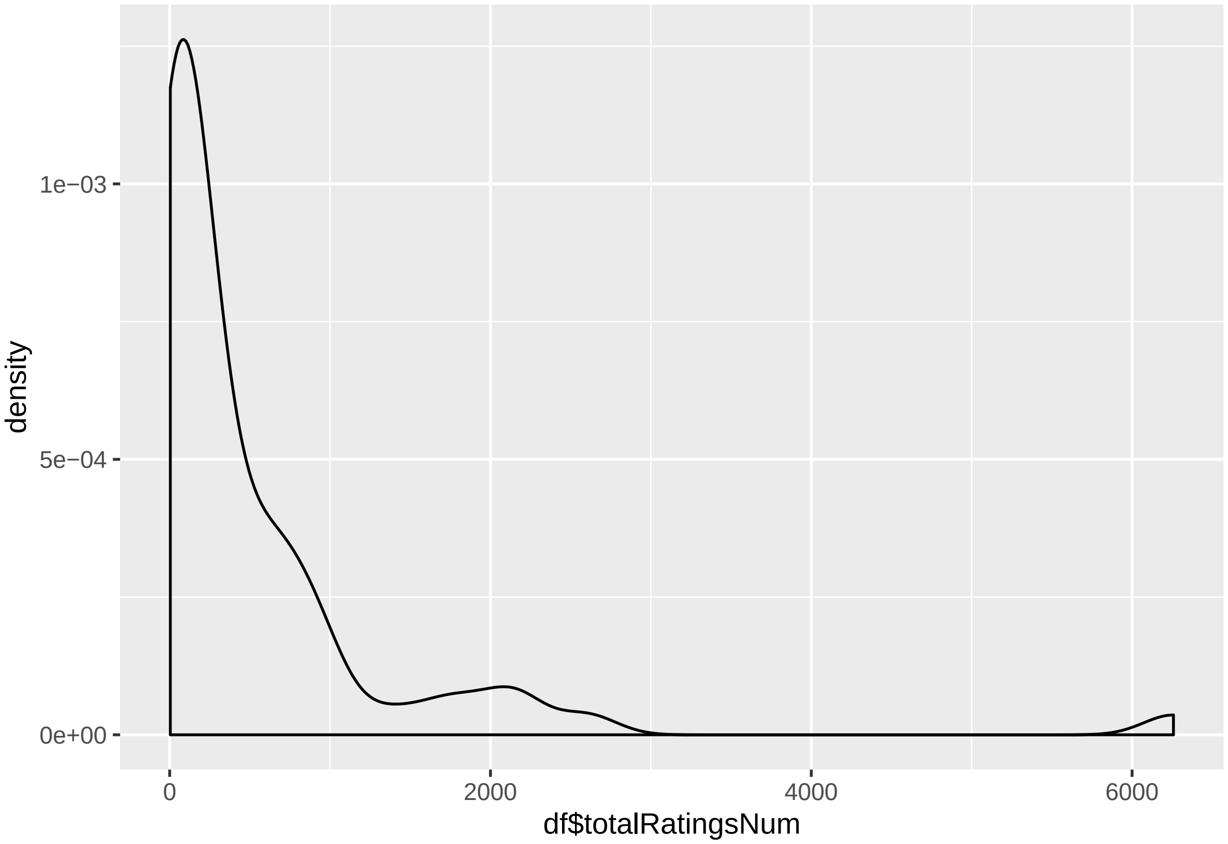
```
ggplot(data = df, aes(x = df$stars , y = df$price)) + geom_jitter(width = 0.1)
```



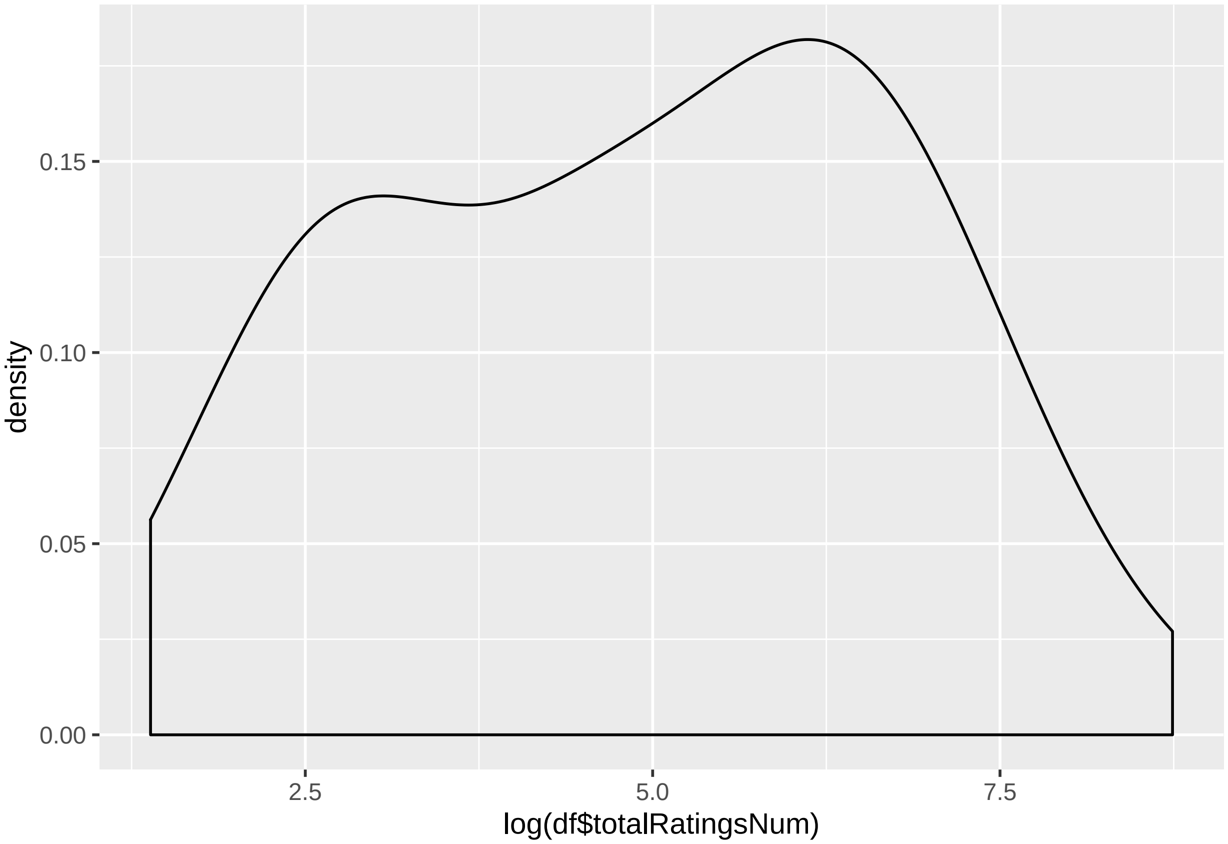
```
ggplot(data = df, aes(x = df$price , y = df$totalRatingsNum)) + geom_jitter(width = 0.1)
```



```
ggplot(data = df, aes(x = df$totalRatingsNum)) + geom_density()
```

