Lecture Assignment 9

Taiki Yamashita

2024-05-02

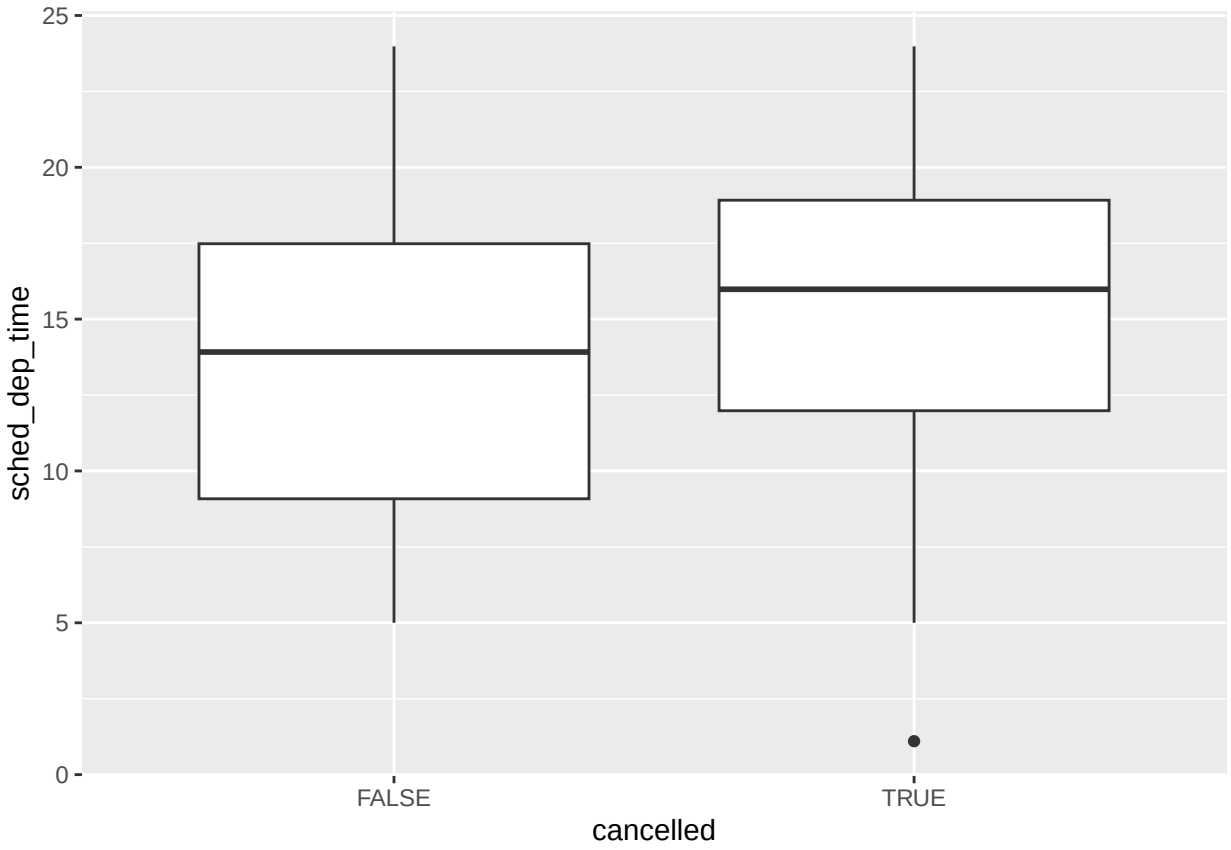
```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.0     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
##
## Attaching package: 'ggstance'
##
##
## The following objects are masked from 'package:ggplot2':
##
##     geom_errorbarh, GeomErrorbarh
```

7.5.1.1 1)

Use what you've learned to improve the visualisation of the departure times of cancelled vs. non-cancelled flights.

What we can do is instead of using a freqplot is that we can now use a box plot!

```
nycflights13::flights %>%
  mutate(
    cancelled = is.na(dep_time),
    sched_hour = sched_dep_time %/% 100,
    sched_min = sched_dep_time %% 100,
    sched_dep_time = sched_hour + sched_min / 60
  ) %>%
  ggplot() +
  geom_boxplot(mapping = aes(y = sched_dep_time, x = cancelled))
```



7.5.1.1 2) # What variable in the diamonds dataset is most important for predicting the price of a diamond? How is that variable correlated with cut? Why does the combination of those two relationships lead to lower quality diamonds being more expensive?

