NHST I

PSYC 2020-A01 / PSYC 6022-A01 | 2025-10-03 | Lab 7

Jessica Helmer

Outline

- Assignment 6 Review
- NHST Review
- Exploratory data analysis

Learning objectives:
R: Making data tidy, line graphs

Housekeeping

[placeholder for any housekeeping stuff]

Assignment 5 Review

[placeholder for Assignment 5 review]

NHST Review

Null Hypothesis Significance Testing

[placeholder for more NHST review]

Exploratory Data Analysis

Exploratory Data Analysis

Going to once again heavily lean on this book!

Feel free to reference for more R content

https://r4ds.hadley.nz/

Exploratory Data Analysis (EDA): Overview

Whether you have a specific testing plan or not, need to explore your data

- Otherwise, leaving information on the table!
- If nothing else, need to investigate quality of your data

An iterative cycle:

1. Generate questions about your data.
2. Search for answers by visualizing, transforming, and modelling your data.
3. Use what you learn to refine your questions and / or generate new questions.

Exploratory Data Analysis (EDA): Overview

Your goal is to develop an understanding of your data

Useful to use questions as guides

We’ve done this some so far! Today we’re going to focus in on it.

Good questions are not always clear at the beginning, but try to follow up every question with a new one.

Exploratory Data Analysis (EDA): Questions

Some good general questions:

1. What type of variation occurs within my variables?
2. What type of associations occur between / among my variables?

In statistics, we learn ways to identify particularly strong variation or associations.

In EDA, we can still get a strong sense of these relationships without statistical tests.

- ○ Can guide future confirmatory testing
- ○ May be representative of population-level relationships with large samples

EDA: Looking for variation

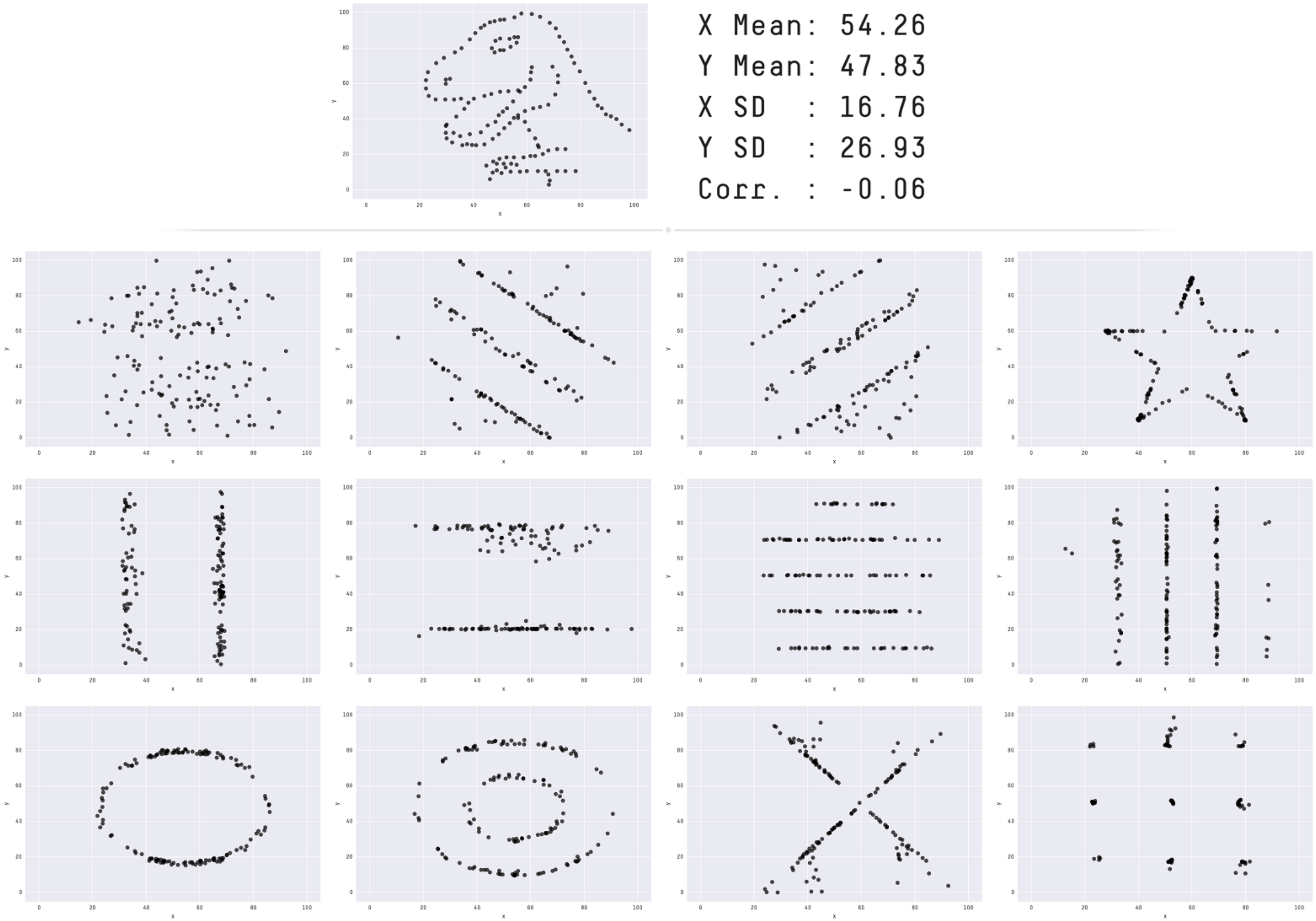
Review: what is variation?

Spread or dispersion in our data

Can also think about the amount of differences we see in our measurement of something

We have statistics to give us summaries of the amount of variance in our data, but that doesn’t tell us exactly what it looks like.

Remember This?



```
X Mean: 54.26
Y Mean: 47.83
X SD  : 16.76
Y SD  : 26.93
Corr. : -0.06
```

EDA: Variation

We’re gonna play with the `diamonds` dataset included within the `tidyverse` library.

```r
diamonds
```

```
# A tibble: 53,940 × 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
 7  0.24 Very Good I     VVS1     62.3    57   336  3.95  3.98  2.47
 8  0.26 Very Good H     SI1      61.9    55   337  4.07  4.11  2.53
 9  0.22 Fair      E     VS2      65.1    61   337  3.87  3.78  2.49
10  0.23 Very Good H     VS1      59.4    61   338  4     4.05  2.39
# ℹ 53,930 more rows
```

```r
nrow(diamonds)
```

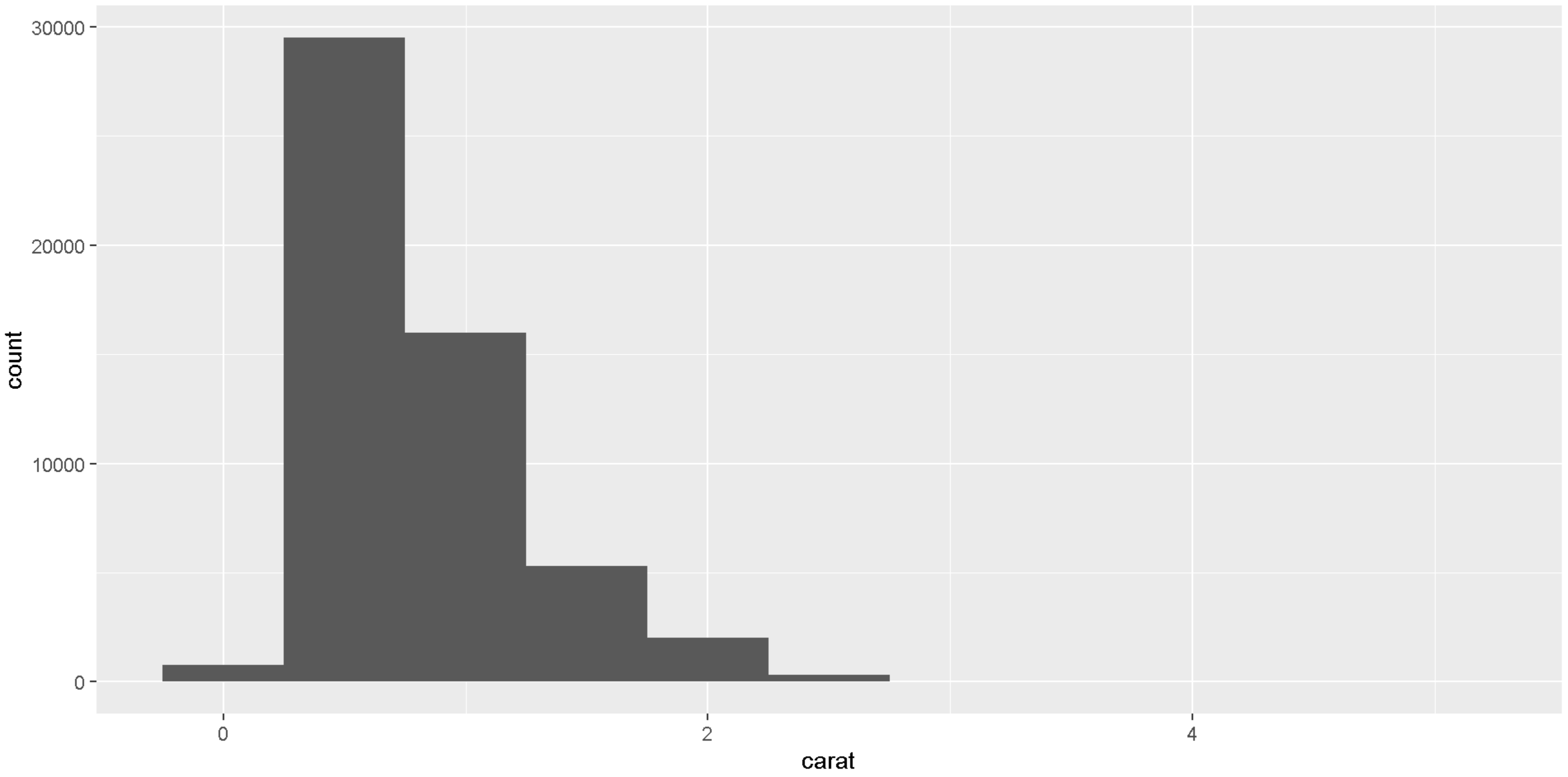
```
[1] 53940
```

EDA: Variation

Let’s start by looking at the distribution of weights (`carat`)

How might we look at this?

```r
ggplot(diamonds, aes(x = carat)) +
  geom_histogram(binwidth = 0.5)
```


30000
20000
10000
0
count
0
2
4
carat

EDA: What to look for?