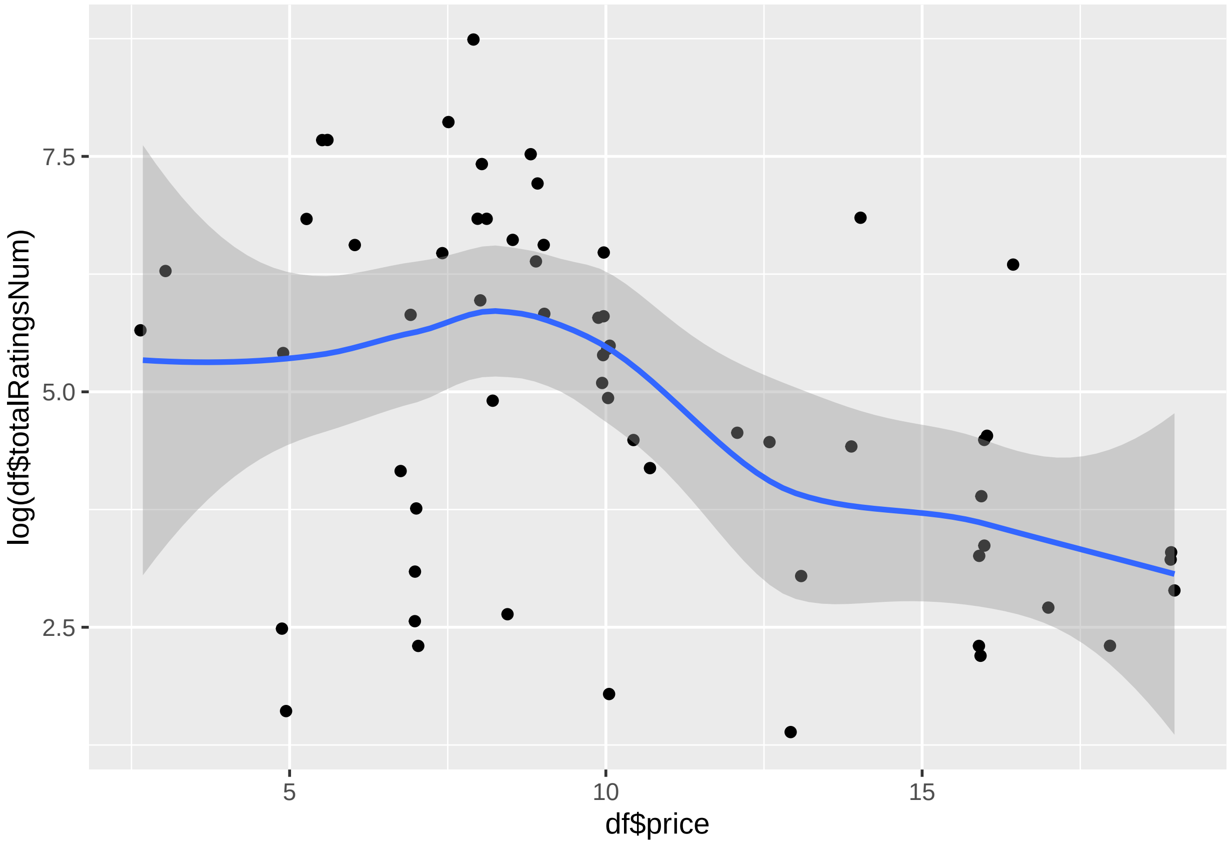


```r
ggplot(data = df, aes(x = log(df$totalRatingsNum))) + geom_density()
```



```
ggplot(data = df, aes(x = df$price , y = log(df$totalRatingsNum))) + geom_jitter(width = 0.1) + geom_smo
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



There is a decreasing trend in the number of ratings as the price increase, so fewer people buy products with higher price.

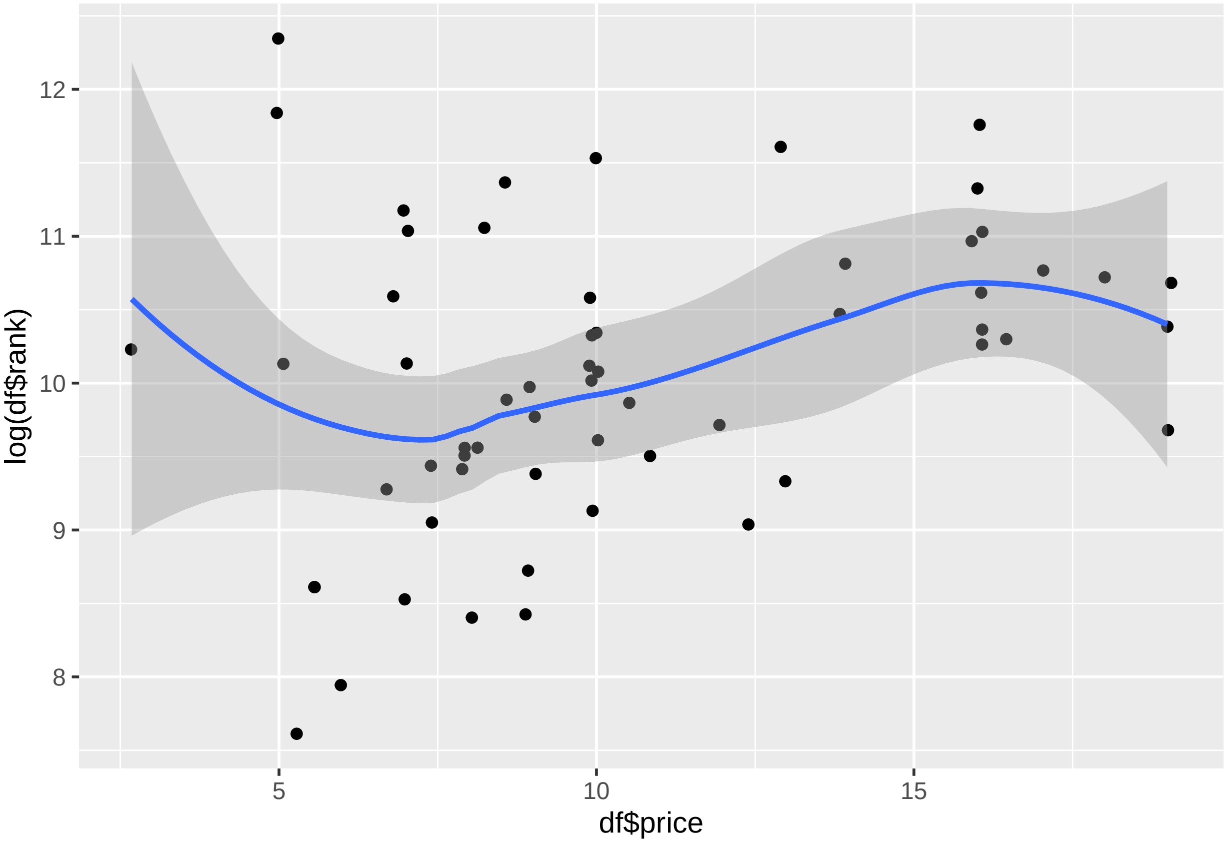
Amazon Rank Vs Price , Rating , Total Ratings

