

# Lista 6 [AD-UFPE-2019]

*Antonio Fernandes*

*20 de maio de 2019*

## Conteúdo

<b>Apresentação</b>	<b>1</b>
Questão I . . . . .	1
Questão II . . . . .	1
Questão III . . . . .	2
Questão IV . . . . .	2
Questão V . . . . .	2
Questão VI . . . . .	2
Questão VII . . . . .	2
Questão VIII . . . . .	2
Questão IX . . . . .	3
Visualização . . . . .	3
Identificando valores típicos . . . . .	7
Identificando outliers . . . . .	9
Valores ausentes ou NA's . . . . .	12
Variação . . . . .	13
Identificando padrões e modelos . . . . .	27
Questão X . . . . .	30
Questão XI . . . . .	32

## Apresentação

Este documento apresenta as respostas da lista de exercício 6 da disciplina de Análise de Dados.

Link github: <https://github.com/alvesat/lista6>

## Questão I

- Variável dependente é a variável que se busca explicar (variável explicativa);
- Variável independente é a variável (ou as variáveis) que explica (m) a variação na variável dependente;
- A relação existente entre a VD e a VI é que se busca explicar a VD por meio da VI utilizada.

## Questão II

A equação apresenta o modelo de regressão linear.

## Questão III

No modelo de regressão linear, o  $Y$  representa a variável dependente, o  $\alpha$  é o intercepto e apresenta o valor de  $Y$  quando  $X$  é igual a 0,  $\beta_1$  representa o coeficiente de regressão sendo este o valor da variação em  $Y$  de acordo com uma unidade de  $X$ . Por fim, o  $e$  representa o termo estocástico que é o erro em explicar  $Y$  a partir de  $X$ .

## Questão IV

O componente sistemático é representado pelo  $\hat{y} = \hat{\alpha} + \hat{\beta}X_1$  onde  $\hat{y}$  representa o valor predito de  $Y_1$ . Em suma, os valores que definem a linha de regressão são os componentes sistemáticos do modelo. Para cada valor de  $X$ , usa-se os valores de  $\hat{\alpha}$  e  $\hat{\beta}$  para encontrar o valor de  $\hat{y}$ .

## Questão V

O componente estocástico do modelo representa a diferença entre o valor observado da variável dependente e o seu valor predito pelo modelo. Ou seja, o componente estocástico representa os resíduos do modelo.

## Questão VI

A diferença entre  $Y$  e  $\hat{y}$  é que  $Y$  é o valor da variável dependente enquanto que  $\hat{y}$  é o valor predito da variável. Nesse sentido, o componente estocástico é resultado da diferença entre o valor observado e o valor predito da variável.

## Questão VII

O modelo de mínimos quadrados ordinários é um método de estimativação dos parâmetros em um modelo de regressão. O objetivo do modelo é minimizar o erro de estimativação na relação entre X e Y.

## Questão VIII

- Em relação ao nome dos arquivos, o ideal é que sejam nomes comprehensíveis e que terminem com .R;

Exemplo: *banco\_professores\_nordeste.R*

- Para os identificadores, o mais adequado é escrever o nome em minúsculo e separado por “\_” em vez de “\_” ou “-”;

Exemplo: *professores.pe*

- No que se refere ao estilo do código (*identitation*), a recomendação é utilizar dois espaços.

Exemplo :

```
media.vel <- mean(cars$speed)
```

- No tocante ao espaçamento, a recomendação é adicionar espaços entre todos os operadores binários.

Exemplo:

```
a <- 3 + 5
```

- Para atribuir valores a um elemento, o recomendado é utilizar “`<-`” em vez do “`=`”.

Exemplo:

```
c <- sqrt(4)
```

- Para realizar comentários no script, a recomendação é utilizar “`#`” seguido de um espaço.

Exemplo:

```
# Comentário do script x
```

- Em relação a função, a recomendação é que a primeira linha apresente os valores padrão e as outras linhas contenham os valores que não são os padrões.

Exemplo:

```
PredictCTR <- function(query, property, num.days,  
                      show.plot = TRUE)
```

- Para a documentação da função, as funções precisam conter uma seção de comentários logo após a linha de definição da função. Esses comentários devem apresentar os argumentos e a descrição do retorno da função.

