

# Data Visualization

- Data detail : [https://bookdown.org/yih\\_huynh/Guide-to-R-Book/diamonds.html](https://bookdown.org/yih_huynh/Guide-to-R-Book/diamonds.html)

```
# install.packages( c("ggthemes", "patchwork", "lubridate", "readr") )
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.2 --
## v ggplot2 3.4.0     v purrr   1.0.1
## v tibble   3.1.8     v dplyr    1.1.0
## v tidyr    1.3.0     v stringr  1.5.0
## v readr    2.1.3     vforcats  1.0.0
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()

library(patchwork)
library(lubridate)

##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

library(ggthemes)
library(ggplot2)
library(readr)
```

## Data set detail

```
diamonds_detail <- read_csv("detail.csv")

## Rows: 10 Columns: 3
## -- Column specification -----
## Delimiter: ","
## chr (3): Variable, Description, Values
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

print(diamonds_detail)

## # A tibble: 10 x 3
##   Variable Description          Values
##   <chr>    <chr>
## 1 price    price in US dollars $326-$18,823
```

```

## 2 carat      weight of the diamond          0.2-5.01
## 3 cut        quality of the cut           Fair, Good, Very G-
## 4 color      diamond color                J (worst) to D (be-
## 5 clarity    measurement of how clear the diamond is I1 (worst), SI2, S-
## 6 x          length in mm                 0-10.74
## 7 y          width in mm                  0-58.9
## 8 z          depth in mm                  0-31.8
## 9 depth     total depth percentage       43-79
## 10 table    width of top of diamond relative to widest point 43-95

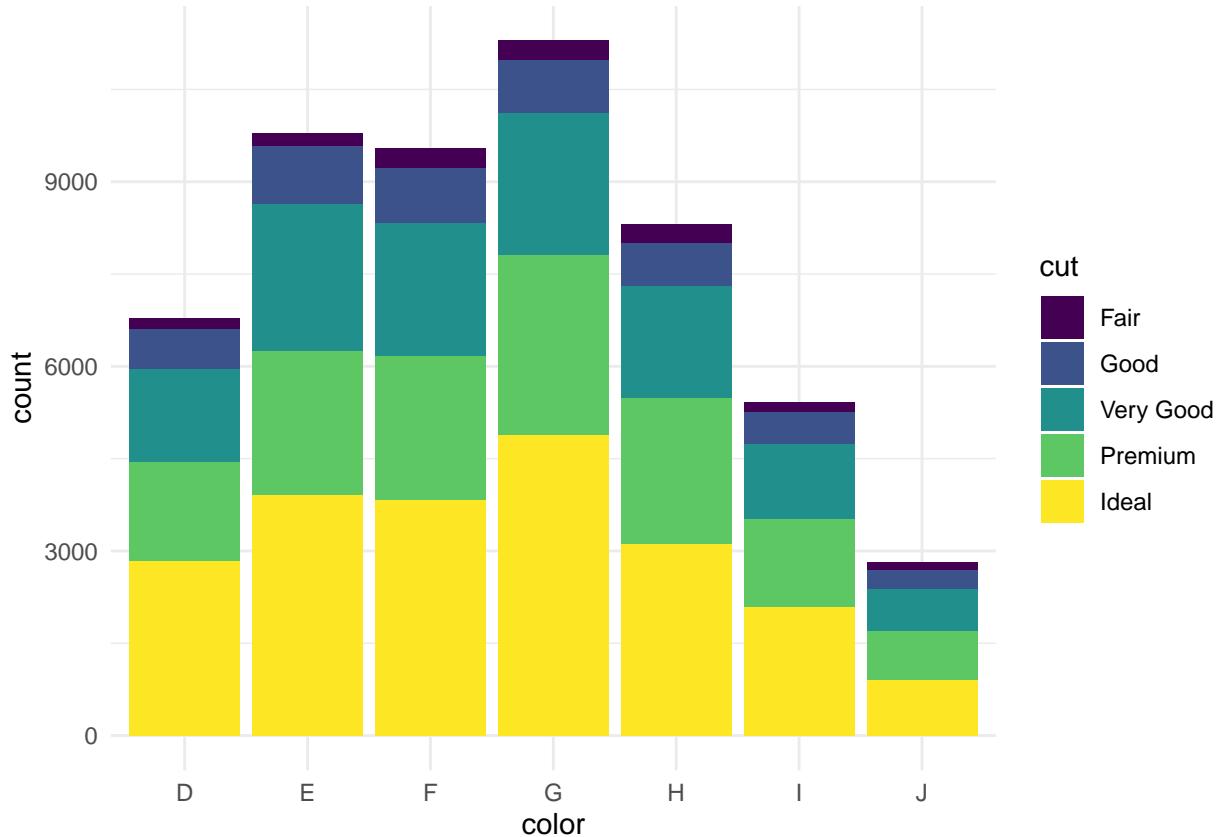
```