Because cut, color, and clarity are ordered categorical variables, I made an assumption that thye could be treated as continuous variables.

```
diamonds %>%
  mutate(cut = as.numeric(cut),
         color = as.numeric(color),
         clarity = as.numeric(clarity)) %>%
  select(price, everything()) %>%
  cor()
```

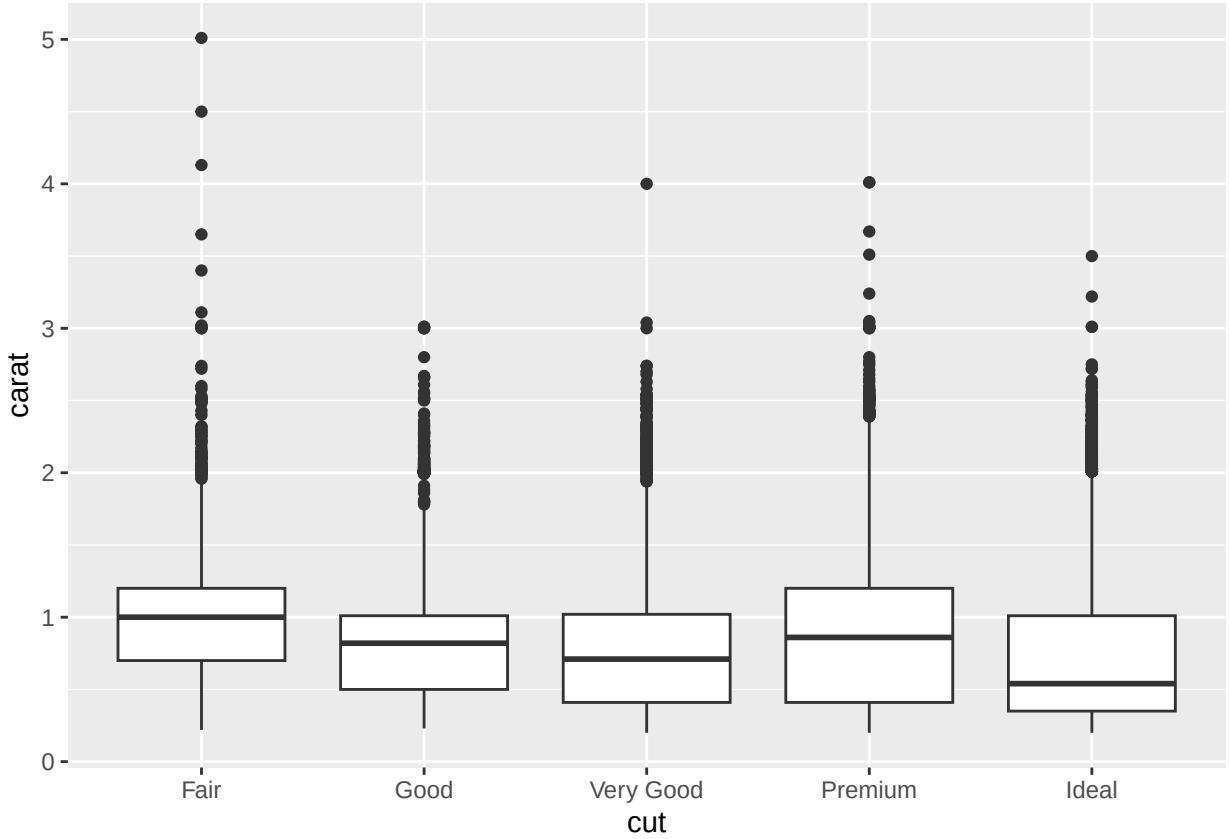
```
##               price       carat         cut       color     clarity       depth
## price    1.00000000  0.92159130 -0.05349066  0.17251093 -0.14680007 -0.01064740
## carat    0.92159130  1.00000000 -0.13496702  0.29143675 -0.35284057  0.02822431
## cut     -0.05349066 -0.13496702  1.00000000 -0.02051852  0.18917474 -0.21805501
## color    0.17251093  0.29143675 -0.02051852  1.00000000  0.02563128  0.04727923
## clarity -0.14680007 -0.35284057  0.18917474  0.02563128  1.00000000 -0.06738444
## depth   -0.01064740  0.02822431 -0.21805501  0.04727923 -0.06738444  1.00000000
## table    0.12713390  0.18161755 -0.43340461  0.02646520 -0.16032684 -0.29577852
## x        0.88443516  0.97509423 -0.12556524  0.27028669 -0.37199853 -0.02528925
## y        0.86542090  0.95172220 -0.12146187  0.26358440 -0.35841962 -0.02934067
## z        0.86124944  0.95338738 -0.14932254  0.26822688 -0.36695200  0.09492388
##              table           x           y           z
## price    0.1271339  0.88443516  0.86542090  0.86124944
## carat    0.1816175  0.97509423  0.95172220  0.95338738
```

```
## cut     -0.4334046 -0.12556524 -0.12146187 -0.14932254
## color    0.0264652  0.27028669  0.26358440  0.26822688
## clarity -0.1603268 -0.37199853 -0.35841962 -0.36695200
## depth   -0.2957785 -0.02528925 -0.02934067  0.09492388
## table    1.0000000  0.19534428  0.18376015  0.15092869
## x        0.1953443  1.00000000  0.97470148  0.97077180
## y        0.1837601  0.97470148  1.00000000  0.95200572
## z        0.1509287  0.97077180  0.95200572  1.00000000
```