## Questão IX

O exercício da questão 9 solicita que seja realizada a replicação dos exemplos apresentados no capítulo 7 do livro R para Ciências de Dados. Em suma, o capítulo apresenta um passo-a-passo do que é conhecido como análise exploratória de dados (*EDA*). Nesse caso, o propósito do capítulo é utilizar os processos de transformação e visualização de dados para responder e gerar perguntas com base nos dados.

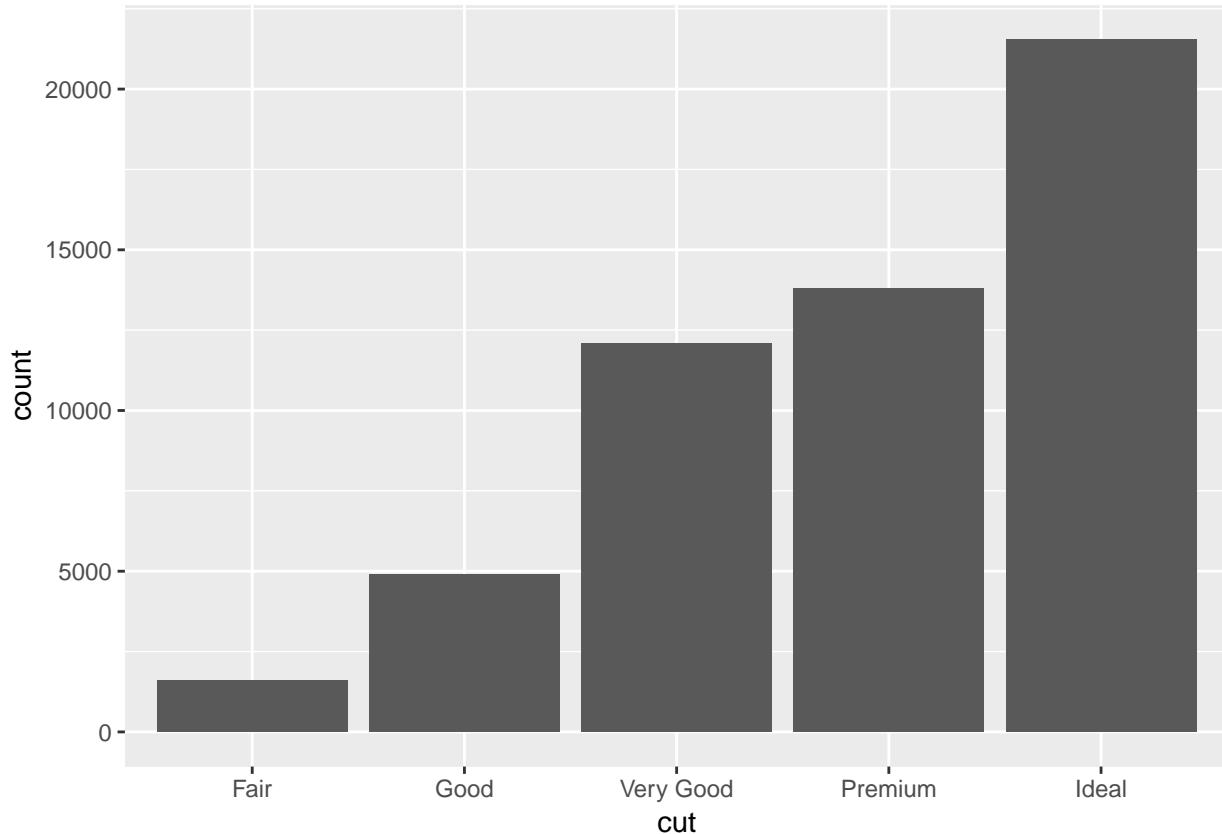
Na primeira parte do capítulo são apresentadas maneiras de visualizar distribuições. Para execução dos gráficos, é necessário o pacote `tidyverse` e `ggplot2`:

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

### Visualização

Primeiramente é apresentado a visualização envolvendo variáveis categóricas:

```
ggplot(data = diamonds) +  
  geom_bar(mapping = aes(x = cut))
```