```
ggplot(data = df , aes(x =df$price , y=log(df$rank) )) + geom_jitter(width = 0.1) + geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

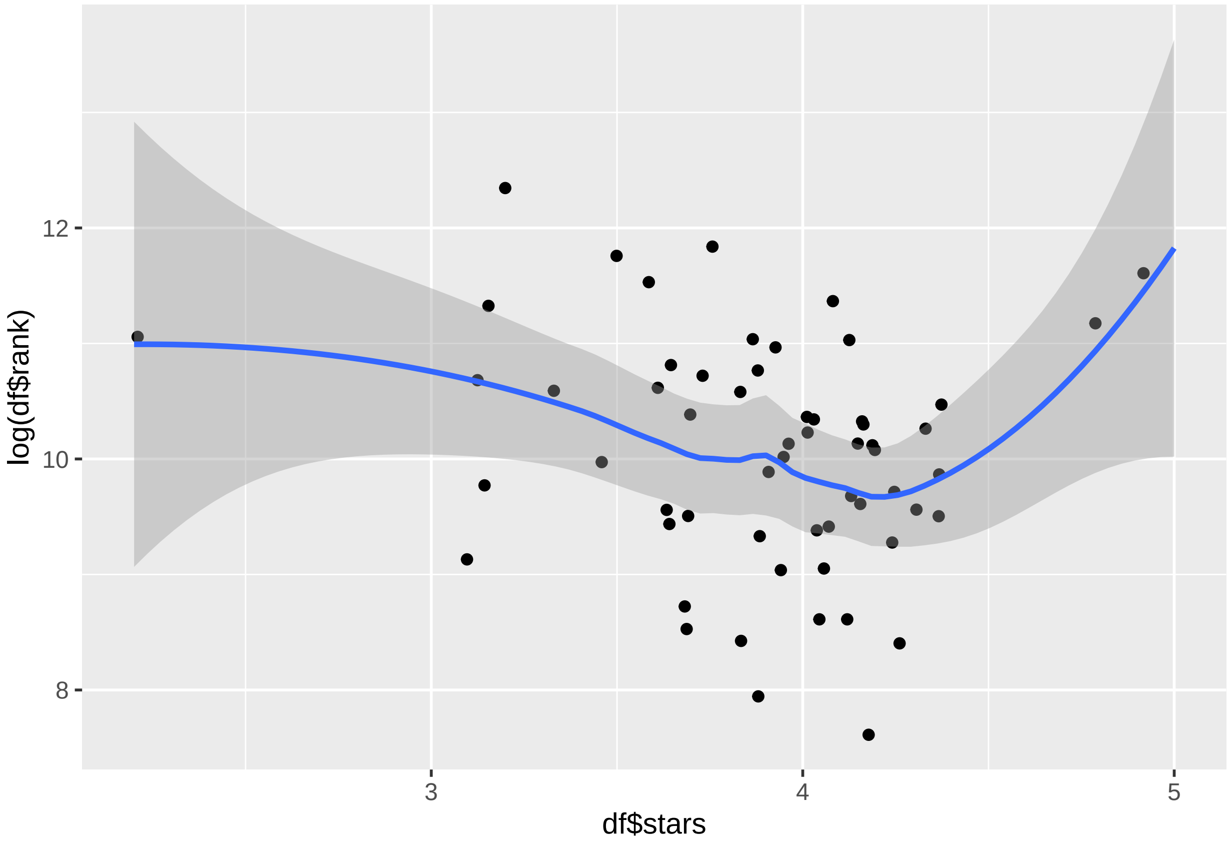
```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
ggplot(data = df , aes(x =df$stars , y=log(df$rank) )) + geom_jitter(width = 0.1) + geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

## Warning: Removed 1 rows containing non-finite values (stat_smooth).

## Warning: Removed 1 rows containing missing values (geom_point).
```

