Data_Visualization_Homework

2022-10-09

install packages

```
install.packages(c("tidyverse","patchwork","lubridate"))
```

```
## Installing packages into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
install.packages("ggthemes")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

include library

```
library(ggplot2)
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
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
```

```
## v tibble  3.1.8     v purrr   0.3.4
## v tidyr   1.2.1     v stringr 1.4.1
## v readr   2.1.3     v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggthemes)
```

dataset 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>
```

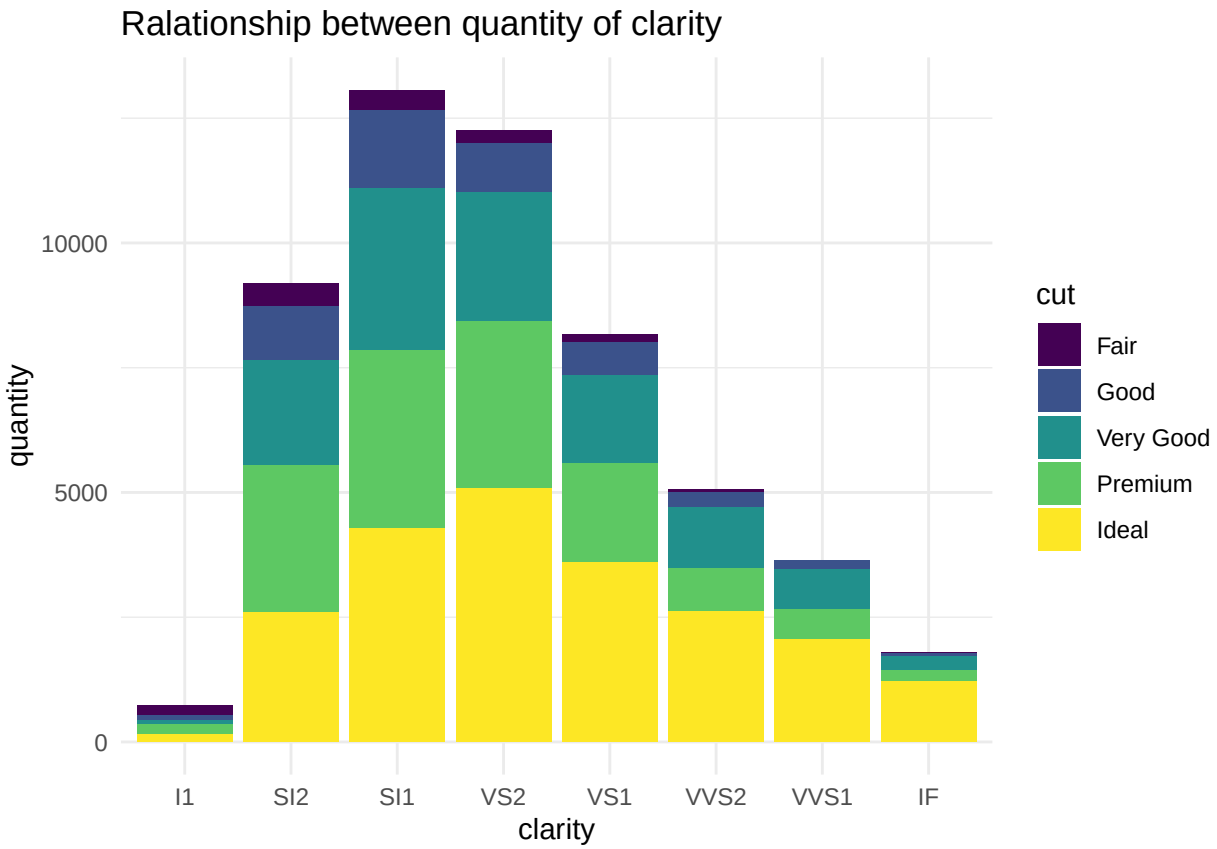
```
## 1  0.23 Ideal     E     SI2      61.5    55   326  3.95  3.98  2.43
## 2  0.21 Premium   E     SI1      59.8    61   326  3.89  3.84  2.31
## 3  0.23 Good      E     VS1      56.9    65   327  4.05  4.07  2.31
## 4  0.29 Premium   I     VS2      62.4    58   334  4.2   4.23  2.63
## 5  0.31 Good      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
```

```
str(diamonds)
```

```
## tibble [53,940 x 10] (S3: tbl_df/tbl/data.frame)
##  $ carat  : num [1:53940] 0.23 0.21 0.23 0.29 0.31 0.24 0.24 0.26 0.22 0.23 ...
##  $ cut    : Ord.factor w/ 5 levels "Fair"<"Good"<..: 5 4 2 4 2 3 3 3 1 3 ...
##  $ color  : Ord.factor w/ 7 levels "D"<"E"<"F"<"G"<..: 2 2 2 6 7 7 6 5 2 5 ...
##  $ clarity: Ord.factor w/ 8 levels "I1"<"SI2"<"SI1"<..: 2 3 5 4 2 6 7 3 4 5 ...
##  $ depth  : num [1:53940] 61.5 59.8 56.9 62.4 63.3 62.8 62.3 61.9 65.1 59.4 ...
##  $ table  : num [1:53940] 55 61 65 58 58 57 57 55 61 61 ...
##  $ price  : int [1:53940] 326 326 327 334 335 336 336 337 337 338 ...
##  $ x      : num [1:53940] 3.95 3.89 4.05 4.2 4.34 3.94 3.95 4.07 3.87 4 ...
##  $ y      : num [1:53940] 3.98 3.84 4.07 4.23 4.35 3.96 3.98 4.11 3.78 4.05 ...
##  $ z      : num [1:53940] 2.43 2.31 2.31 2.63 2.75 2.48 2.47 2.53 2.49 2.39 ...
```