## The color of each quality diamonds

```

ggplot(diamonds, aes(color, fill= cut)) +
  geom_bar() +
  theme_minimal()

```

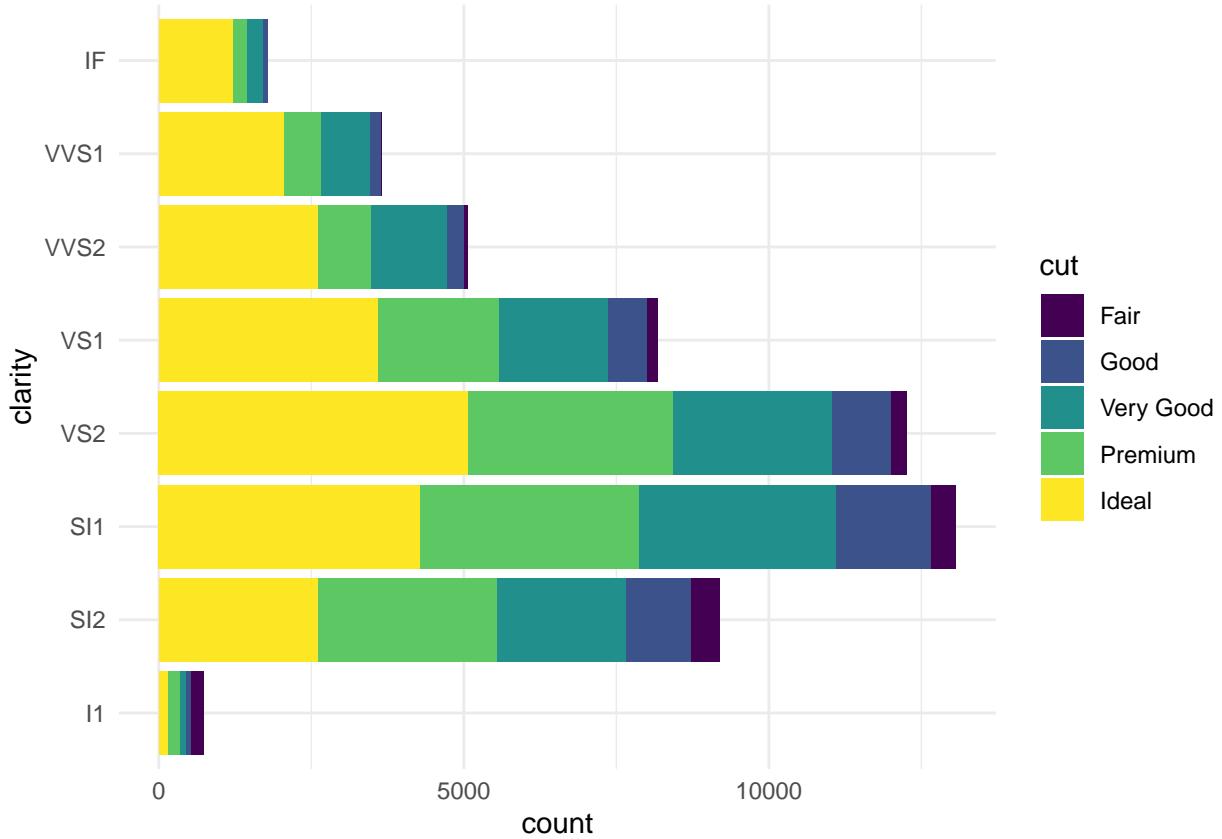


