Homework Data Viz Batch 10

Jiruspak Franc

2024-07-22

Diamonds Data Analysis

Data preparation

```
library(ggplot2)
library(tidyverse)
```

```
## -- 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 lubridate 1.9.3     v tibble    3.2.1
## v purrr     1.0.2     v tidyr     1.3.1
## -- 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
```

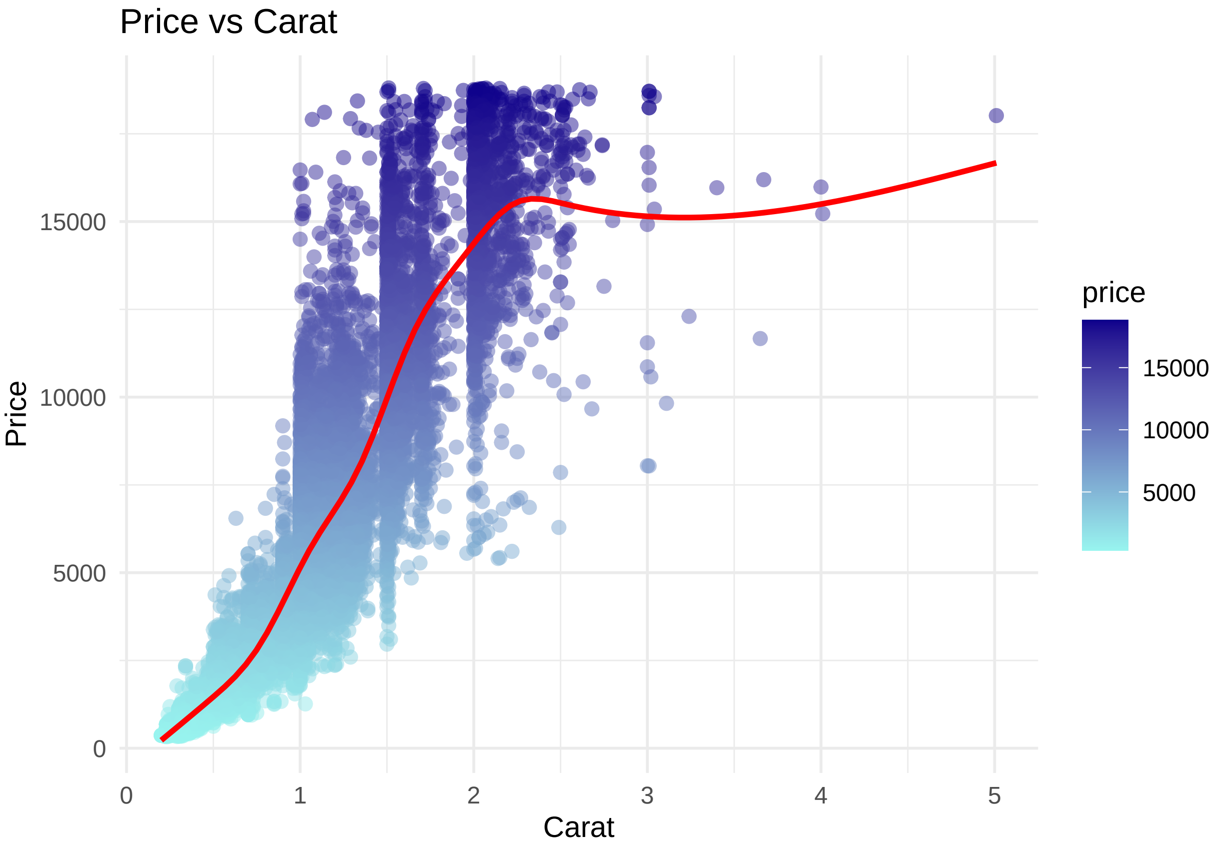
```
set.seed(42)
small_df <- diamonds %>%
  sample_frac(0.5)
tibble(small_df)
```

```
## # A tibble: 26,970 x 10
##    carat cut       color clarity depth table price     x     y     z
##    <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
##  1  0.39 Ideal     I     VVS2     60.8    56   849  4.74  4.76  2.89
##  2  1.12 Very Good G     SI2      63.3    58  4478  6.7   6.63  4.22
##  3  0.51 Very Good G     VVS2     62.9    57  1750  5.06  5.12  3.2
##  4  0.52 Very Good D     VS1      62.5    57  1829  5.11  5.16  3.21
##  5  0.28 Very Good E     VVS2     61.4    55   612  4.22  4.25  2.6
##  6  1.01 Fair      F     SI1      67.2    60  4276  6.06  6     4.05
##  7  0.4  Very Good D     VS1      60.8    59   954  4.74  4.76  2.89
##  8  0.9  Ideal     D     SI1      62.1    57  4523  6.18  6.25  3.86
##  9  0.33 Ideal     G     VVS1     62      55   838  4.45  4.49  2.77
## 10  0.71 Premium   G     VS2      62.1    62  2623  5.71  5.65  3.53
## # i 26,960 more rows
```