quality number of cuts. from the clarity of diamonds

```
diamonds %>%
  ggplot(aes(clarity,fill=cut)) +
  geom_bar() +
  theme_minimal()+
  labs(
    title = "Ralationship between quantity of clarity",
    x = "clarity",
    y = "quantity"
  )
```



sales and average price of carat

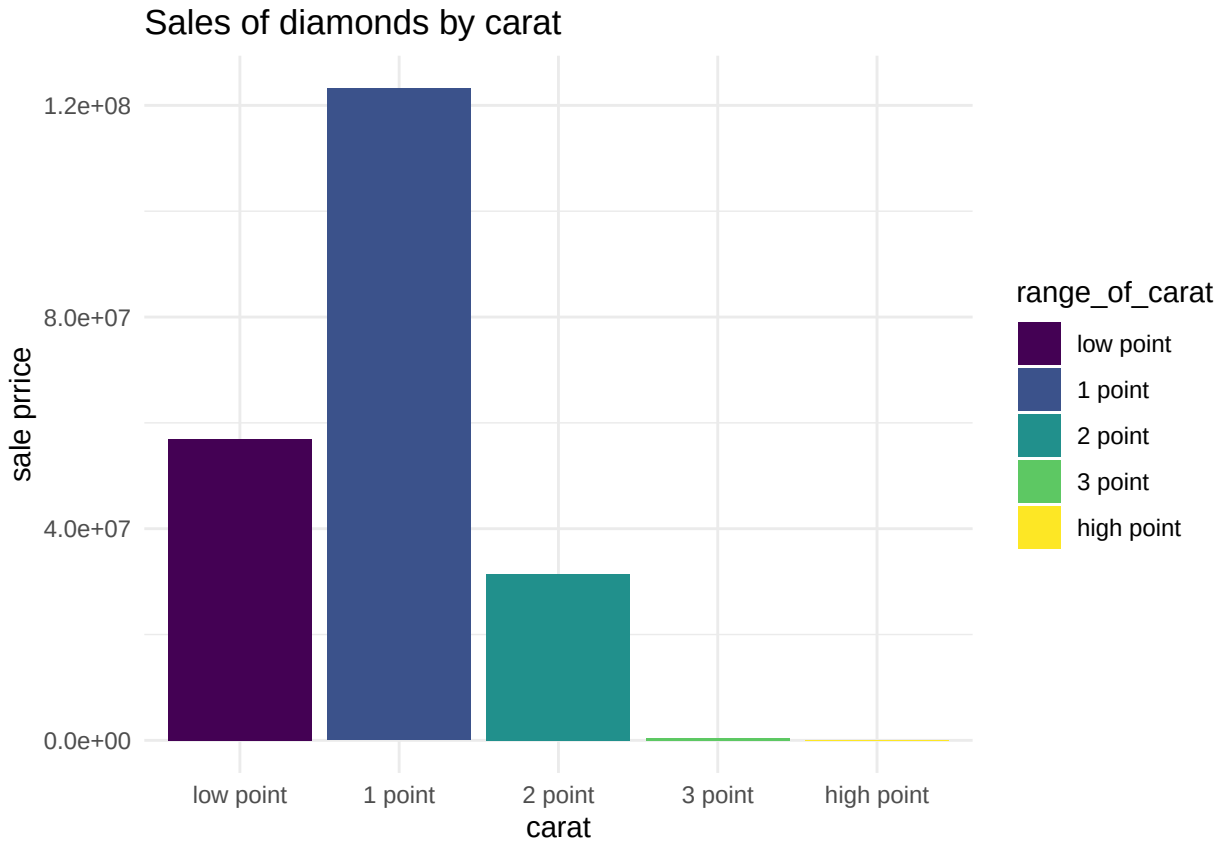
tranform data

```
sal_ave <- diamonds %>%
  select(price, carat) %>%
  mutate(range_of_carat = case_when(
    carat < 1.00 ~ "low point",
    carat < 2.00 ~ "1 point",
    carat < 3.00 ~ "2 point",
    carat < 4.00 ~ "3 point",
    TRUE ~ "high point"
  )) %>%
  mutate(range_of_carat = factor(
    range_of_carat,
    labels = c("low point", "1 point", "2 point", "3 point", "high point"),
    levels = c("low point", "1 point", "2 point", "3 point", "high point"),
    ordered = TRUE)) %>%
  group_by(range_of_carat) %>%
  summarise(sum_price = sum(price), mean_price = mean(price))
```