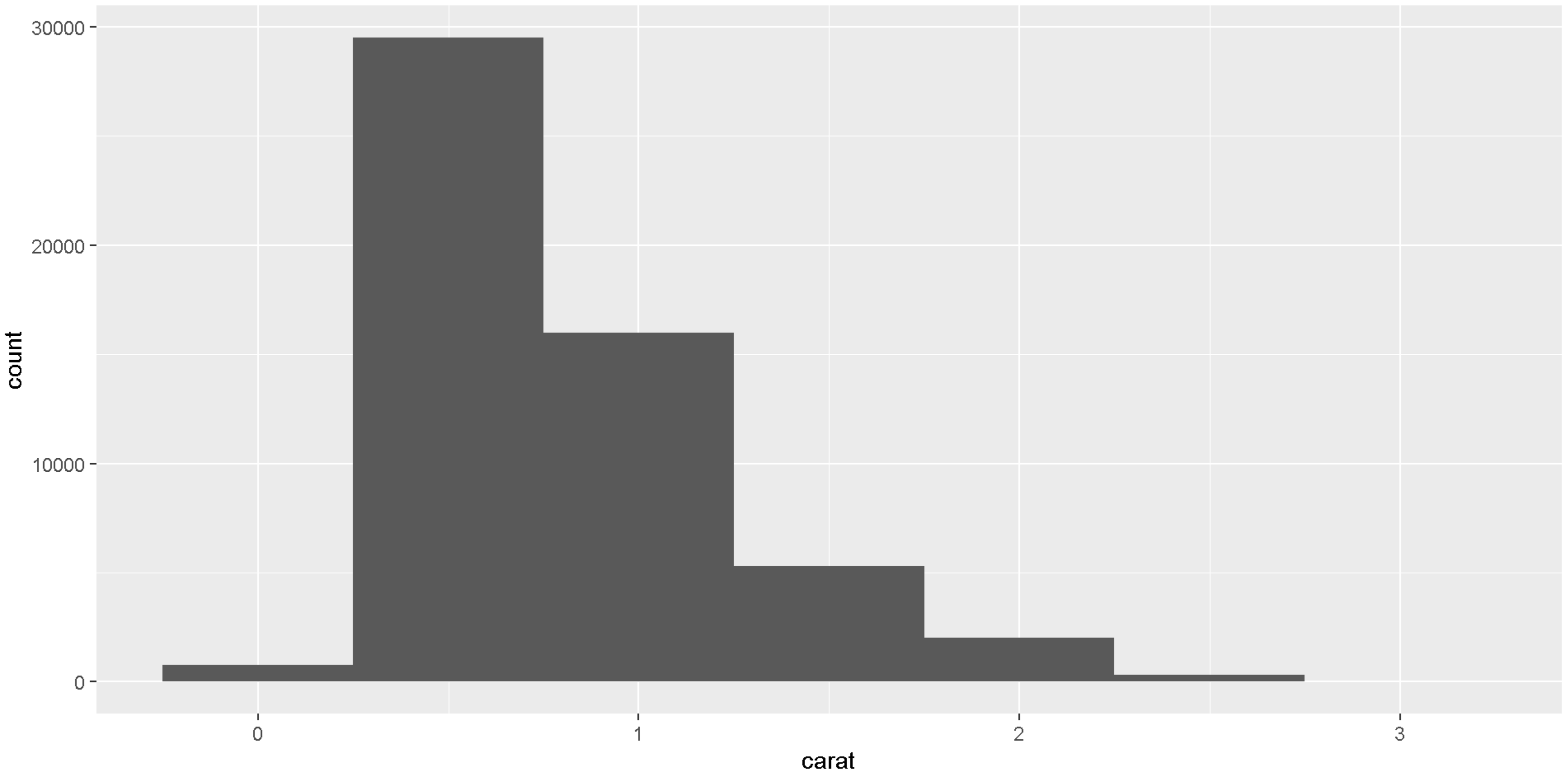
- Which values are the most common? Why?
- Which values are rare? Why? Does that match your expectations?
- Can you see any unusual patterns? What might explain them?

Want to rely on our curiousity (what do we want to know more about?) and our skepticism (how could this be misleading?)

EDA: Small diamonds

Let’s start by zooming in on small diamonds

```r
diamonds |>
  filter(carat <= 3) |>
  ggplot(aes(x = carat)) +
  geom_histogram(binwidth = 0.5)
```



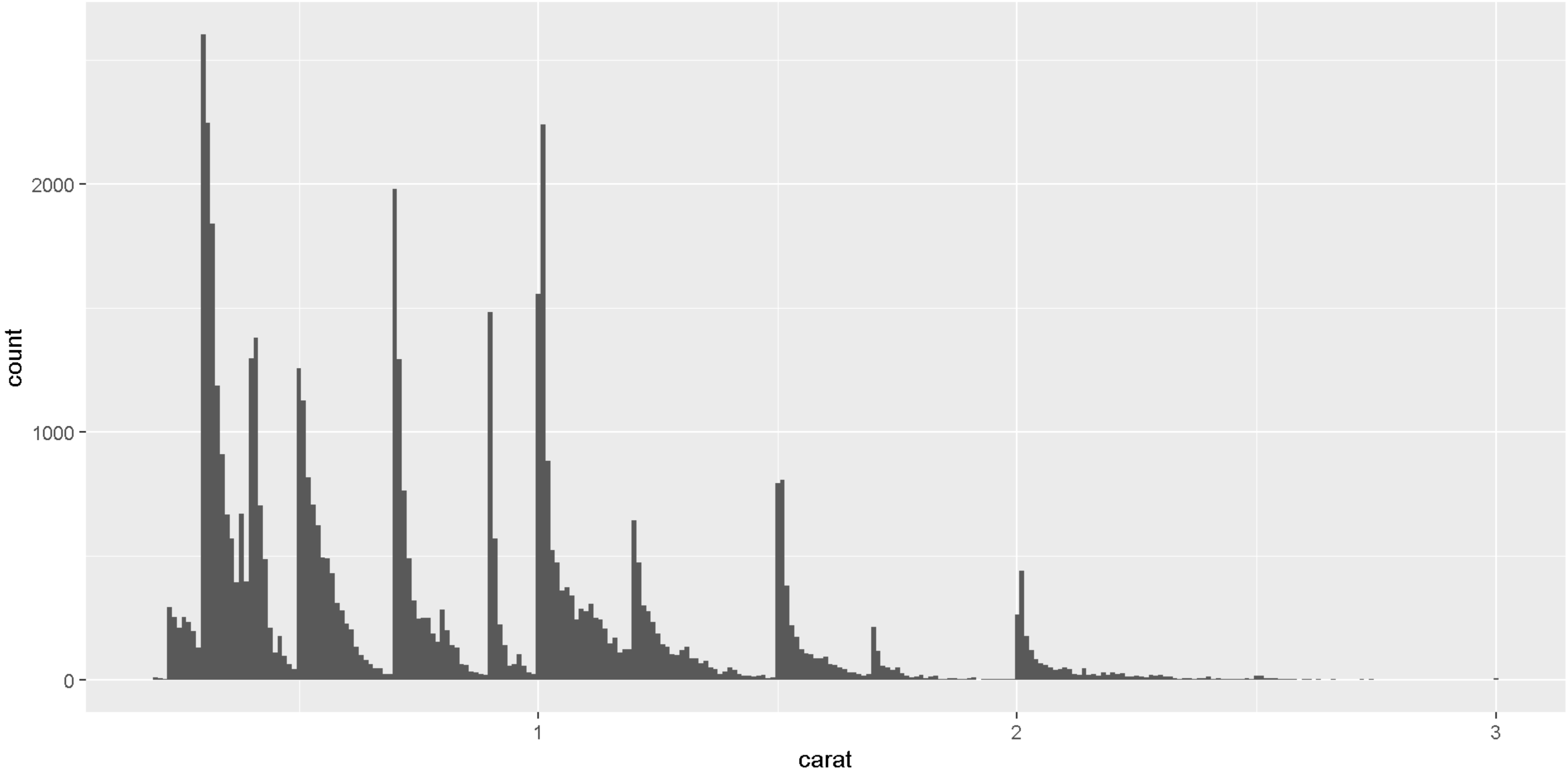
What do we want to know more about?

How might this be misleading?

EDA: Small diamonds

Let’s look at this distribution with more precision!

```r
diamonds |>
  filter(carat <= 3) |>
  ggplot(aes(x = carat)) +
  geom_histogram(binwidth = 0.01)
```



Questions?

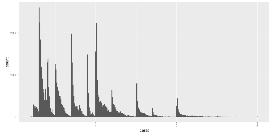
Why are there more diamonds at whole carats and common fractions of carats?