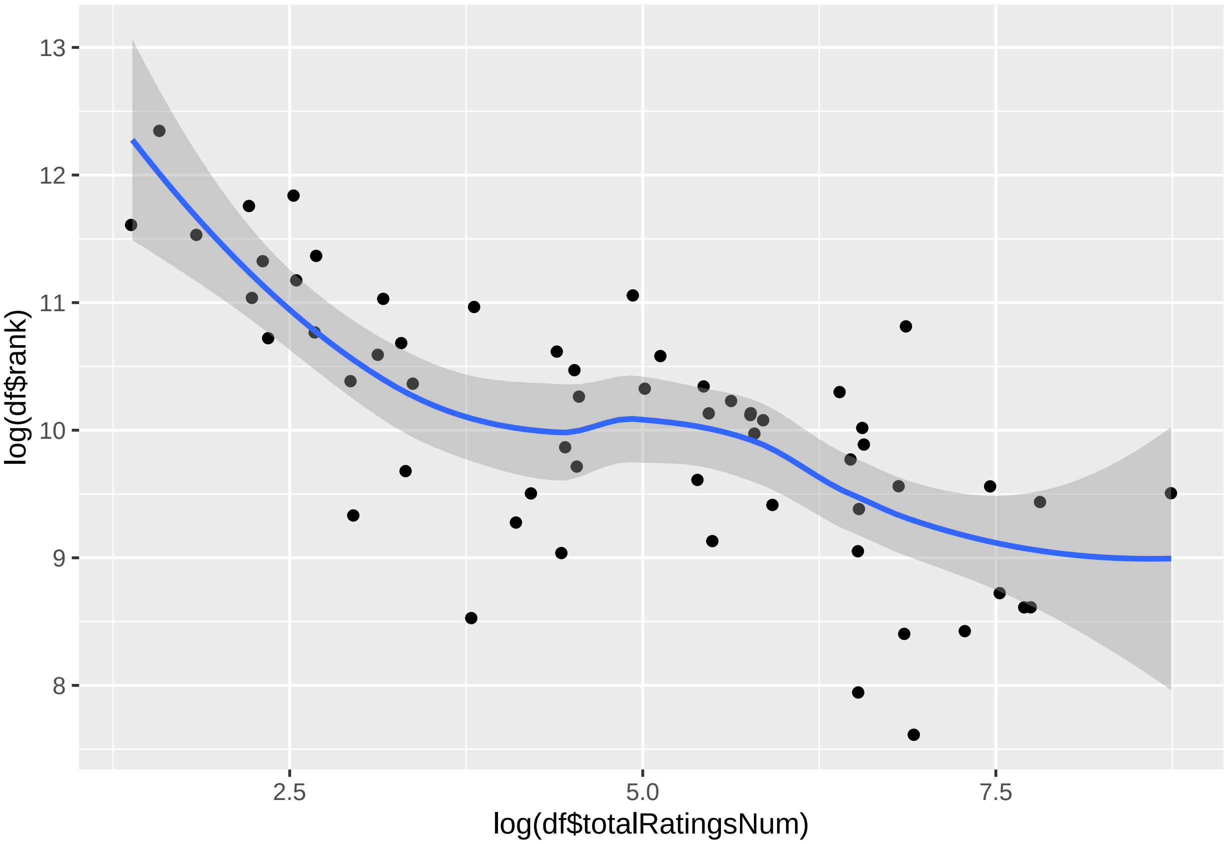


```
ggplot(data = df , aes(x =log(df$totalRatingsNum) , y=log(df$rank) )) + geom_jitter(width = 0.1) + geom_
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



As price increases the products are not as popular. Rank decreases as the number of people rating the product increases.

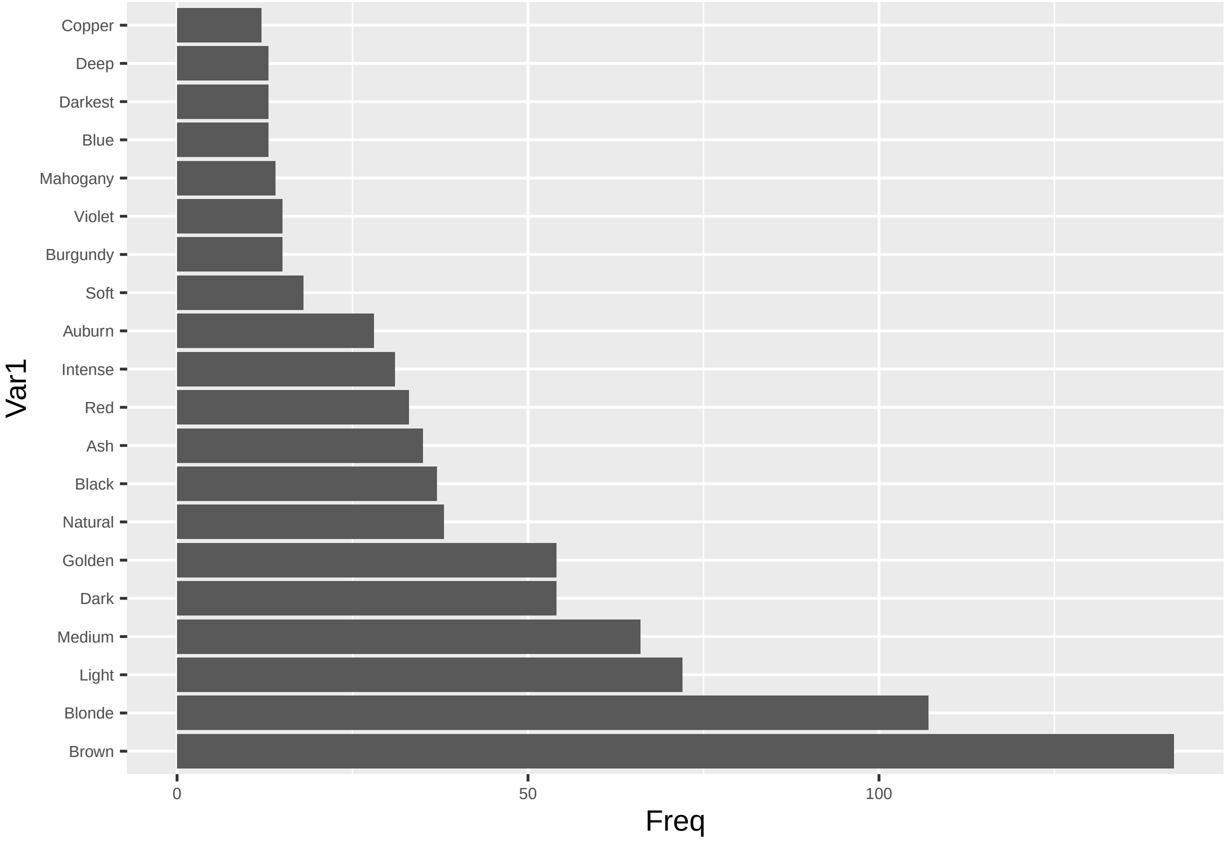
Color Analysis

- Color Distribution