carat is the most correlated variable with price, so it is the most important variable in predicting price of diamonds.

carat and cut are slightly negatively correlated. The diamonds of higher weights tend to have a lower cut rating. We can do...

```
ggplot(diamonds) +
  geom_boxplot(aes(x = cut, y = carat))
```



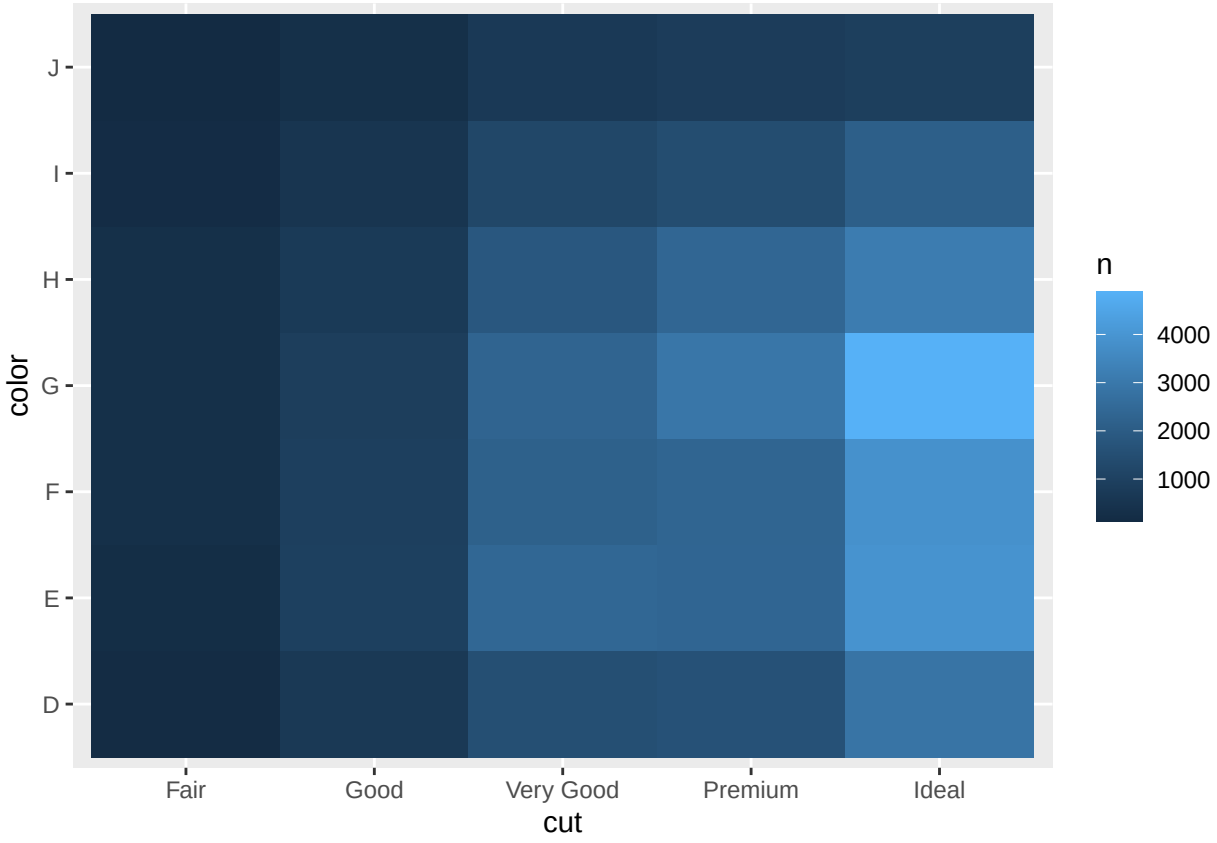
Because better cut has lower carat which makes their price lower, if we don't look at the carat, we can see that cut has lower price.