1. Distribution of price versus weight (Price vs Carat)

```
ggplot(small_df, aes(x = carat, y = price, color = price)) +
  geom_point(size = 2, pch = 19,  alpha = 0.5) +
  geom_smooth(se = FALSE, col = "red") +
  theme_minimal() +
    scale_color_gradient(low = "#98f5ef", high = "#0d018c") +
  labs(title = "Price vs Carat",
       x = "Carat",
       y = "Price")
```

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

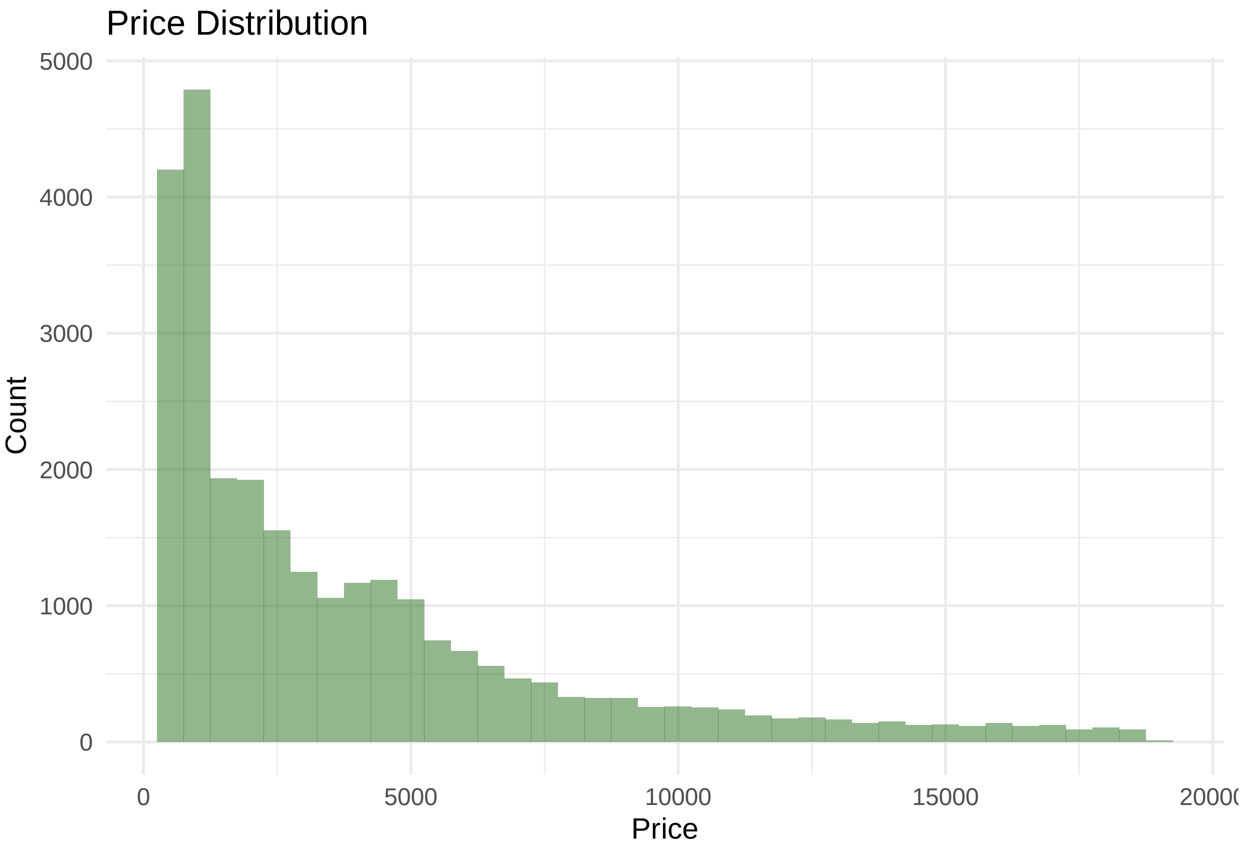


Analyze data from the graph

Price tends to increase with the weight of the diamond (Carat). However, there is variability in price at different weights, which may be due to other factors such as cut, clarity, and color.