Nesse caso, verificamos que para uma variável categórica, a ferramenta mais apropriada para visualização é um gráfico de barras. Podemos verificar a quantidade de casos para cada categoria presente no eixo  $X$

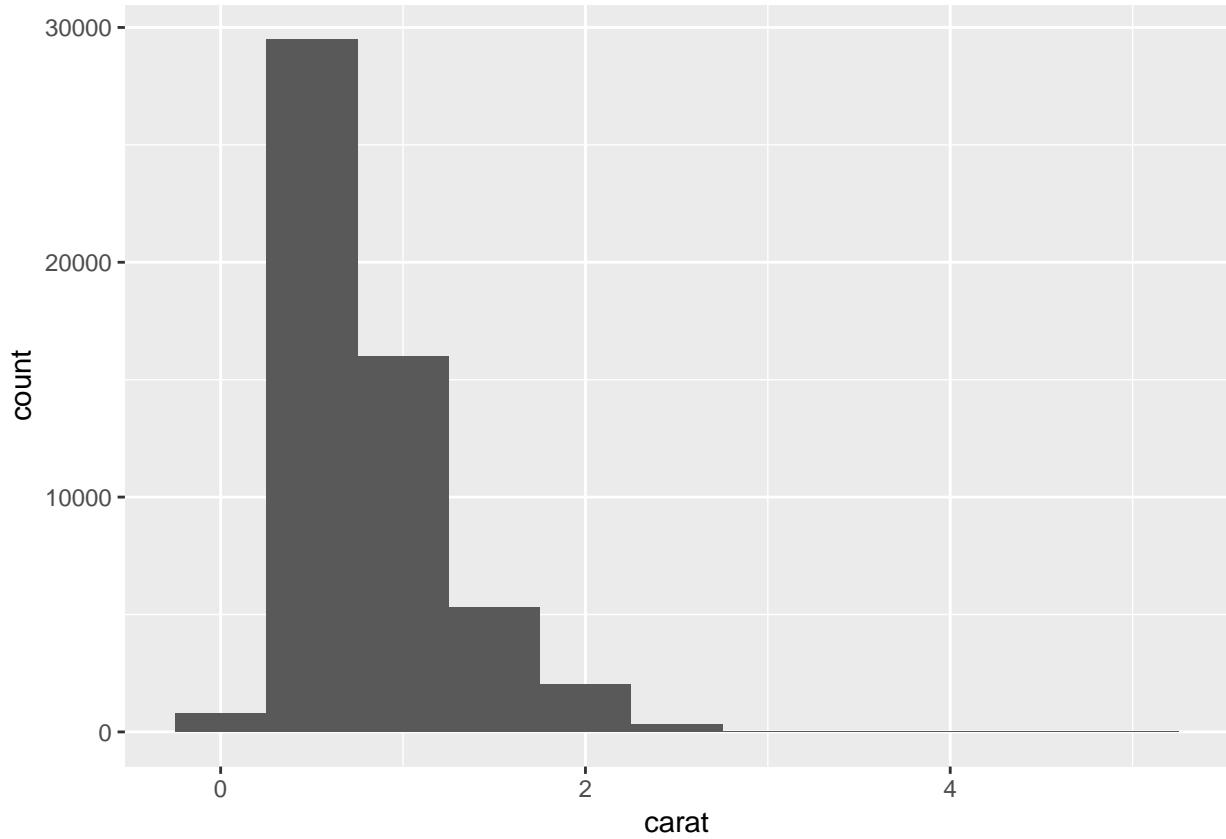
Também é possível visualizar por meio de uma tabela a frequência de cada coluna:

```
diamonds %>%
  count(cut)
```

```
## # A tibble: 5 x 2
##   cut      n
##   <ord>    <int>
## 1 Fair     1610
## 2 Good     4906
## 3 Very Good 12082
## 4 Premium   13791
## 5 Ideal    21551
```

Quando a variável em questão é uma variável contínua, o mais apropriado é visualizar a distribuição por meio de um histograma:

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



Por se tratar de uma variável contínua,  $n$  valores podem estar presentes em um intervalo (tenha como exemplo a variável tempo). Por meio do histograma obtemos uma melhor visualização desse tipo de dado.