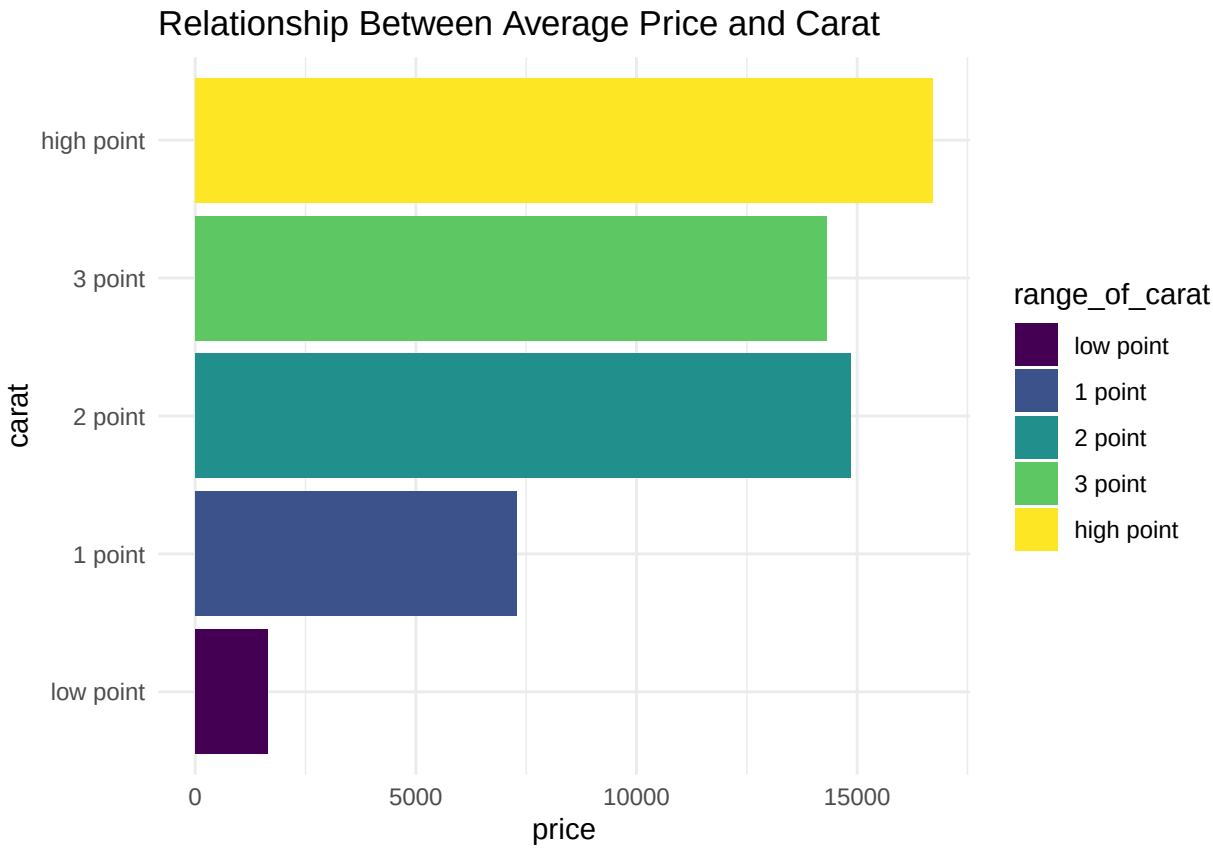
plot chart

```
sal_ave %>%
  ggplot(aes(range_of_carat, sum_price,fill=range_of_carat))+
  geom_col()+
  theme_minimal()+
```

```
  labs(
    title = "Sales of diamonds by carat",
    x = "carat",
    y = "sale prrice"
  )
```




```
sal_ave %>%
  ggplot(aes(mean_price,range_of_carat,fill=range_of_carat))+
  geom_col()+
  theme_minimal()+
  labs(
    title = "Relationship Between Average Price and Carat",
    x = "price",
    y = "carat"
  )
```



area of cut ### tranfrom data

```
area_of_cut <- diamonds %>%
  select(x,y,z,price,cut) %>%
  mutate(area = round(x*y*z,digits = 1))
```