```
colorDf <- read.csv("./../DataExtract/Data/processed.json_color.csv")
colorDf$Color <- as.character(colorDf$Color)
splitColors <- strsplit(colorDf$Color , split = ' ')

combinedColors <- Reduce(f = function(a,b) c(a , b) , x = splitColors , init=list())
freqColors <- as.data.frame(table(unlist(splitColors)))
freqColors$Var1 <- as.character(freqColors$Var1)

sortedFreqColors <- freqColors %>% arrange(desc(Freq))
sortedFreqColors$Var1 <- factor(as.character(sortedFreqColors$Var1), levels=sortedFreqColors$Var1 )

head(sortedFreqColors)
```

```
##      Var1 Freq
## 1  Brown  142
## 2 Blonde  107
## 3  Light   72
## 4 Medium   66
## 5   Dark   54
## 6 Golden   54
```

```
ggplot(data= sortedFreqColors[1:20,] , aes(x = Var1 , y=Freq)) +
  theme(axis.text=element_text(size=6)) +
  geom_bar(stat='identity') +
  coord_flip()
```



- Summarising Various Colors

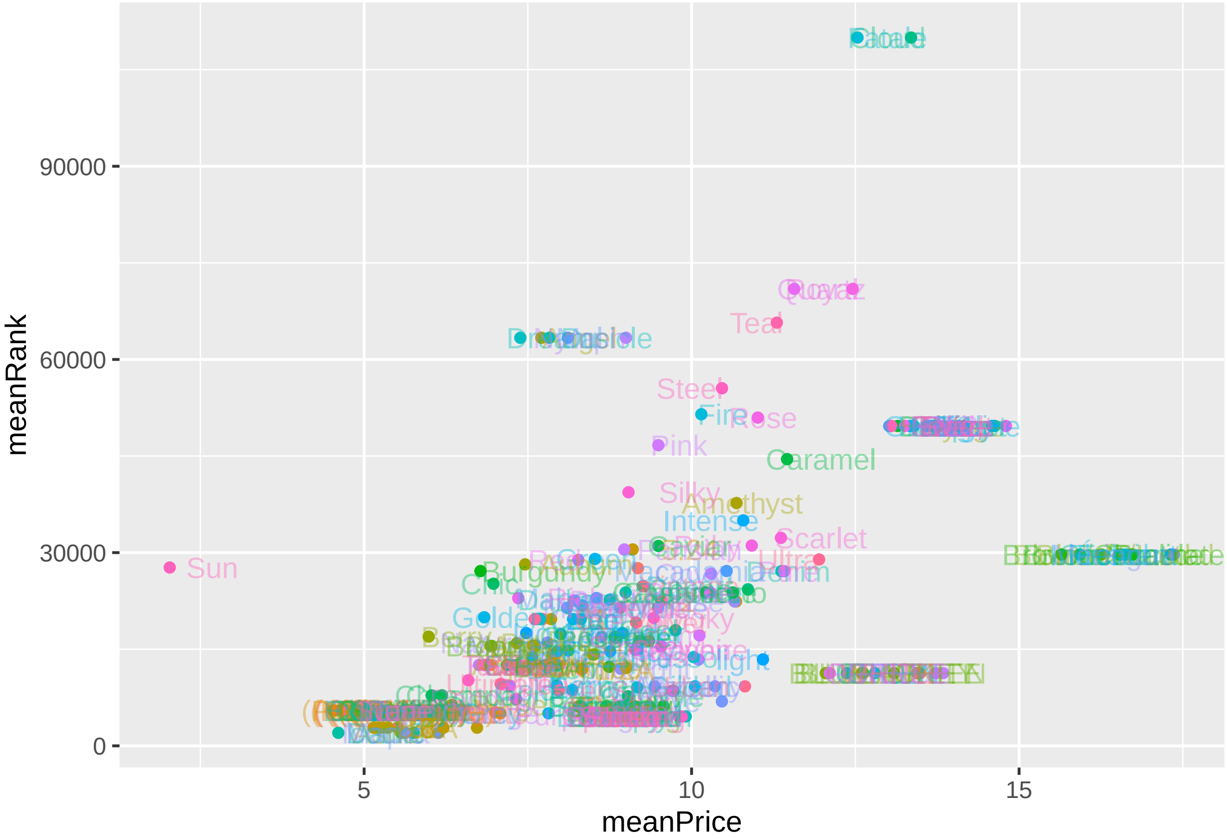
```
#combined = data.frame(price = rep(colorDf[,c("price", "totalRatings")], sapply(splitColors, length)),