Do mesmo modo que é possível criar uma tabela com a frequência de cada coluna de uma variável categórica, também é possível fazer isso com uma variável contínua:

```
diamonds %>%
  count(cut_width(carat, 0.5))
```

```
## # A tibble: 11 x 2
##   `cut_width(carat, 0.5)`     n
##   <fct>                  <int>
## 1 [-0.25,0.25]            785
## 2 (0.25,0.75]           29498
## 3 (0.75,1.25]           15977
## 4 (1.25,1.75]            5313
## 5 (1.75,2.25]            2002
## 6 (2.25,2.75]             322
## 7 (2.75,3.25]              32
## 8 (3.25,3.75]                5
## 9 (3.75,4.25]                4
## 10 (4.25,4.75]               1
## 11 (4.75,5.25]               1
```

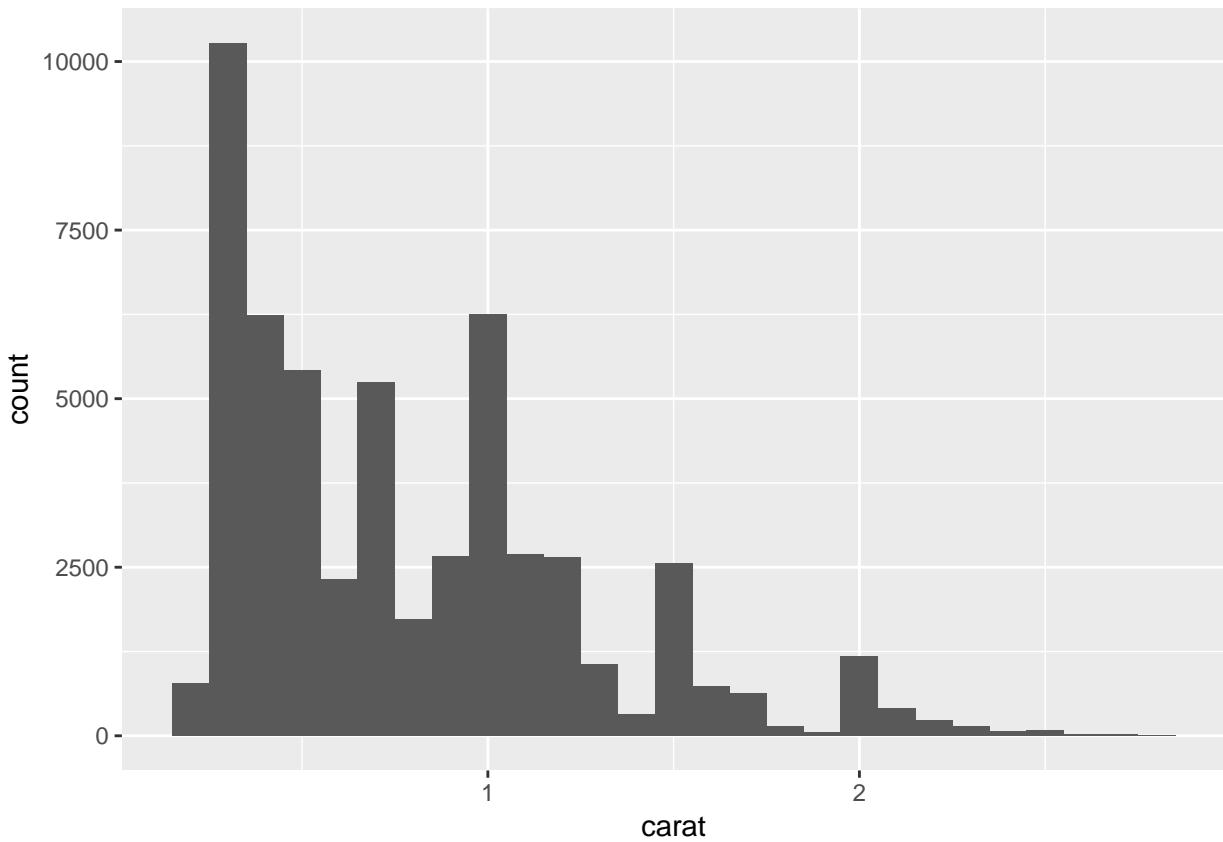
Ao observar a tabela, percebemos que o próprio R faz uma divisão entre escalas. Nesse caso, também é possível acrescentar no gráfico esse intervalo:

```

smaller <- diamonds %>%
  filter(carat < 3)

ggplot(data = smaller, mapping = aes(x = carat)) +
  geom_histogram(binwidth = 0.1)