Why are there more diamonds slightly to the right of each peak than

EDA: Clustering

Seeing clustering may mean we have subgroups in our data

```r
diamonds |>
  filter(carat <= 3) |>
  ggplot(aes(x = carat)) +
  geom_histogram(binwidth = 0.01)
```



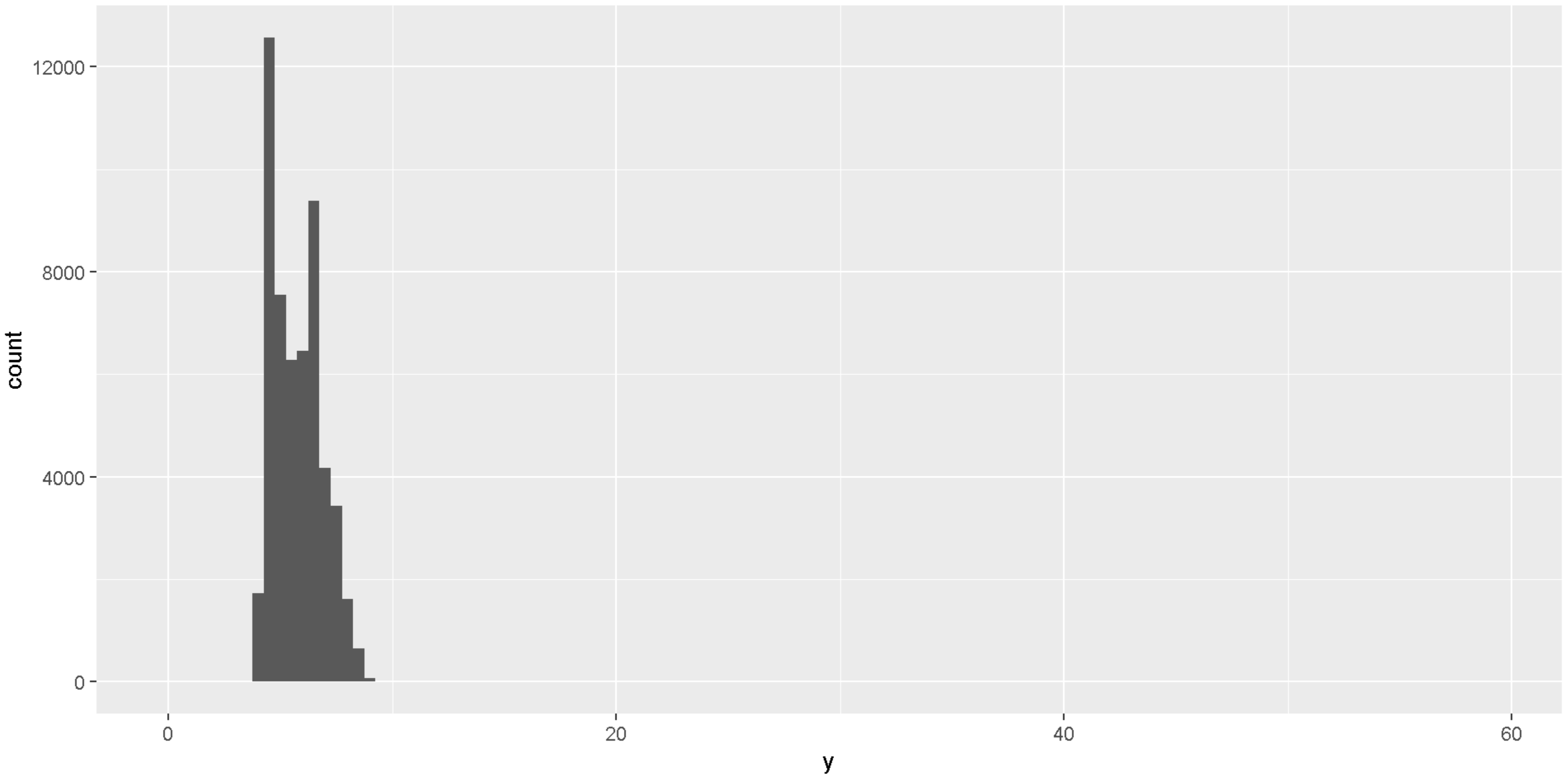
- How are the observations within each subgroup similar to each other?
- How are the observations in separate clusters different from each other?
- How can you explain or describe the clusters?
- Why might the appearance of clusters be misleading?

EDA: Unusual Values

Like outliers! Things that don’t fit the rest of the pattern.

Let’s look at the y variable (diamond width) in this dataset

```r
ggplot(diamonds, aes(x = y)) +
  geom_histogram(binwidth = 0.5)
```



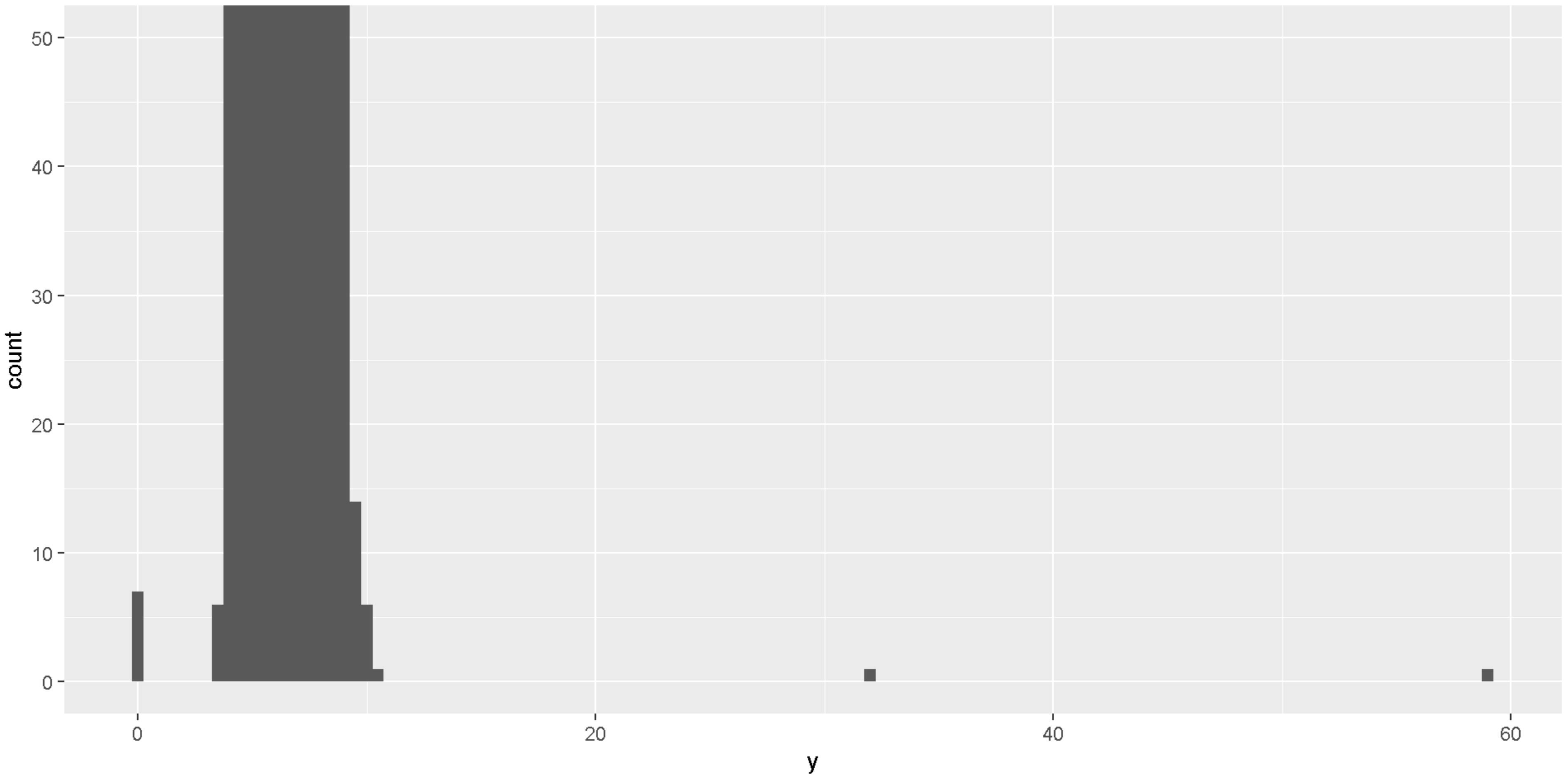
What do we notice?

EDA: Unusual Values

Can’t even see any outliers because the high points are so high

Let’s zoom in (`coord_cartesian()`)

```r
ggplot(diamonds, aes(x = y)) +
  geom_histogram(binwidth = 0.5) +
  coord_cartesian(ylim = c(0, 50))
```



Note

`ggplot2` also has `xlim()` and `ylim()` functions, but they are different: they throw away the data outside the limits

Where are the unusual data?

EDA: Unusual Values

When we find unusual values, it’s good to then go back to the data

```r
diamonds |>
  filter(y < 3 | y > 20) |>
  select(price, x, y, z) |>
  arrange(y)
```

```
# A tibble: 9 × 4
  price     x     y     z
  <int> <dbl> <dbl> <dbl>
1  5139  0      0    0
2  6381  0      0    0
3 12800  0      0    0
4 15686  0      0    0
5 18034  0      0    0
6  2130  0      0    0
7  2130  0      0    0
8  2075  5.15  31.8  5.12
9 12210  8.09  58.9  8.06
```

Since width cannot be zero, we know we found missing data that was coded as zero!

What’s going on with the large ones? Can we use other variables to infer if they’re accurate or not?