repd <- colorDf[rep(row.names(colorDf) ,sapply(splitColors, length) ) , c("stars" ,"price" , "totalRati
repd$color <- unlist(splitColors)

repd$totalRatingsNum <- as.numeric(gsub("," , "", as.character(repd$totalRatings)))
repd$rank <- as.numeric(gsub("," , "", as.character(repd$rank)))
repd[which(regexpr("L'" , repd$brand) >= 0), ]$brand <- "L'Oreal"

head(repd)
```

```
##     stars price totalRatings rank   brand   color totalRatingsNum
## 1     3.8  8.97        1,356 4562 L'Oreal    Pure            1356
## 1.1   3.8  8.97        1,356 4562 L'Oreal Diamond            1356
## 2     3.8  8.97        1,356 4562 L'Oreal     Icy            1356
## 2.1   3.8  8.97        1,356 4562 L'Oreal  Blonde            1356
## 2.2   3.8  8.97        1,356 4562 L'Oreal   Ultra            1356
## 2.3   3.8  8.97        1,356 4562 L'Oreal    Cool            1356
```

```
meanByColor <- repd %>%
  group_by(color) %>%
  summarise(meanStar=mean(stars), meanPrice = mean(price) , meanNumRatings = mean(totalRatingsNum) , mea
  arrange(desc(meanNumRatings))

head(meanByColor)
```

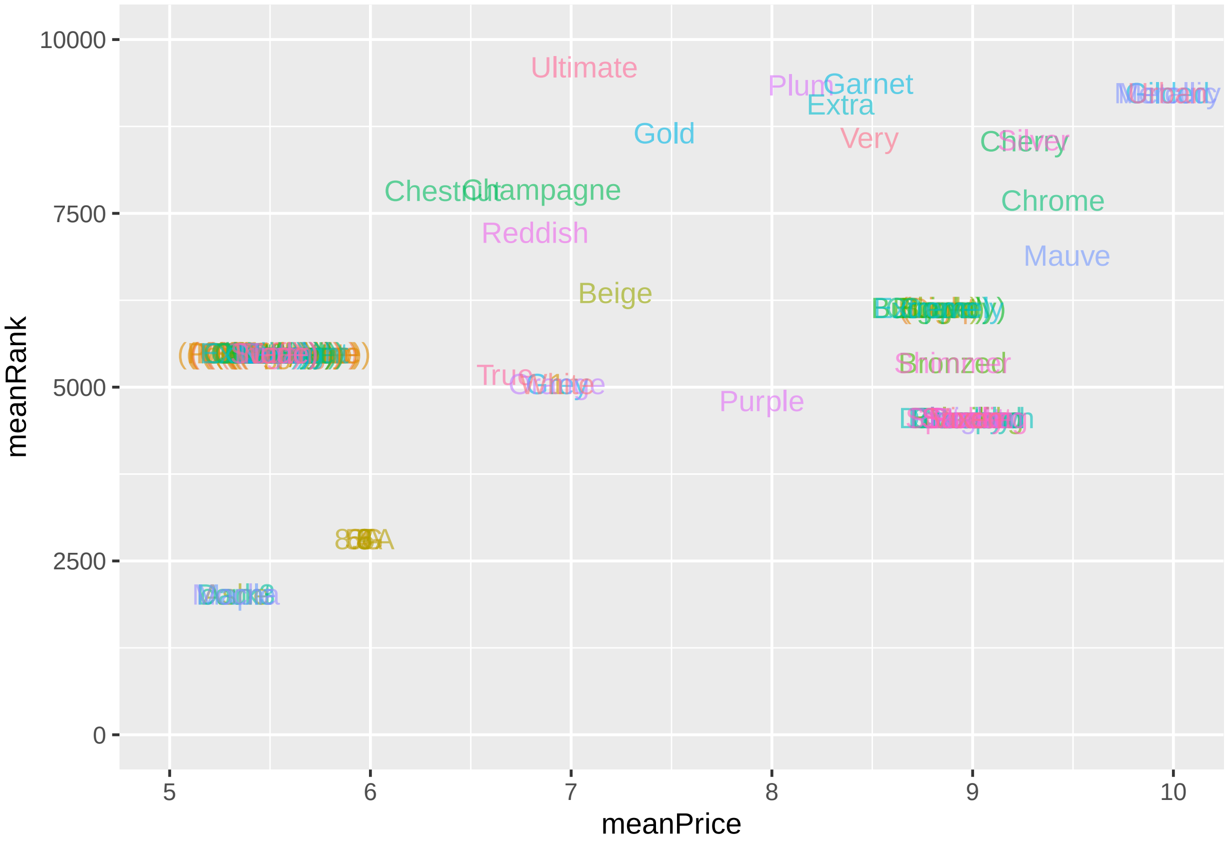
```
## # A tibble: 6 x 6
##   color    meanStar meanPrice meanNumRatings meanRank totalCount
##   <chr>       <dbl>     <dbl>          <dbl>    <dbl>      <int>
## 1 Adjustor      3.6      7.46           2602    12549          1
## 2 Candy         3.6      7.46           2602    12549          1
## 3 Flame         3.6      7.46           2602    12549          1
## 4 On            3.6      7.46           2602    12549          1
## 5 PastelFX      3.6      7.46           2602    12549          1
## 6 Shade         3.6      7.46           2602    12549          1
```

```
ggplot(data = meanByColor , aes(x = meanPrice , y= meanRank , label=color , color = color)) +
    geom_jitter(width=1 , show.legend = FALSE) +
    geom_text(alpha = 0.4, show.legend = FALSE)
```