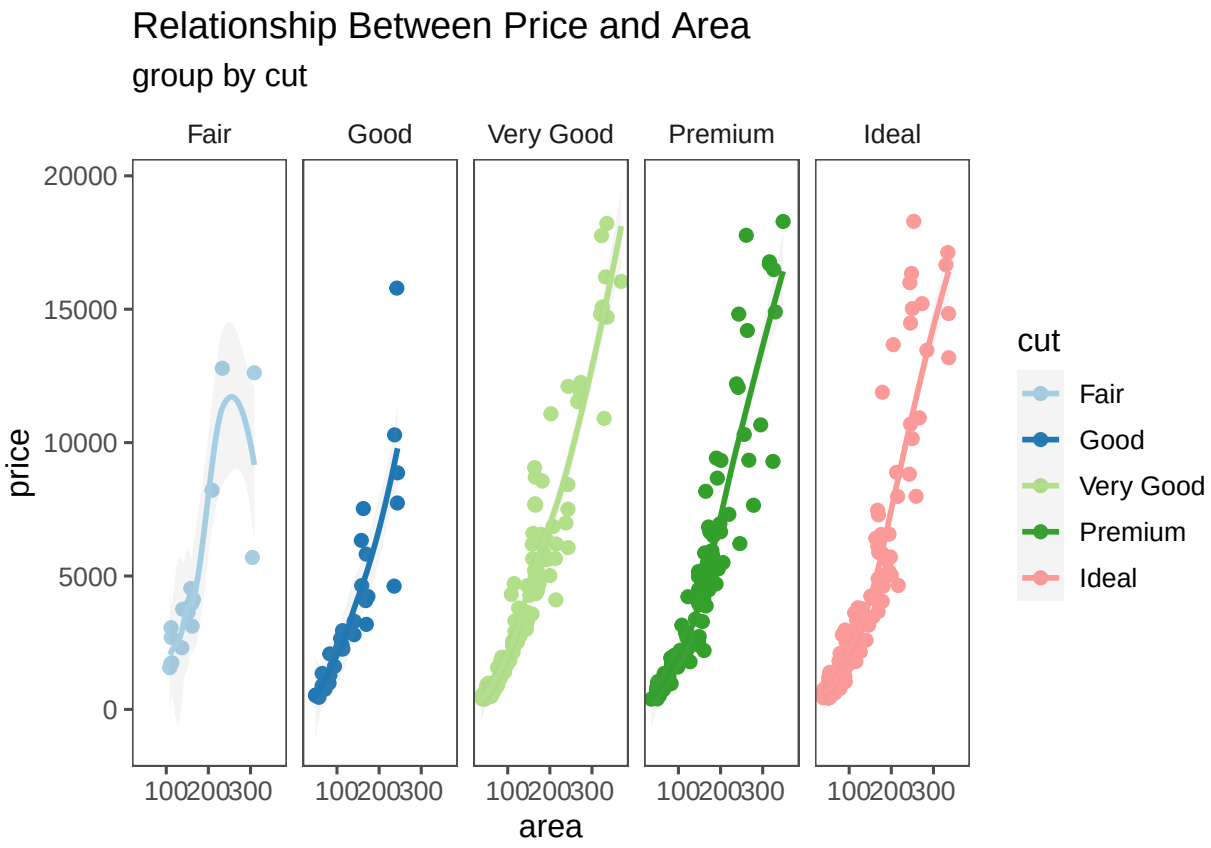
plot chart

```
set.seed(25)
sample_n(area_of_cut,500) %>%
  ggplot(aes(area, price, color=cut)) +
  geom_point(alpha=0.5,size=2)+
  geom_smooth(method = "lm", se=F)+
  theme_few()+
  scale_color_brewer(type = "qual",palette = 3)+
  labs(
    title = "Relationship Between Price and Area",
    x = "area",
    y = "price"
  )
```