EDA: Unusual Values

Outliers: what do?

Above all else, be transparent, and don’t remove them without making a note in your report

Lots and lots of potential ways of dealing with outliers, and lots and lots of ways that can have implications for your analysis

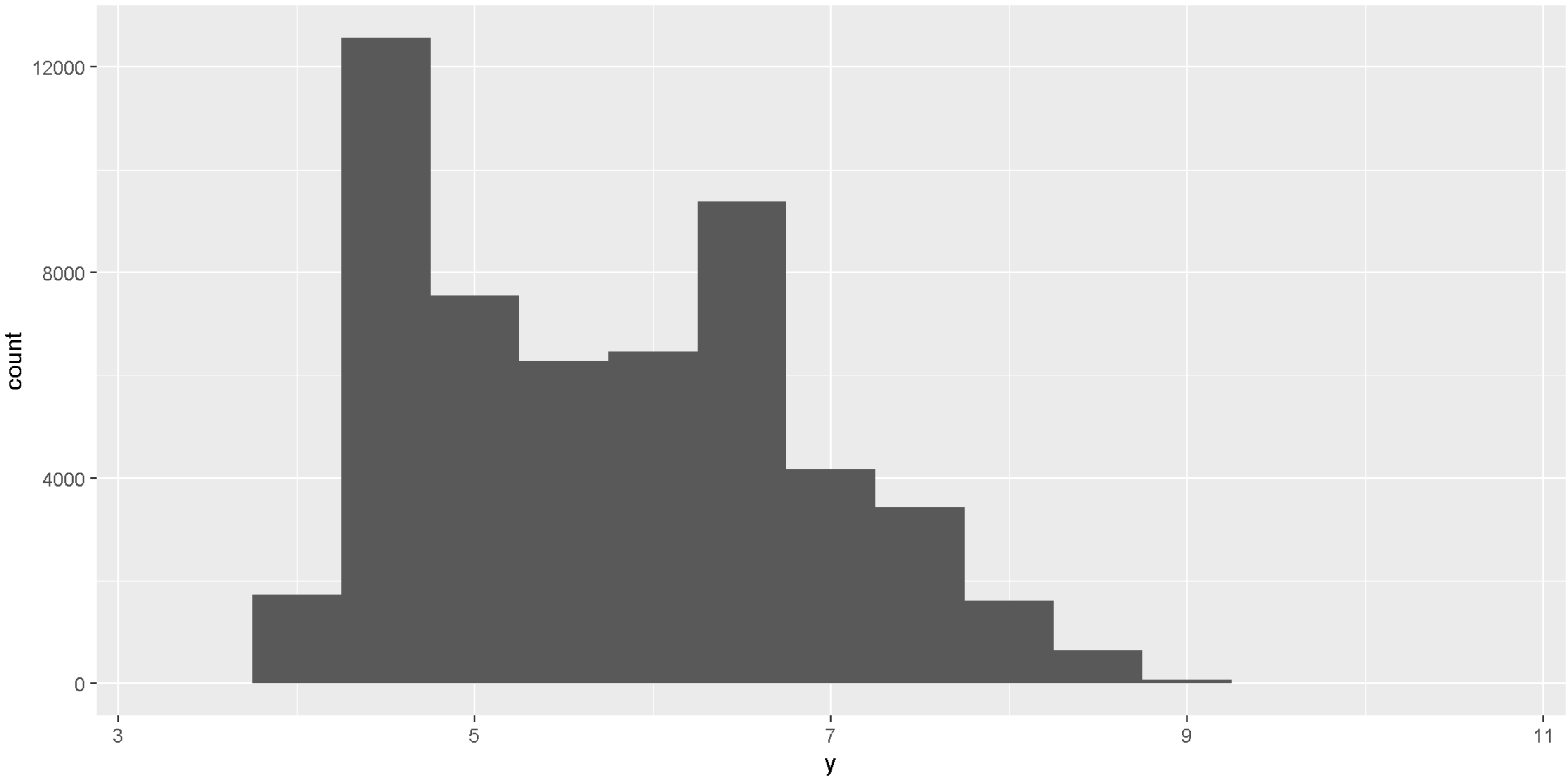
A basic guide is to try doing your analysis with and without the outliers to see how much impact removing them has

More on that in advanced statistics classes!

EDA: Unusual values

For our purposes, let’s at least change those unusual values to NA

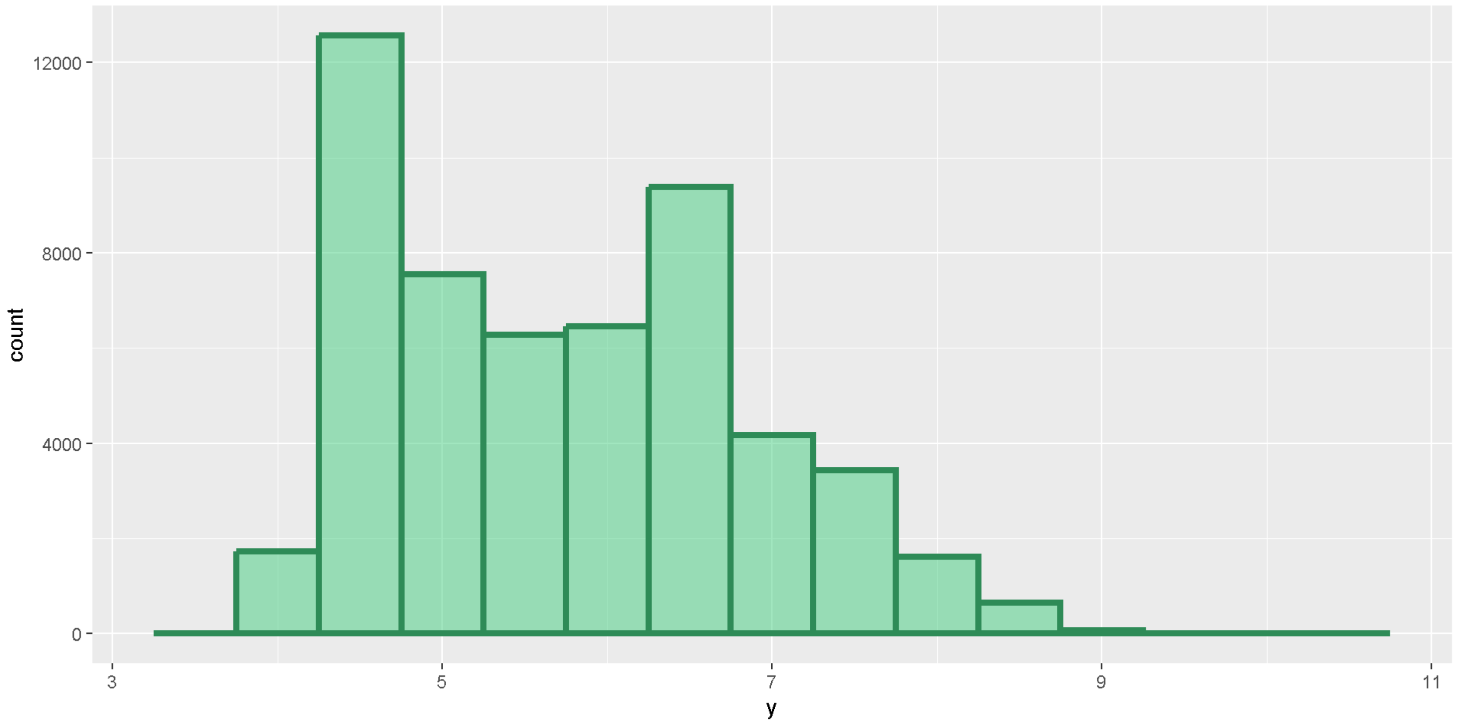
```r
diamonds2 <- diamonds |>
  mutate(y = ifelse(y < 3 | y > 20, NA, y))

ggplot(diamonds2, aes(x = y)) +
  geom_histogram(binwidth = 0.5, na.rm = T)
```


12000
8000
4000
0
count
3
5
7
9
11
y

Histograms: An aesthetics interlude

What can we do to change the appearance of a `geom_histogram()`?