```
ggplot(data = meanByColor , aes(x = meanPrice , y= meanRank , label=color , color = color)) +
    geom_text(alpha = 0.6, show.legend = FALSE) +
  scale_x_continuous(limits = c(5,10)) +
  scale_y_continuous(limits = c(0,10000))
```

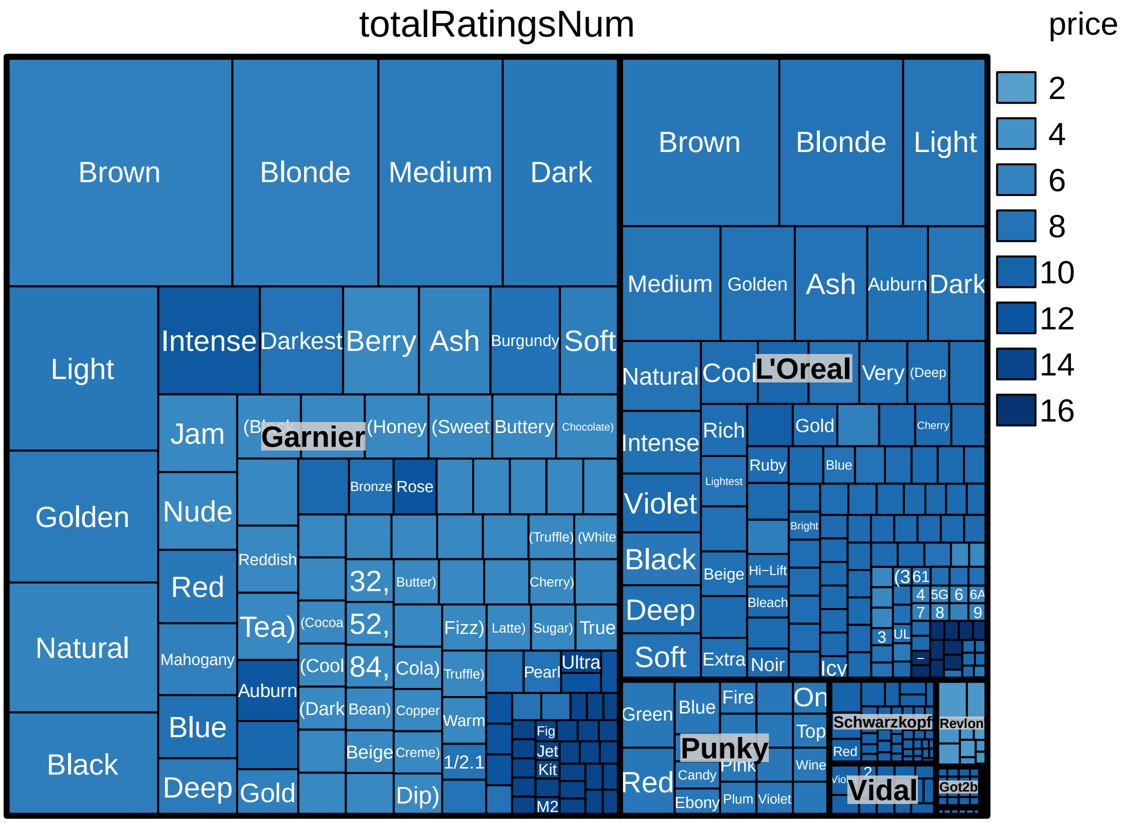
```
## Warning: Removed 150 rows containing missing values (geom_text).
```



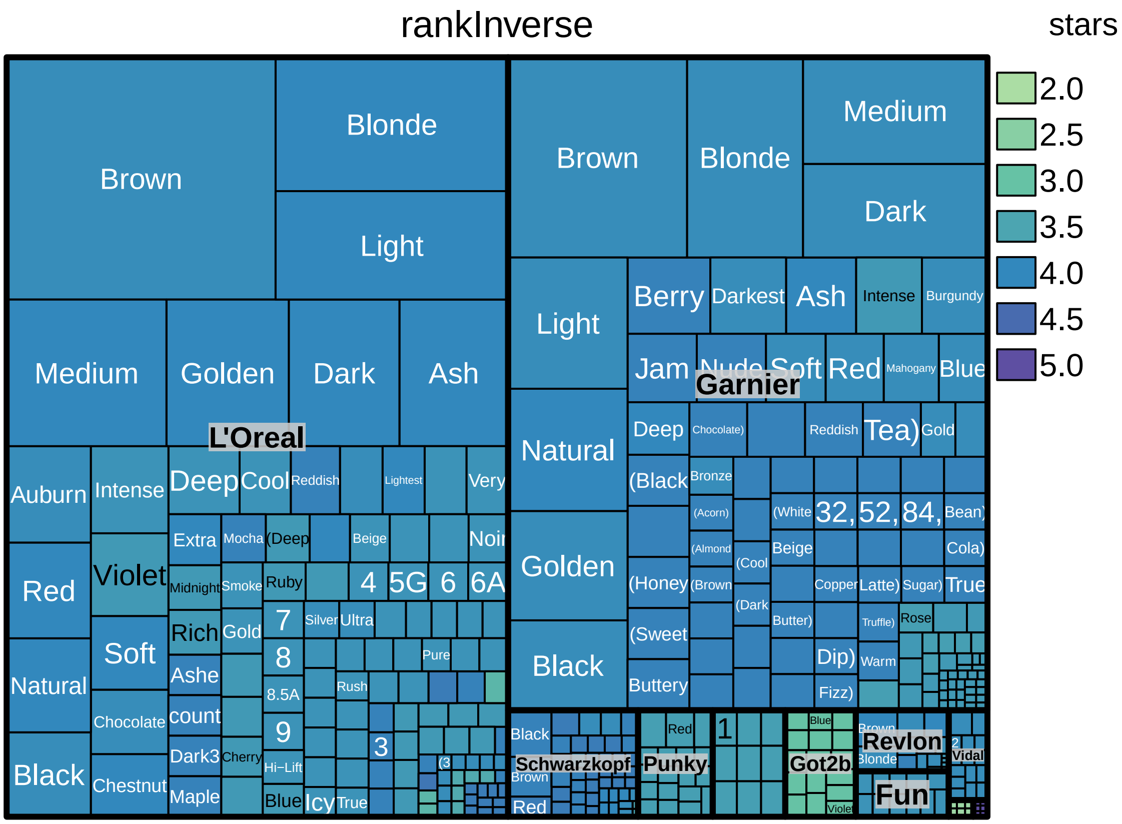
```
library(treemap)