2. Price distribution

```
ggplot(small_df, aes(x = price)) +
  geom_histogram(binwidth=500, fill = "#286e1f", alpha = 0.5) +
  theme_minimal() +
  labs(title = "Price Distribution",
       x = "Price",
       y = "Count")
```

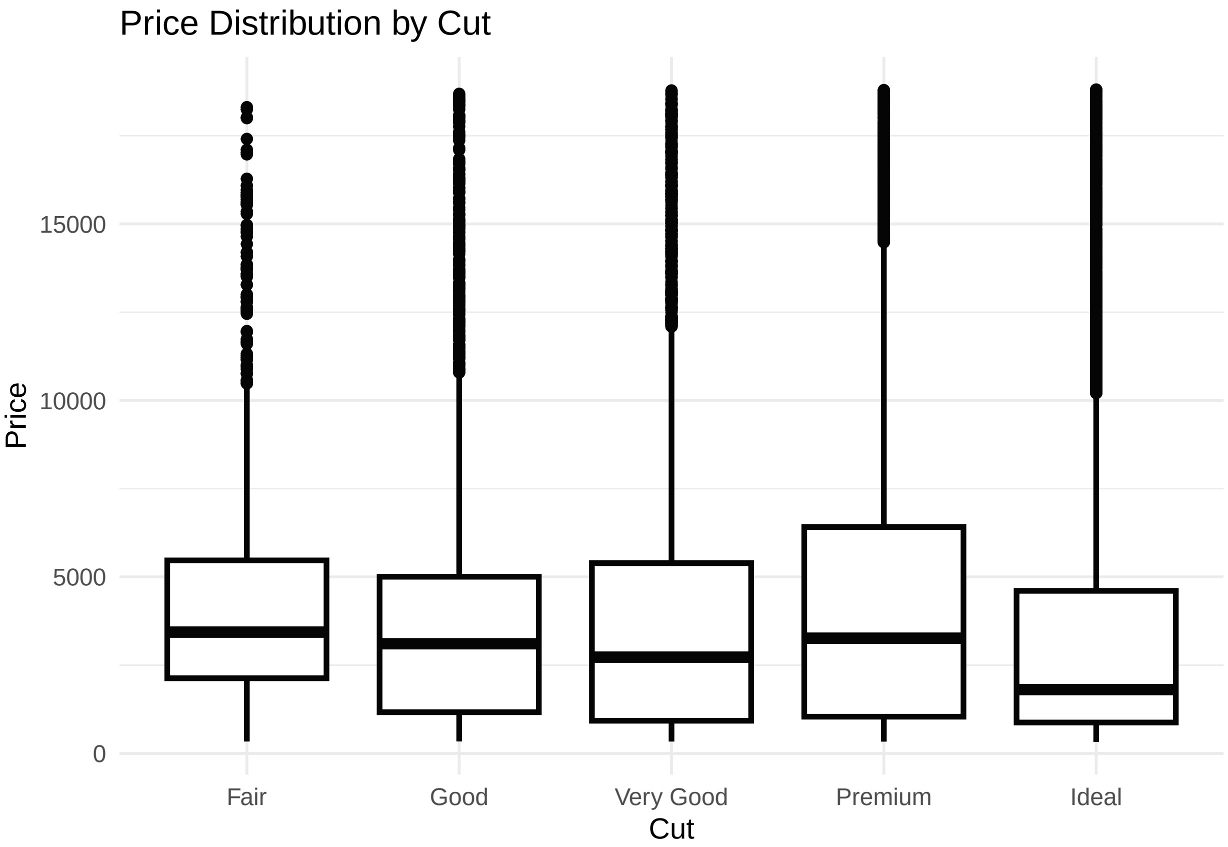


Analyze data from the graph

It shows that most diamond prices are in the low to mid-range (below $5,000), with high-priced diamonds being rare and representing a small portion of the data.