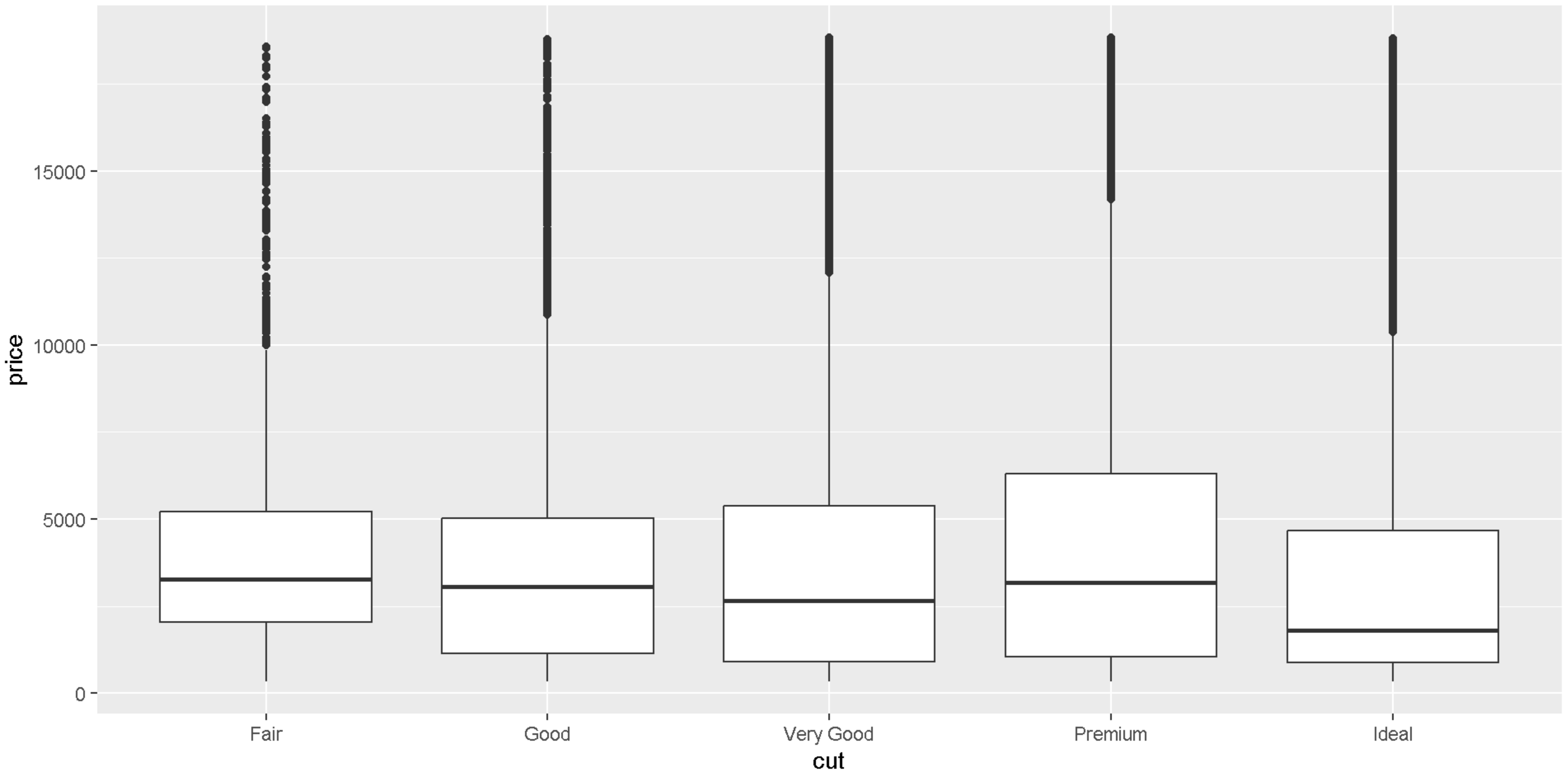
```r
ggplot(diamonds2, aes(x = y)) +
  geom_histogram(binwidth = 0.5, na.rm = T,
                 fill = "seagreen3", color = "seagreen",
                 alpha = .5, linewidth = 1.5)
```



EDA: Categorical and numeric variable associations

How might the price of a diamond vary by its quality (`cut`)?

```r
diamonds |>
  ggplot(aes(y = price, x = cut)) +
  geom_boxplot()
```



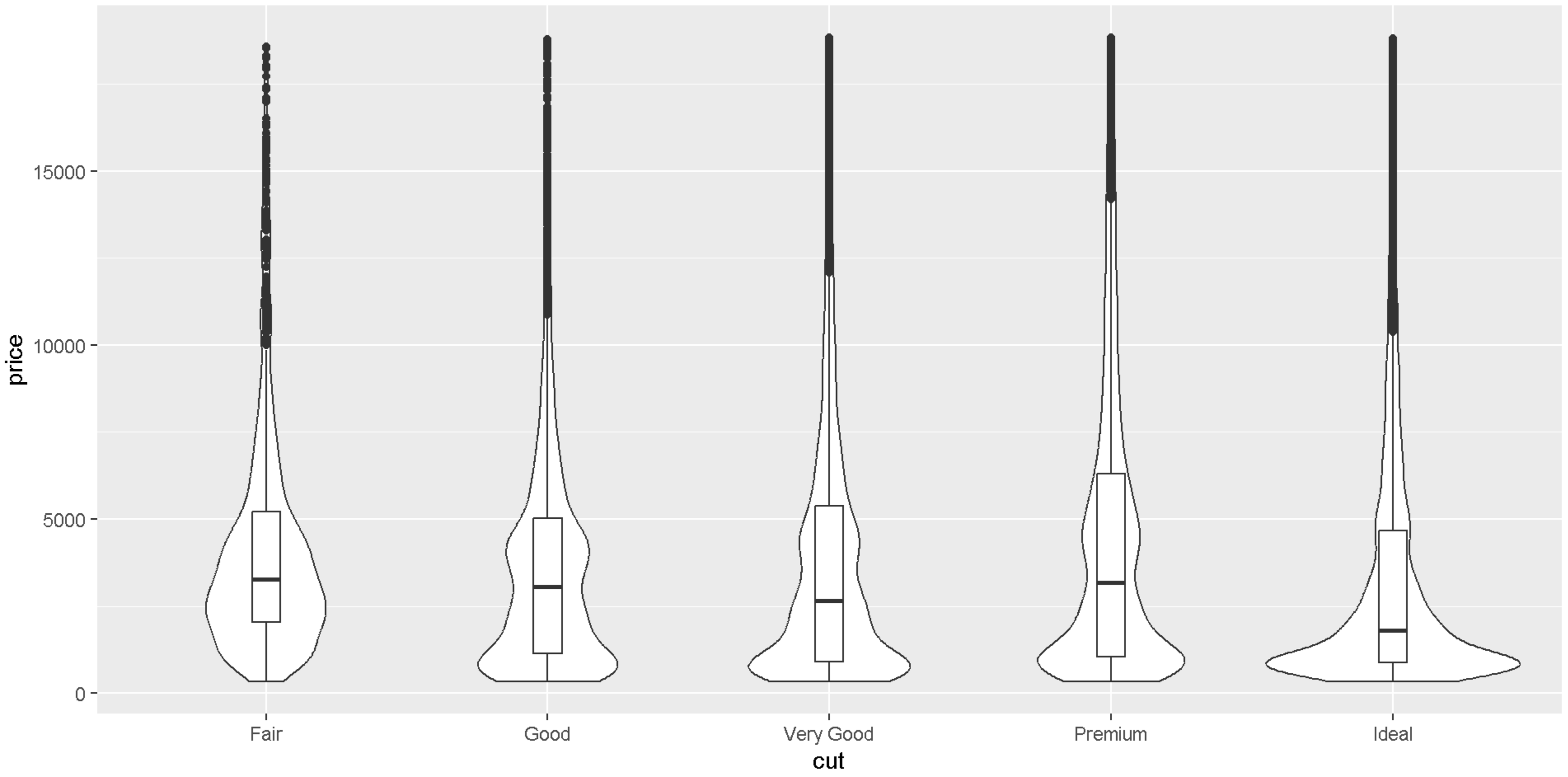
What do we want to know more about?

How might this be misleading?

EDA: Categorical and numeric variable associations

Want to see the distributions as well too!

```
diamonds |>
  ggplot(aes(y = price, x = cut)) +
  geom_violin() +
  geom_boxplot(width = .1)
```



Notice that `cut` was already ordered for us! Thank you, `factor`s!

```
diamonds |> pull(cut) |> levels()
```

```
[1] "Fair"      "Good"      "Very Good" "Premium"   "Ideal"
```

Factors: An ordering interlude

What if our variable is a factor, but we want to order it by a different variable?

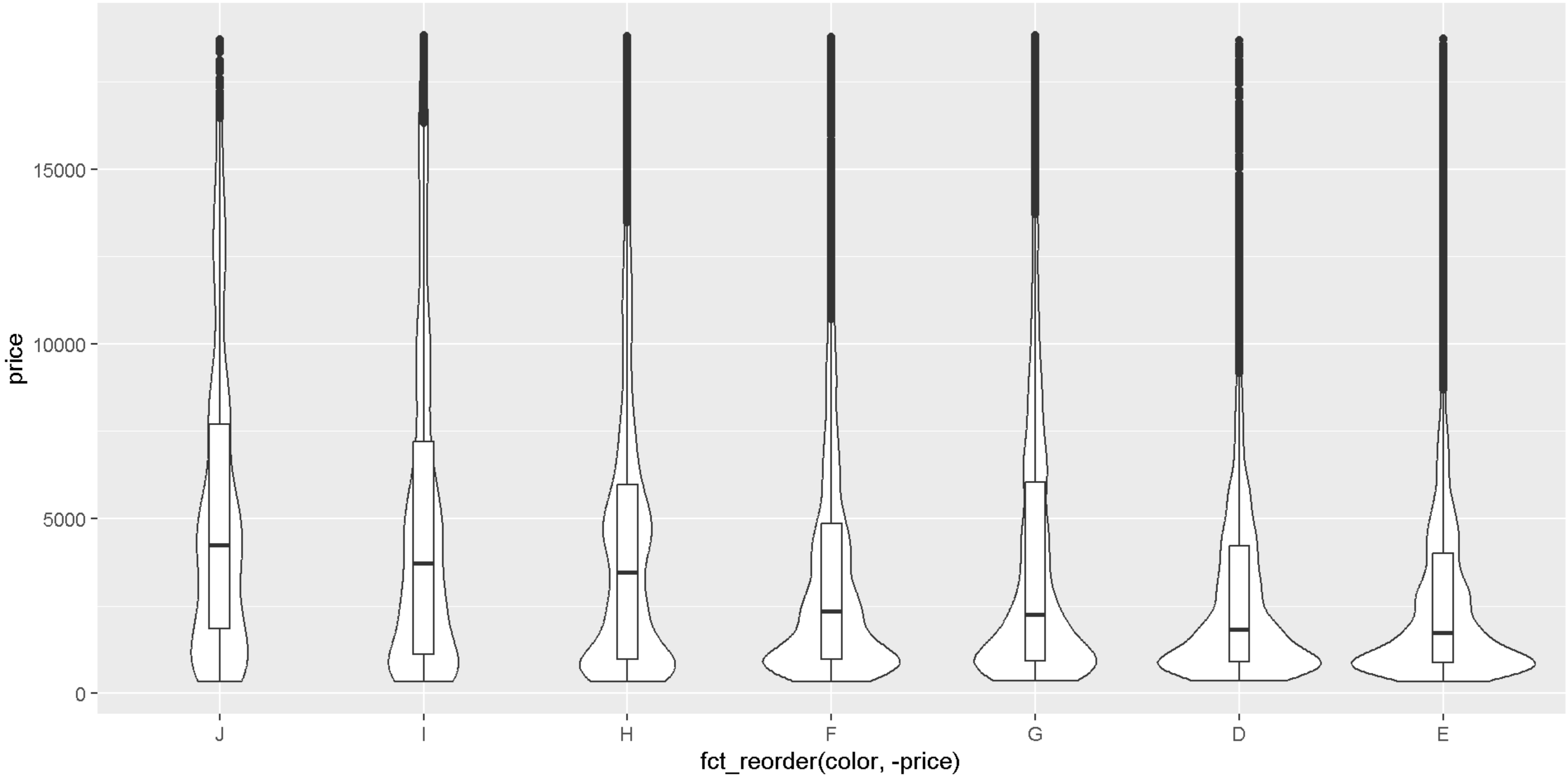
```r
levels(diamonds$color)
```

```
[1] "D" "E" "F" "G" "H" "I" "J"
```

```r
fct_reorder(diamonds$color, diamonds$price) |> levels()
```