## The clarity of each quality diamonds

```

ggplot(diamonds, aes(y=clarity, fill = cut)) +
  geom_bar() +
  theme_minimal()

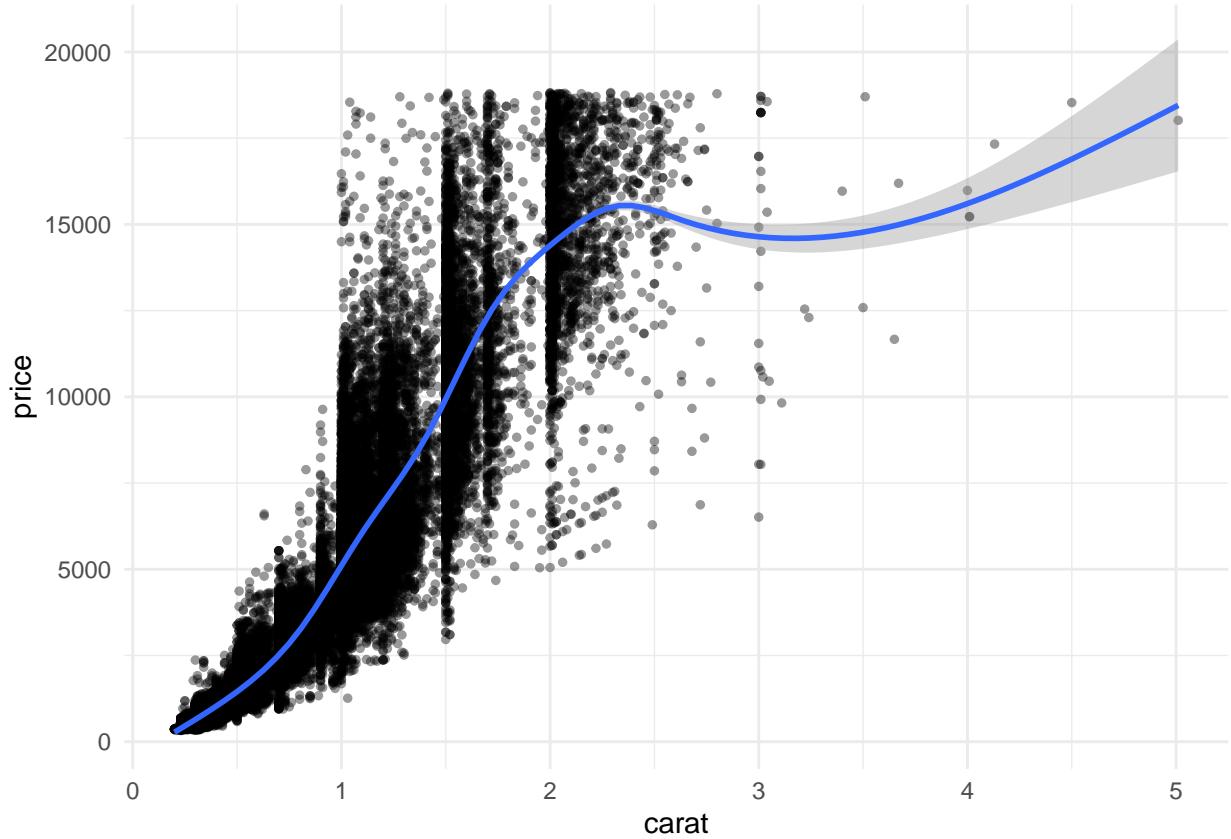
```