3. Price distribution by cut

```
ggplot(small_df, aes(x = cut, y = price)) +
  geom_boxplot(size = 1, col = "#050505") +
  theme_minimal() +
  labs(title = "Price Distribution by Cut",
       x = "Cut",
       y = "Price")
```

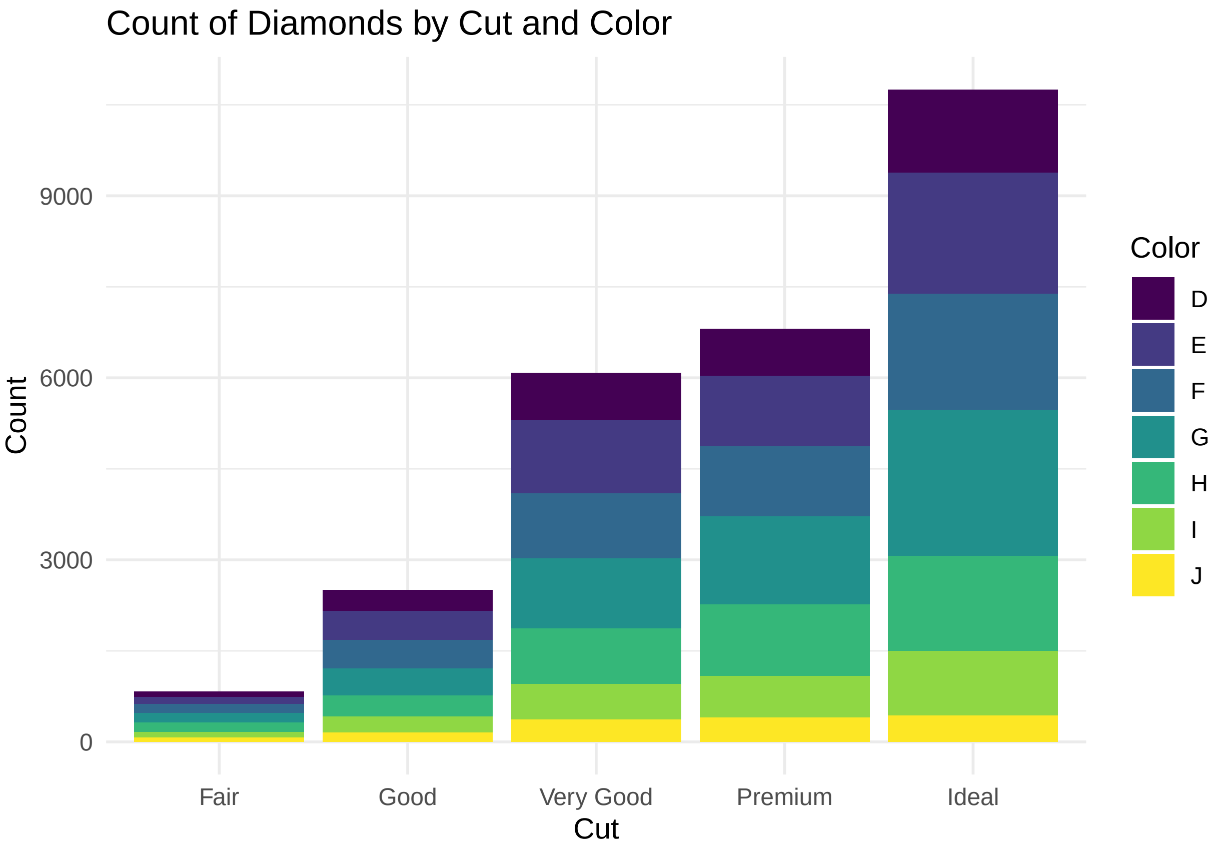


Analyze data from the graph

Better cuts (e.g., Ideal, Premium) are priced higher than lower cuts (e.g., Fair, Good). The price distribution within each cut category indicates variability within the same group.