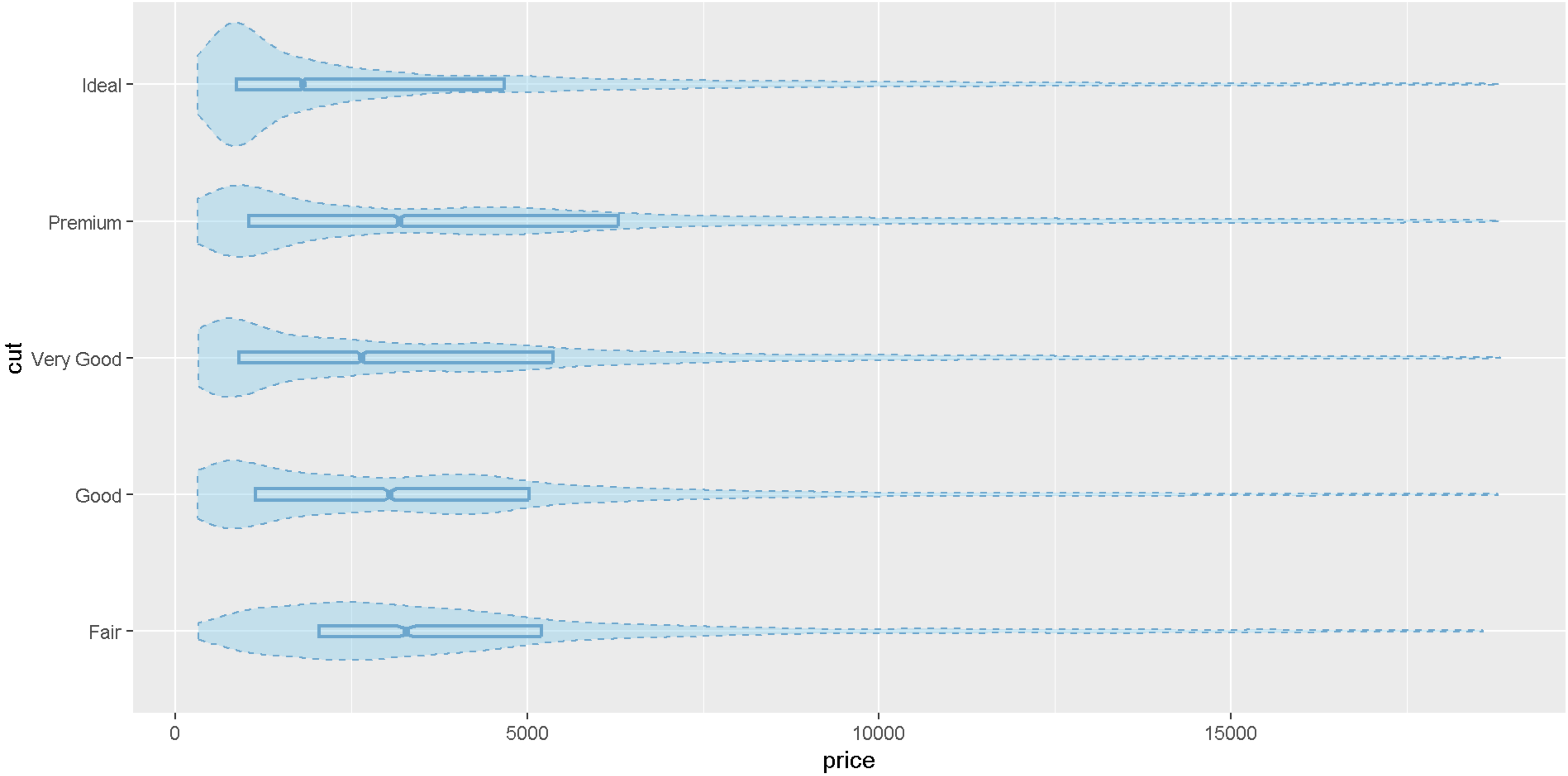
```
[1] "E" "D" "G" "F" "H" "I" "J"
```

```r
diamonds |>
  ggplot(aes(y = price, x = fct_reorder(color, -price))) + # or .desc = T
  geom_violin() +
  geom_boxplot(width = .1)
```


15000
10000
5000
0
price
J
I
H
F
G
D
E
fct_reorder(color, -price)

Boxplots and Violinplots: An aesthetics interlude

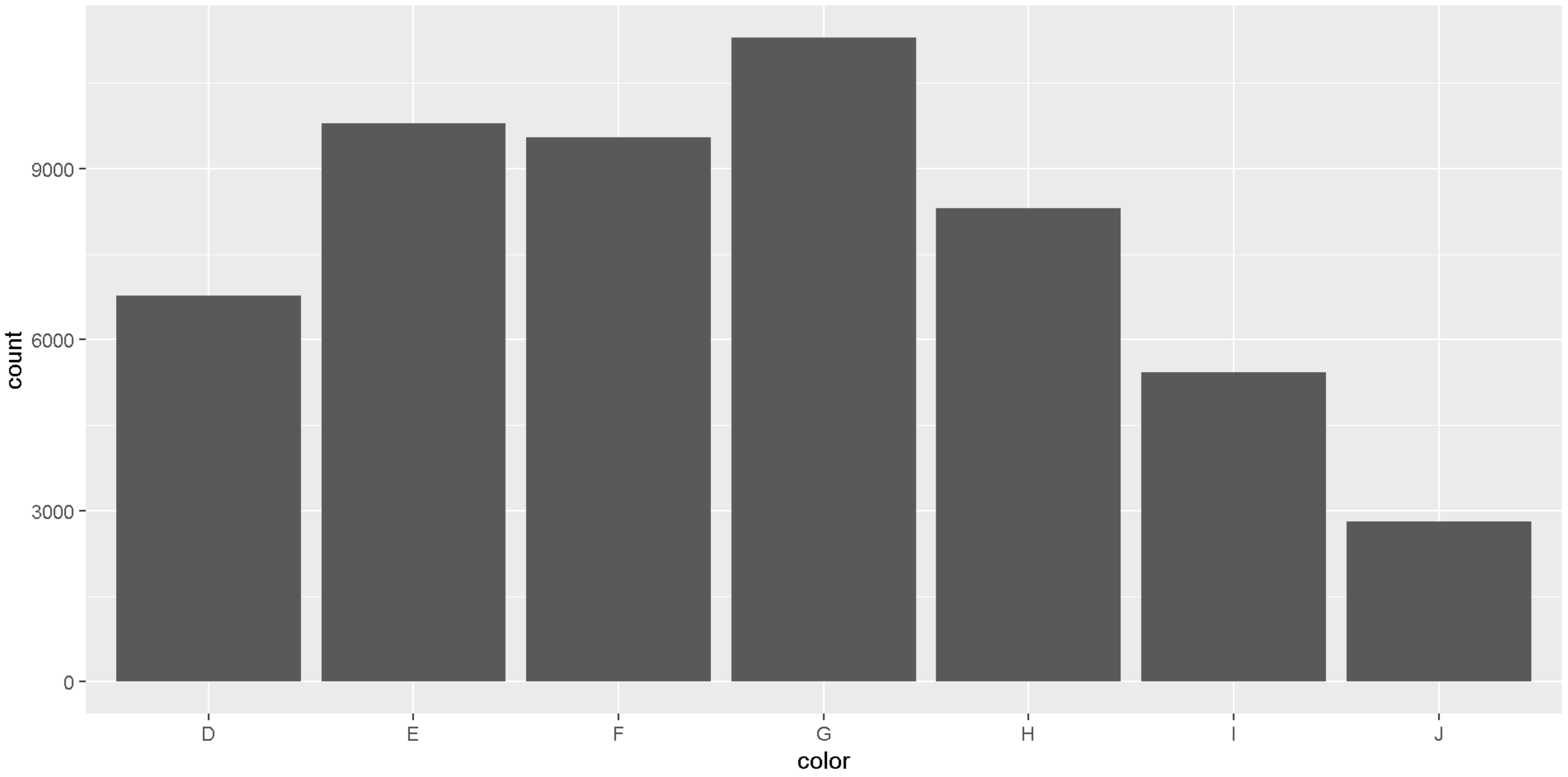
```r
diamonds |>
  ggplot(aes(x = price, y = cut)) +
  geom_violin(color = "skyblue3", fill = "skyblue",
              alpha = .4, linewidth = .5, linetype = "dashed") +
  geom_boxplot(width = .08, color = "skyblue3", whisker.color = NA,
               fill = "transparent", linewidth = .8,
               outliers = F, notch = T)
```


cut
Ideal
Premium
Very Good
Good
Fair
0
5000
10000
15000
price

EDA: Categorical and categorical variable associations

Often exploring counts and / or proportions

```
diamonds |>
  ggplot(aes(x = color)) +
  geom_bar()
```