7.5.2.1 3)

Why is it slightly better to use aes(x = color, y = cut) rather than aes(x = cut, y = color) in the example above?

Usually it's better to use the categorical variable with a larger number of categories or the longer labels on the y axis. But, switching the order will not result in overlapping labels. Labels should be horizontal because it is easier to read.

```
diamonds %>%
  count(color, cut) %>%
  ggplot(mapping = aes(y = color, x = cut)) +
  geom_tile(mapping = aes(fill = n))
```

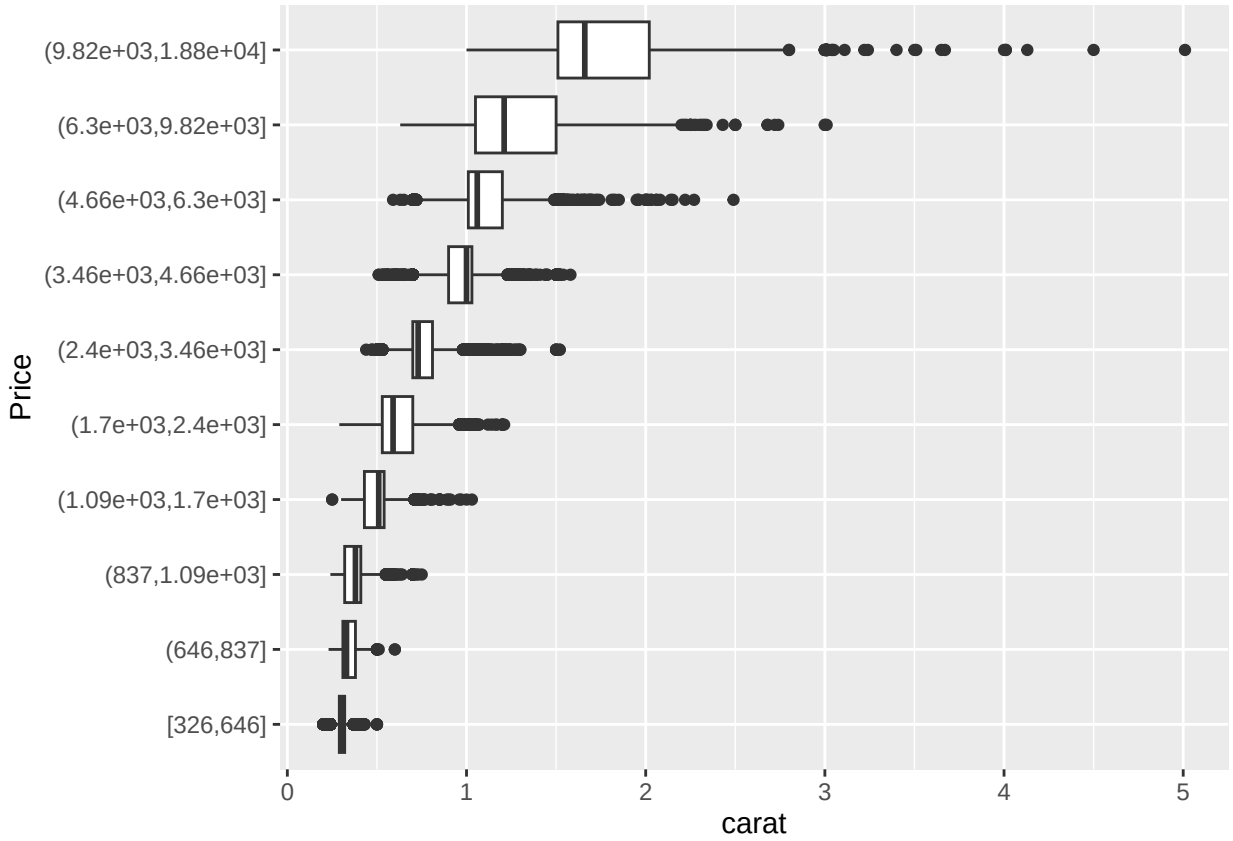


7.5.3.1 2)

Visualise the distribution of carat, partitioned by price.

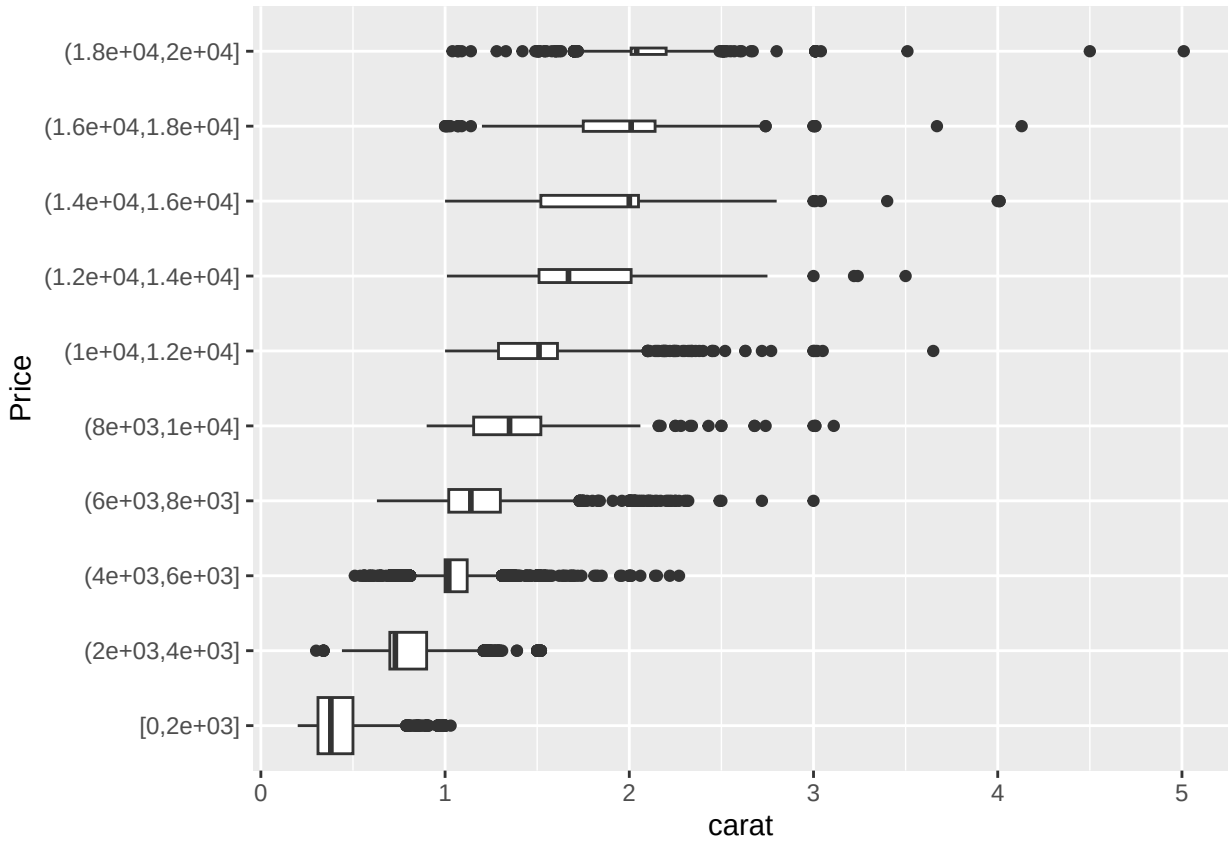
A graph of a box plot with 10 bins an equal number of observations. The width is determined by the number of observations.

```
ggplot(diamonds, aes(x = cut_number(price, 10), y = carat)) +
  geom_boxplot() +
  coord_flip() +
  xlab("Price")
```