4. Number of diamonds by cut type (Count of Diamonds by Cut)

```r
ggplot(small_df, mapping = aes(x = cut, fill = color)) +
  geom_bar(position = "stack") +
  theme_minimal() +
  labs(title = "Count of Diamonds by Cut and Color",
       x = "Cut",
       y = "Count",
       fill = "Color")
```

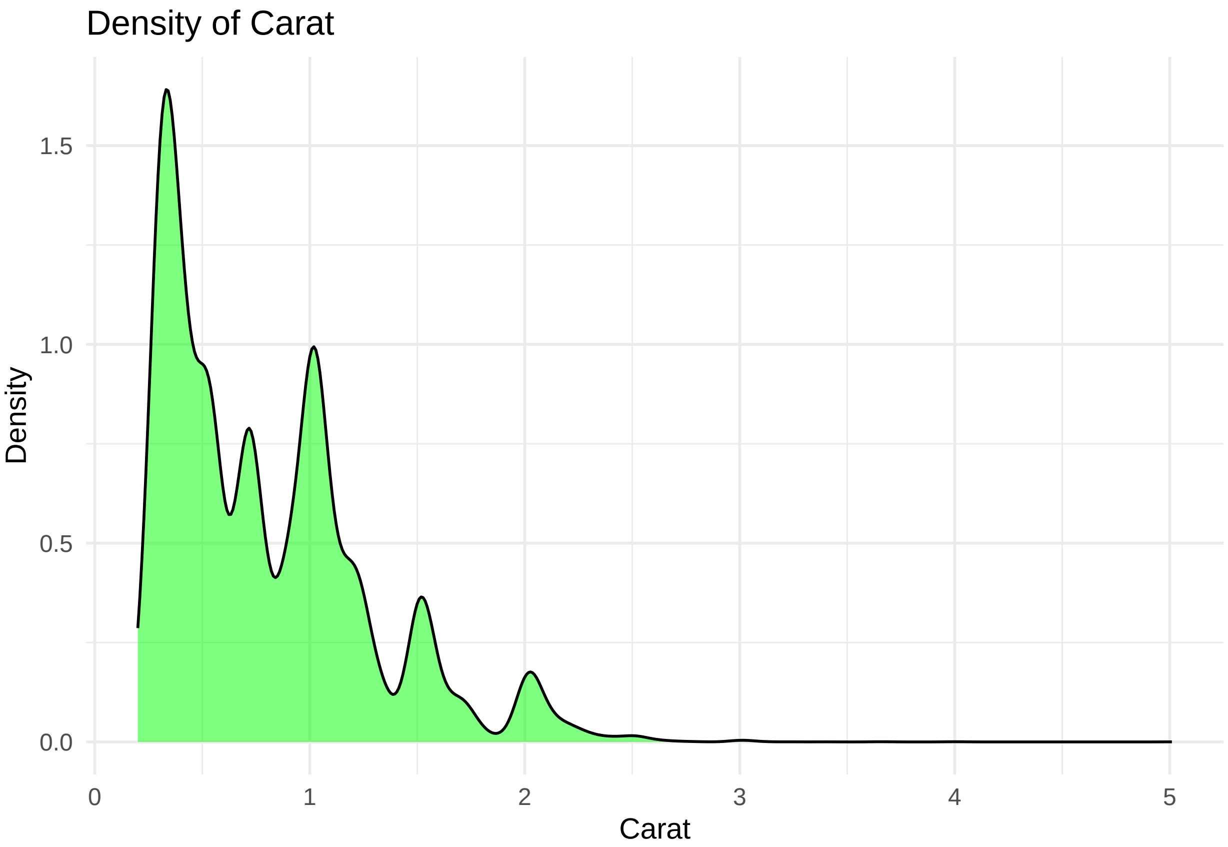


Analyze data from the graph

- Ideal: The number of diamonds with Ideal cuts is generally higher across all colors.
- Premium and Very Good: There are a high and similar number of diamonds in several colors.
- Good and Fair: The number of diamonds in these cut categories is lower and often shows high variability in each color.
- Most common colors: Colors G, H, and I are commonly found in various cut categories, especially in Ideal.
- Less common colors: Colors D and J tend to have lower distribution in certain cut categories, such as Fair or Good.

5. Density of diamond weight (Density of Carat)

```r
ggplot(small_df, aes(x = carat)) +
  geom_density(fill = "green", alpha = 0.5) +
  theme_minimal() +
  labs(title = "Density of Carat",
       x = "Carat",
       y = "Density")
```



Analyze data from the graph

The distribution of weight (Carat) shows that diamonds are commonly found in the range of approximately 0.2 to 1.0 carats. Diamonds with higher weights are less dense, indicating that there are fewer diamonds with higher weights.