## Relationship between carat and price

```
ggplot(diamonds, aes(carat, price))+
  geom_point(size=1, alpha=0.4) +
  geom_smooth() +
  theme_minimal()

## 'geom_smooth()' using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```



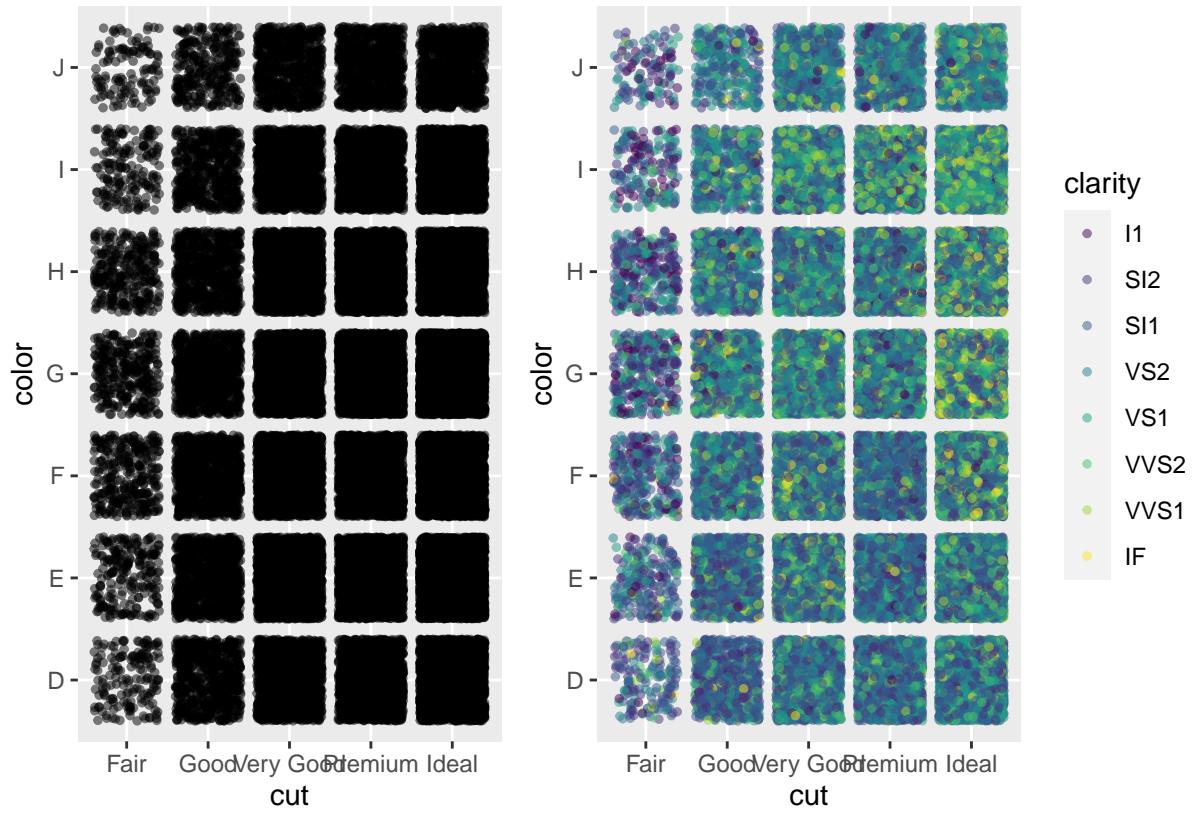
## Relationship between cut and color

- data discrete x discrete