Another visualization would be a box plot with 10 equal-width bins of 2000 dollars. boundary = 0 is what ensures that the first bin is 0 to 2000 dollars.

```
ggplot(diamonds, aes(x = cut_width(price, 2000, boundary = 0), y = carat)) +
  geom_boxplot(varwidth = TRUE) +
  coord_flip() +
  xlab("Price")
```

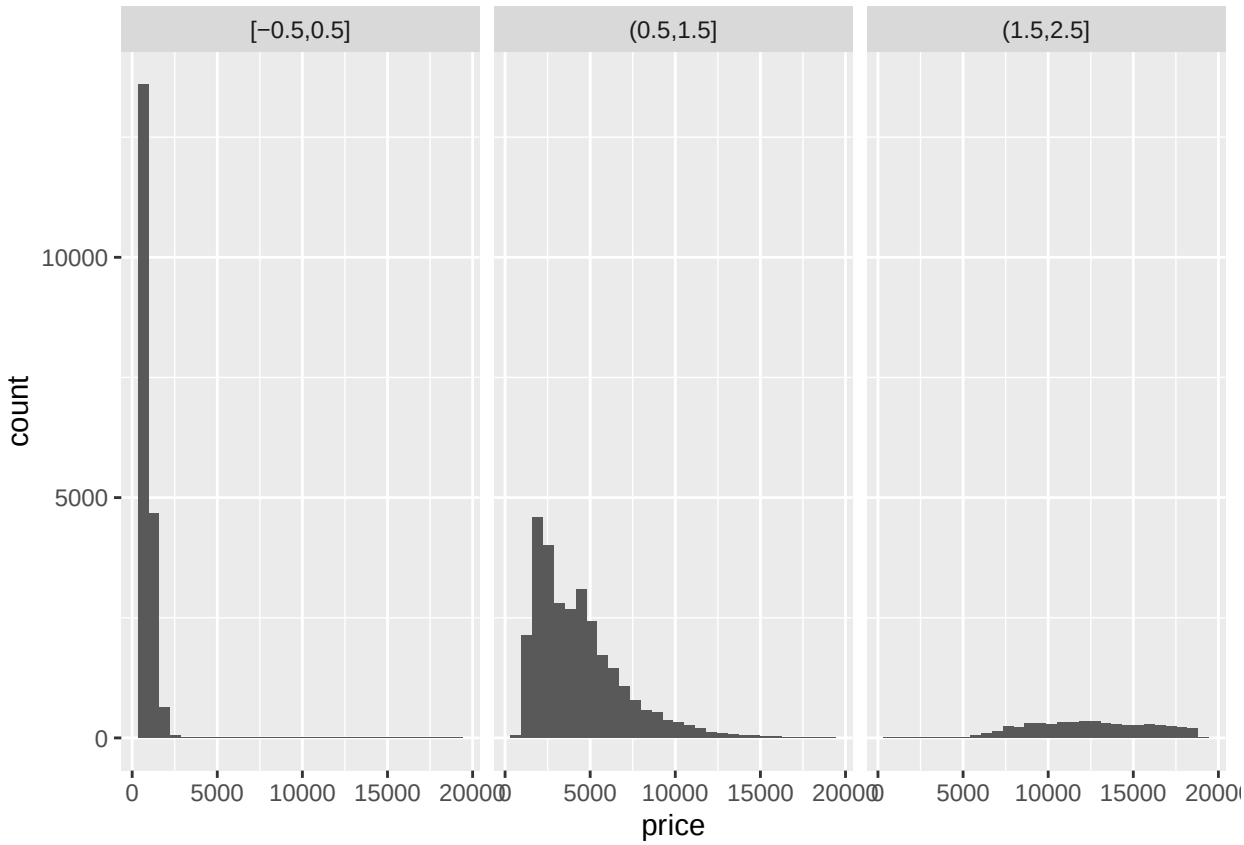


7.5.3.1 3)