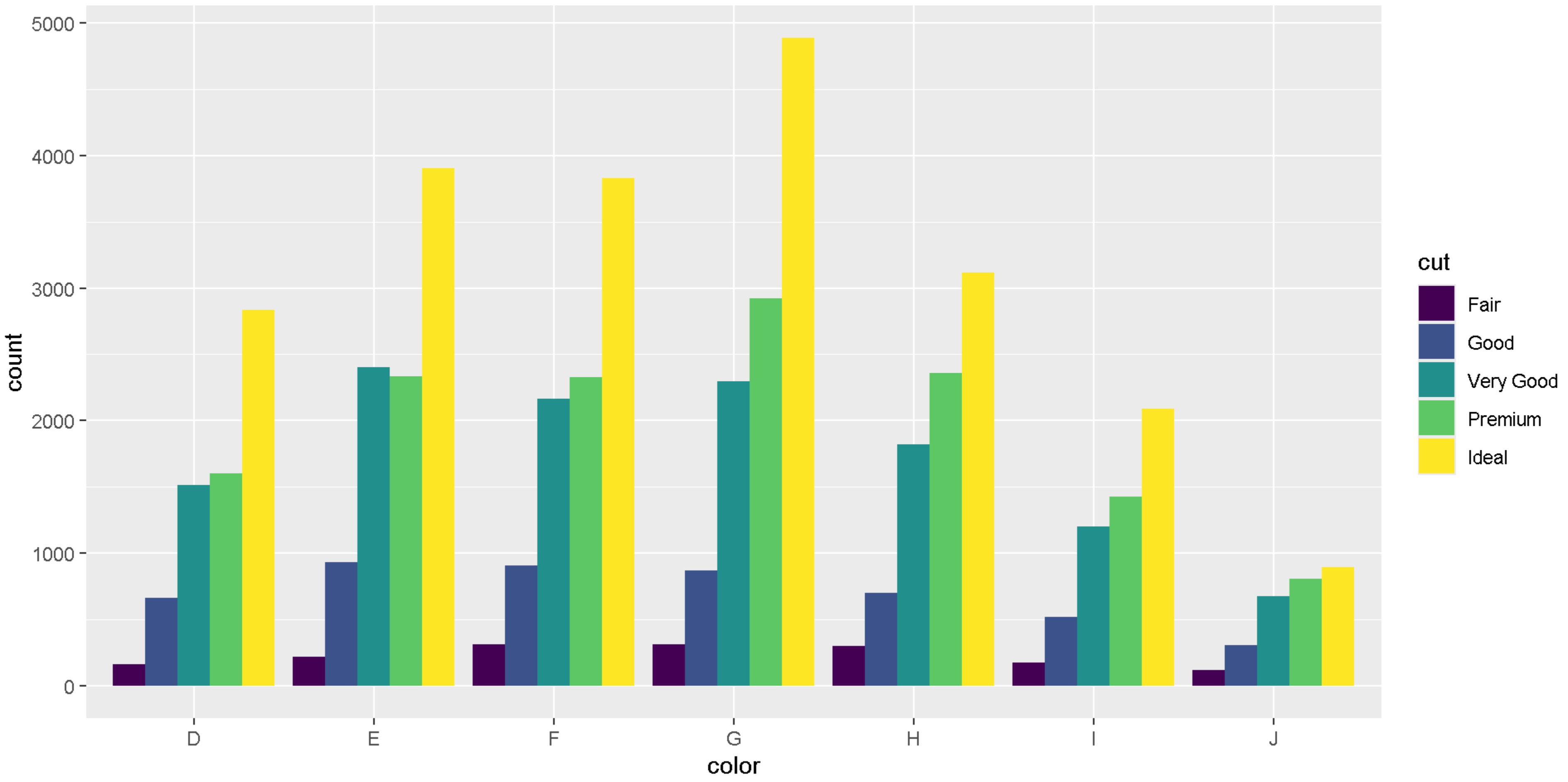


Does this change for diamonds of different qualities (`cut`s)?

EDA: Categorical and categorical variable associations

Often exploring counts and / or proportions

```r
diamonds |>
  ggplot(aes(x = color, fill = cut)) +
  geom_bar(position = "dodge")
```

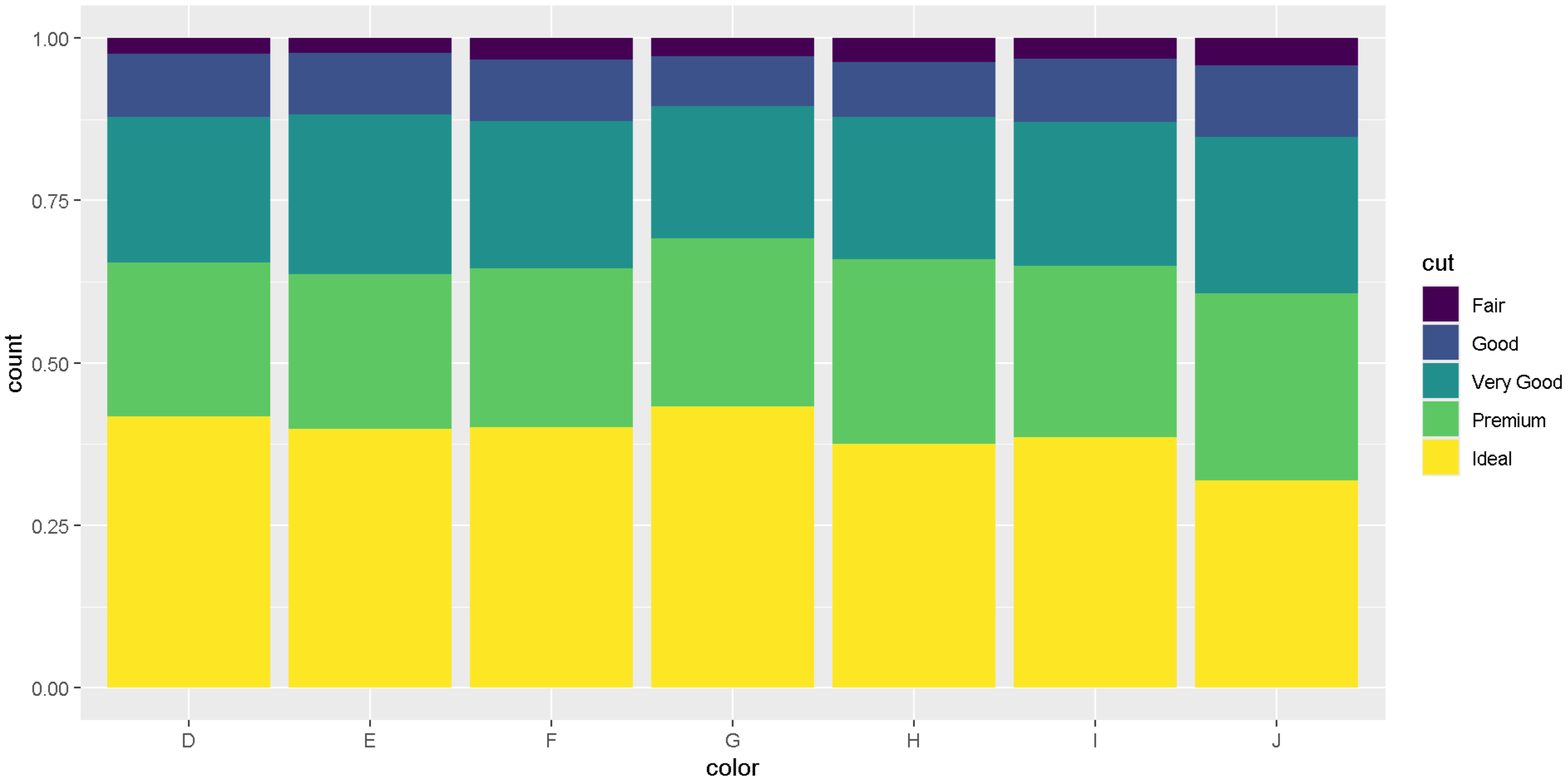


Is the total count for each color obscuring the proportions of qualities within color?