```
dd1 = # cut >> color
ggplot(diamonds, aes(cut , color)) +
  geom_jitter(size=1, alpha=0.5)

dd2 = # cut >> color // clarity
ggplot(diamonds, aes(cut , color, color=clarity)) +
  geom_jitter(size=1, alpha=0.5)

dd1 + dd2
```



## Relationship between

(unit price : million) - cut and price - color and price separated by cut - clarity and price separated by cut

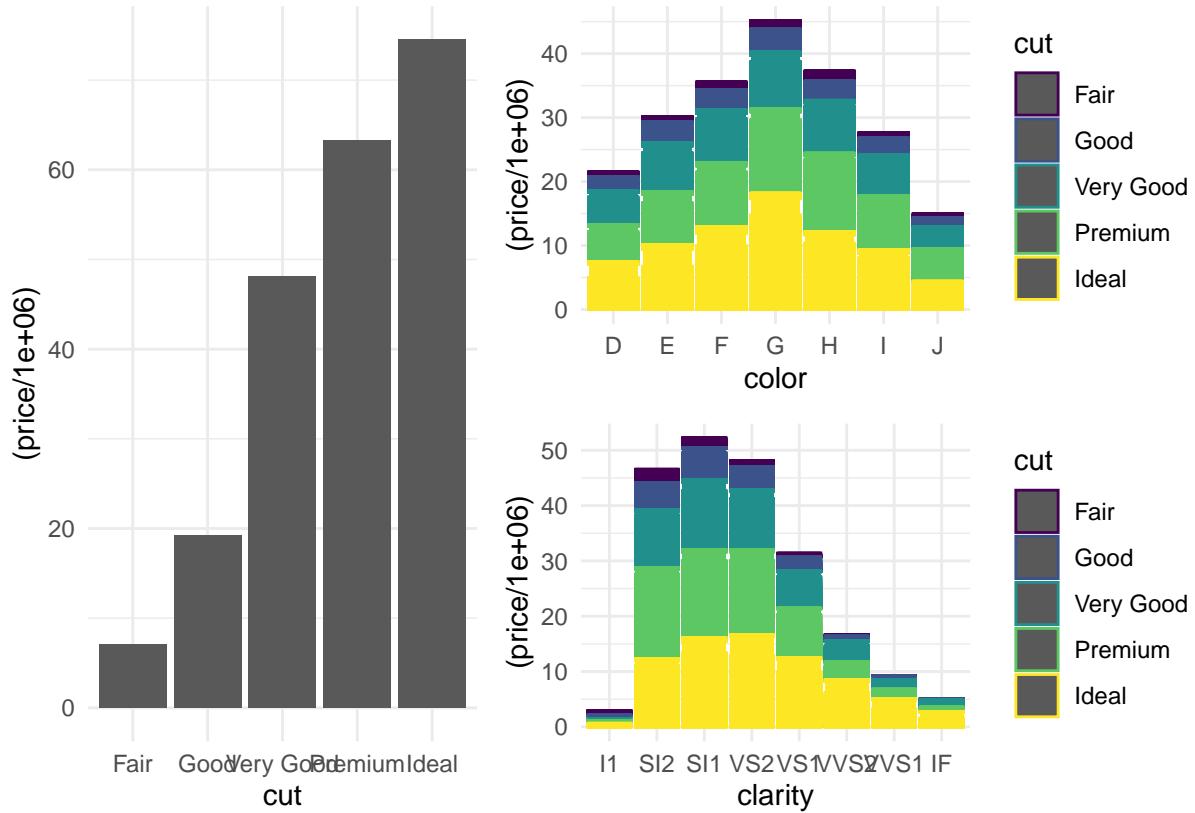
```

g1 = # cut >> price
ggplot(diamonds, aes(cut, (price/1000000)))+
  geom_bar(stat="identity") +
  theme_minimal()

g2 = # color >> price
ggplot(diamonds, aes(color, (price/1000000), color=cut))+ 
  geom_bar(stat="identity") +
  theme_minimal()

g3 = # clarity >> price
ggplot(diamonds, aes(clarity, (price/1000000), color=cut))+ 
  geom_bar(stat="identity") +
  theme_minimal()

g1+g2/g3
  
```



## Relationship between carat and price separated by cut

```
ggplot(diamonds, aes(carat, price))+
  geom_point(size=0.7, alpha=0.4) +
  geom_smooth() +
  theme_minimal() +
  facet_wrap(~ cut #, ncol=5
)
```

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