How does the price distribution of very large diamonds compare to small diamonds? Is it as you expect, or does it surprise you?

```
diamonds %>%
  filter(between(carat, 0, 2.5)) %>%
  mutate(carat = cut_width(carat, 1)) %>%
  ggplot(aes(price)) +
  geom_histogram() +
  facet_wrap(~ carat)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

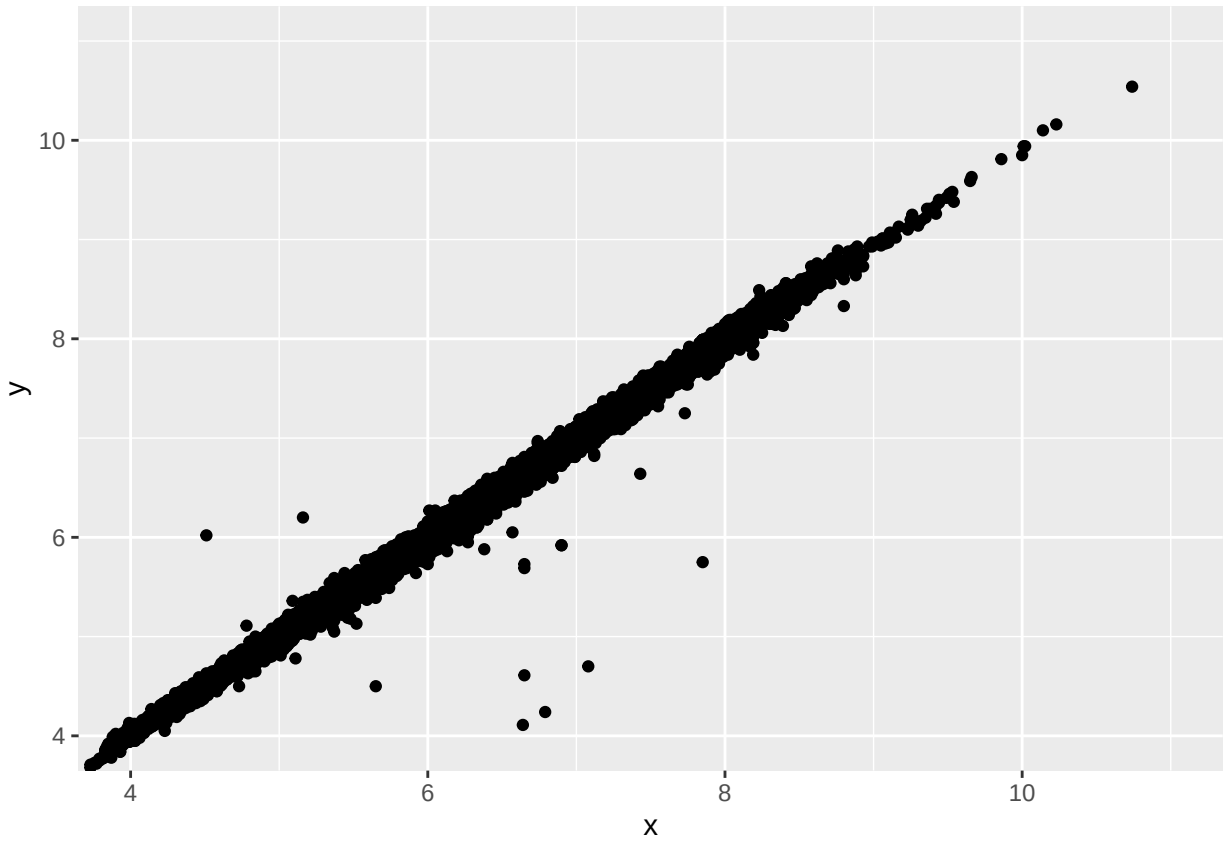


This was very surprising as I was expecting very little variance. It seems as though big diamonds can cost anything between 5000 and 18000. Whereas small ones have very little variance.

7.5.3.1 5)

Two dimensional plots reveal outliers that are not visible in one dimensional plots. For example, some points in the plot below have an unusual combination of x and y values, which makes the points outliers even though their x and y values appear normal when examined separately.

```
ggplot(data = diamonds) +
  geom_point(mapping = aes(x=x, y=y)) +
  coord_cartesian(xlim = c(4, 11), ylim = c(4, 11))
```



There is a strong relationship between x and y. The outliers are not as extreme in either x or y. A binned plot would not reveal these outliers, and may lead us to conclude that the largest value of x was an outlier even thogh it appears to fit the bivariate pattern well.