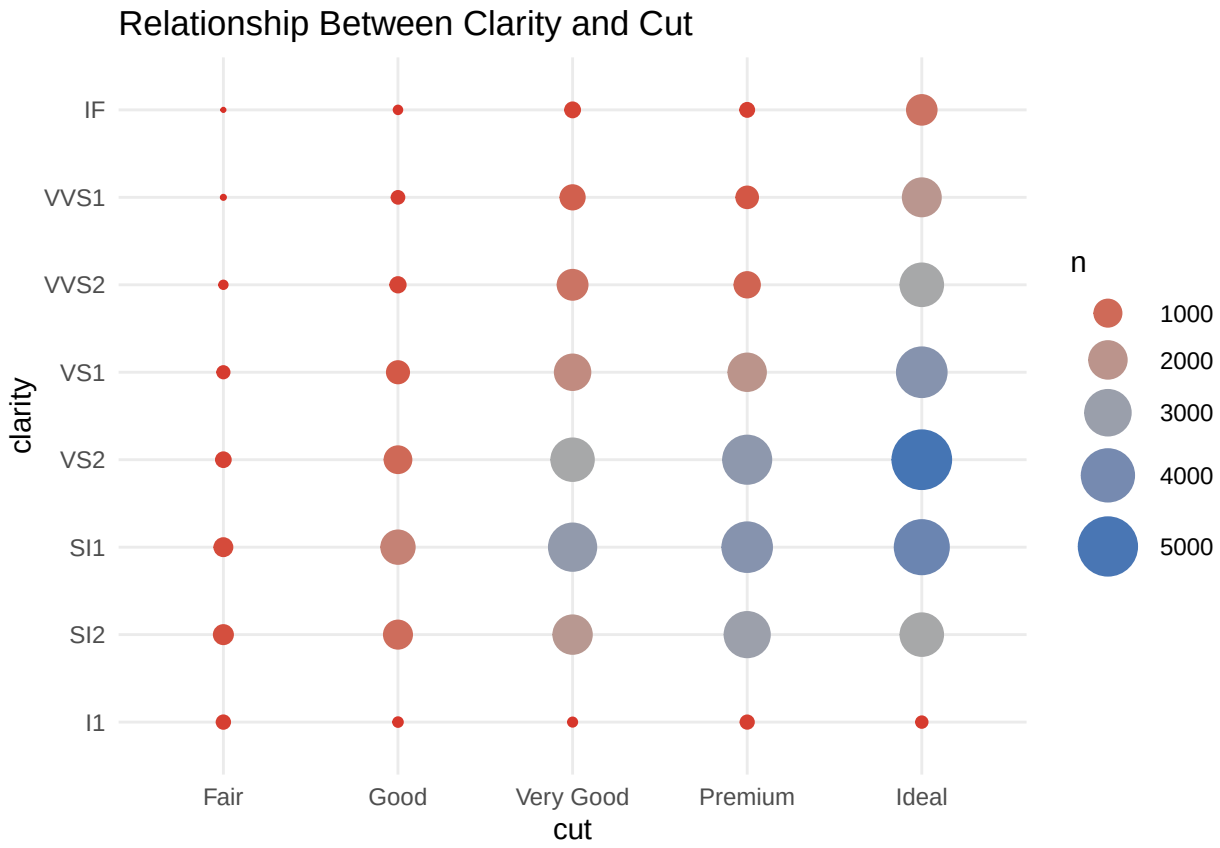
```
## `geom_smooth()` using formula 'y ~ x'
```


```
set.seed(25)
sample_n(area_of_cut,500) %>%
  ggplot(aes(area, price, color=cut)) +
  geom_point(alpha=1,size=2)+
  geom_smooth(alpha=0.1)+
  facet_wrap(~cut,ncol = 5)+
  theme_few()+
  scale_color_brewer(type = "qual",palette = 3)+
  labs(
    title = "Relationship Between Price and Area",
    subtitle = "group by cut",
    x = "area",
    y = "price"
  )
```

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




```
ggplot(diamonds, aes(cut, clarity)) +
  geom_count(aes(color = ..n.., size = ..n..)) +
  guides(color = 'legend')+
  scale_colour_gradientn(colours = c("#d73027","darkgrey","#4575b4"))+
  scale_size_area(max_size = 10)+
  theme_minimal()+
  labs(
    title = "Relationship Between Clarity and Cut",
    x = "cut",
    y = "clarity"
  )
```


Relationship Between Clarity and Cut
IF
VVS1
VVS2
VS1
VS2
SI1
SI2
I1
clarity
Fair
Good
Very Good
Premium
Ideal
cut
n
1000
2000
3000
4000
5000