EDA: Categorical and categorical variable associations

Often exploring counts and / or proportions

```r
diamonds |>
  ggplot(aes(x = color, fill = cut)) +
  geom_bar(position = "fill")
```

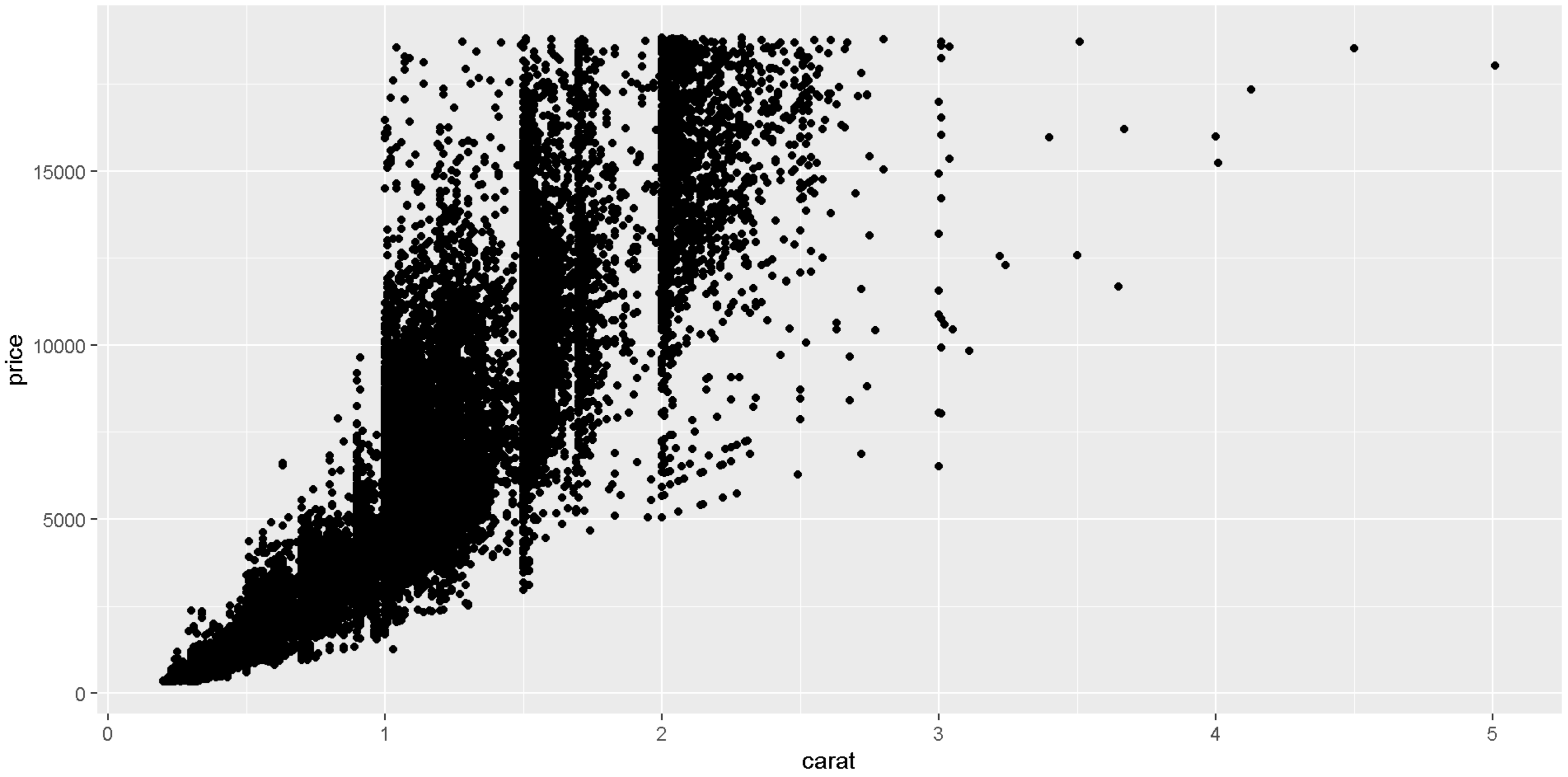


Seems more like there’s not many differences here!

EDA: Numeric and numeric variable associations

Scatterplot!

```
diamonds |>
  ggplot(aes(x = carat, y = price)) +
  geom_point()
```

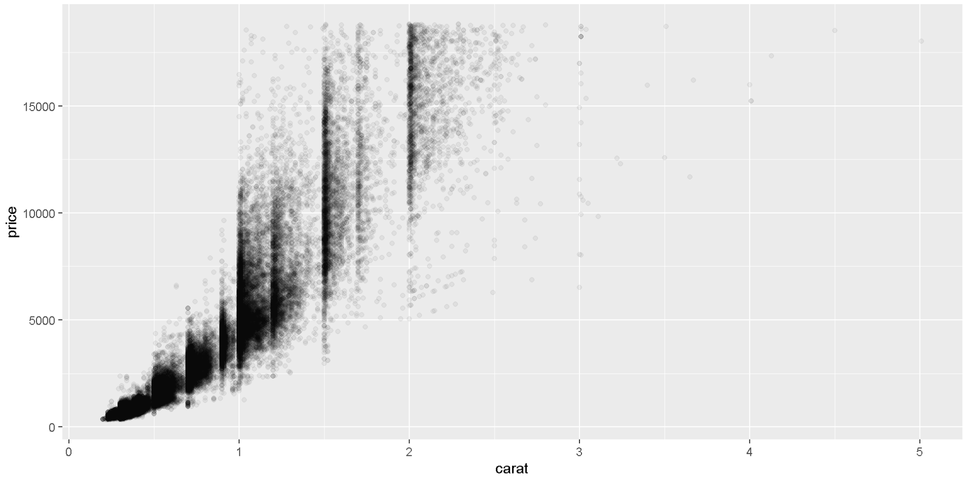


Hard to see the trends at the bottom?

EDA: Numeric and numeric variable associations

Scatterplot!

```r
diamonds |>
  ggplot(aes(x = carat, y = price)) +
  geom_point(alpha = .05)
```

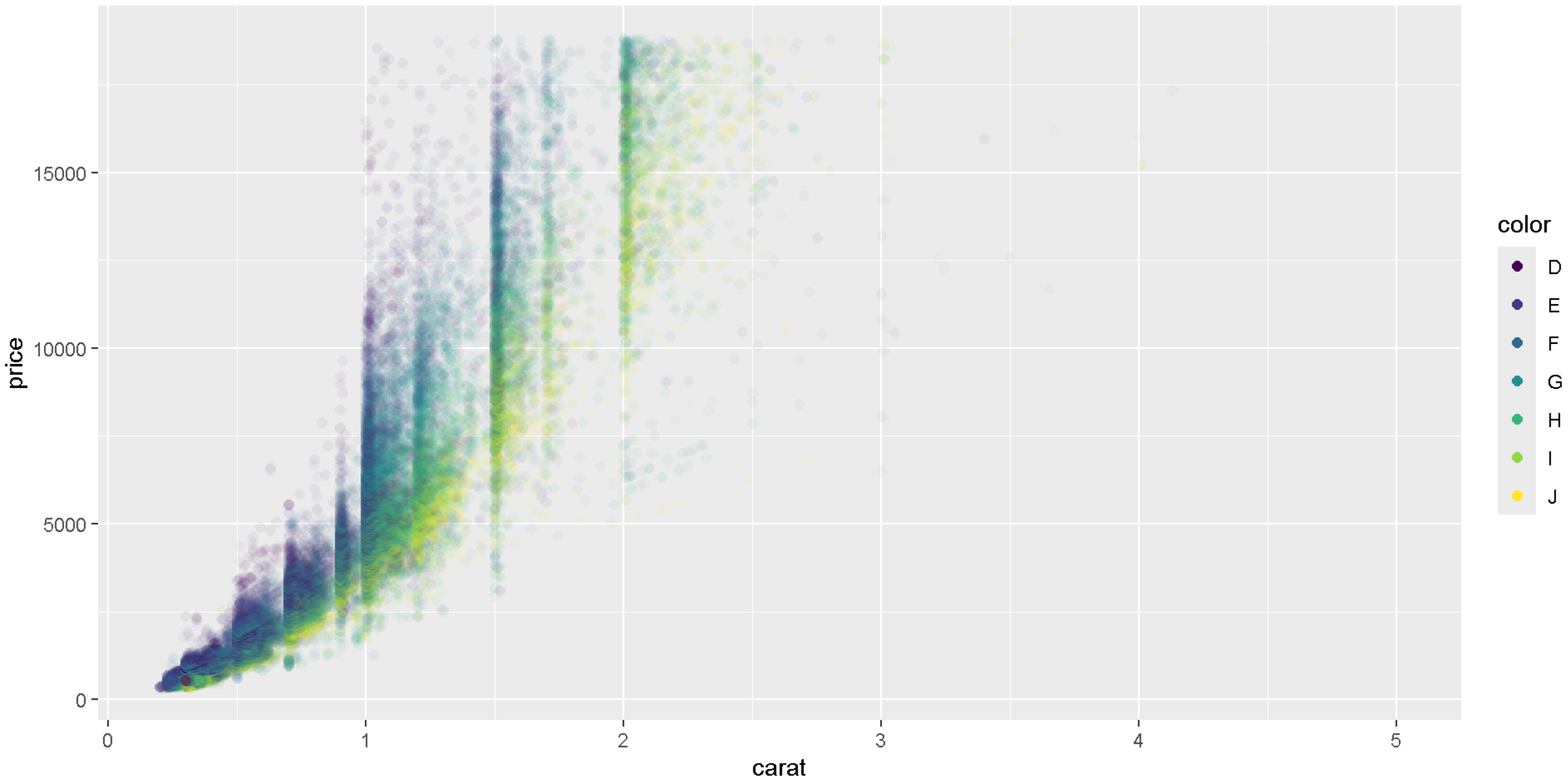


What’s that clustering? Any guesses?

EDA: Numeric and numeric variable associations

Scatterplot!

```r
diamonds |>
  ggplot(aes(x = carat, y = price, color = color)) +
  geom_point(alpha = .05, shape = 16,
             size = 2) +
  guides(color = guide_legend(override.aes = list(alpha = 1)))
```


price
15000
10000
5000
0
0
1
2
3
4
5
carat
color
D
E
F
G
H
I
J

Assignment 7