repd %>% group_by(brand ,color) %>% summarise(mprice = mean(price))
```

```
## # A tibble: 382 x 3
## # Groups:   brand [11]
##    brand color      mprice
##    <fct> <chr>       <dbl>
##  1 Fun   BLACK        13.0
##  2 Fun   BLUE         13.0
##  3 Fun   BLUE+GREEN   13.0
##  4 Fun   BLUE+GREY    13.0
##  5 Fun   BLUE+WHITE   13.0
##  6 Fun   GREEN        13.0
##  7 Fun   GREY         13.0
##  8 Fun   ORANGE       13.0
##  9 Fun   PURPLE       13.0
## 10 Fun   RED          13.0
## # ... with 372 more rows
```

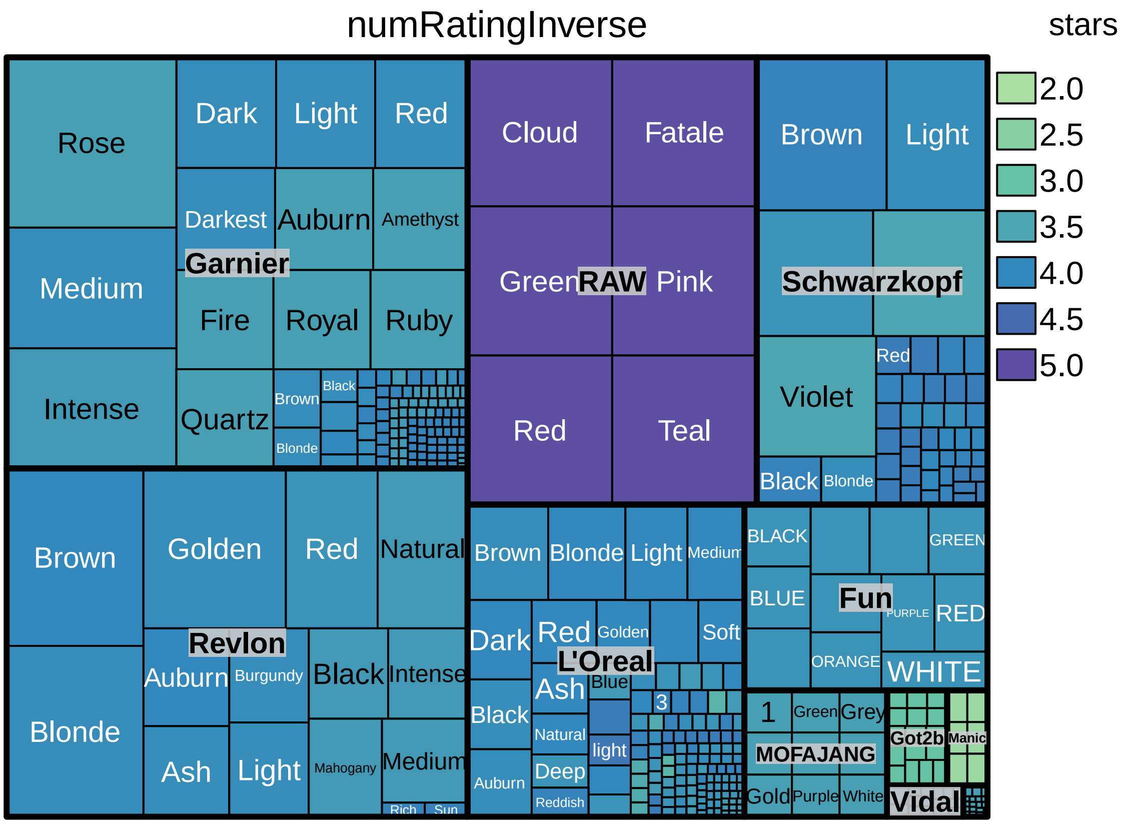
```
# Garnier is most popular. Popluarity proportional to total Number of ratings
treemap(repd,
        vSize = "totalRatingsNum",
        vColor = "price",
        index = c("brand" , "color")
        ,palette = "Blues"
        , fun.aggregate = "mean"
        , position.legend = "right"
        , type="value")
```



```
# Brown and Blonde are Ranked by amazon at the top in most brands
# Got2b is struggling as people don't really like their products
repd$rankInverse = 1/repd$rank
treemap(repd,
        vSize = "rankInverse",
        vColor = "stars",
        index = c("brand" , "color")
        ,palette = "Spectral"
        , fun.aggregate = "mean"
        , position.legend = "right"
        , type="value")
```



```r
# RAW is being used by fewer people but they seem to like it
# It has very different palette on offer compared to others. Very stark colors.
# Most colors in this chart are ones that are less common.
repd$numRatingInverse = 1/repd$totalRatingsNum
treemap(repd,
        vSize = "numRatingInverse",
        vColor = "stars",
        index = c("brand" , "color")
        ,palette = "Spectral"
        , fun.aggregate = "mean"
        , position.legend = "right"
        , type="value")
```


numRatingInverse
stars
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Garnier
Rose
Medium
Intense
Dark
Light
Red
Darkest
Auburn
Amethyst
Fire
Royal
Ruby
Quartz
Brown
Black
Blonde
RAW
Cloud
Fatale
Green
Pink
Red
Teal
Schwarzkopf
Brown
Light
Violet
Red
Black
Blonde
Revlon
Brown
Golden
Red
Natural
Blonde
Auburn
Burgundy
Black
Intense
Ash
Light
Mahogany
Medium
Rich
Sun
L'Oreal
Brown
Blonde
Light
Medium
Dark
Red
Golden
Soft
Ash
Blue
3
Black
Natural
light
Auburn
Deep
Reddish
Fun
BLACK
GREEN
BLUE
PURPLE
RED
ORANGE
WHITE
MOFAJANG
1
Green
Grey
Gold
Purple
White
Got2b
Manic
Vidal