```



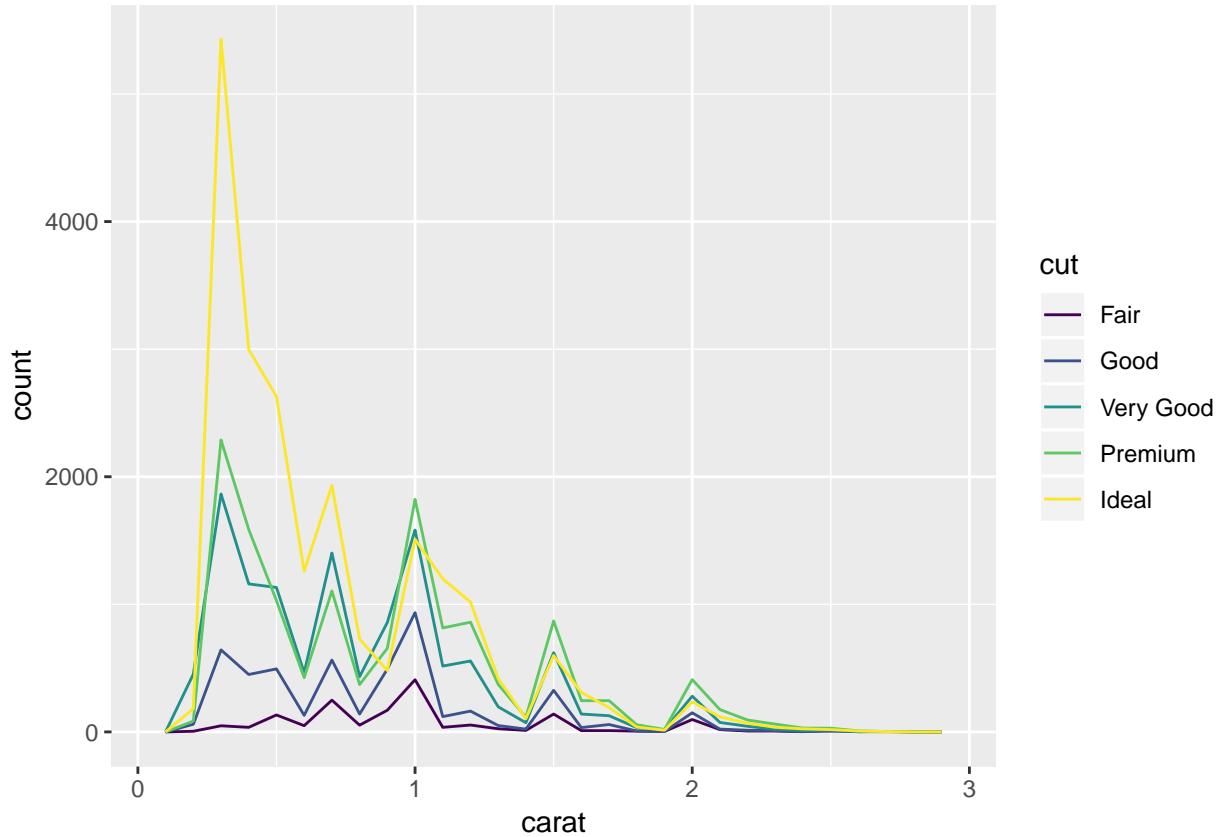
Ou seja, agora temos um histograma com valores agregados a cada 0.1 unidade do eixo x.

Também é apresentado no capítulo como apresentar vários histogramas em um mesmo gráfico. Para isso, é recomendado que se utilize um gráfico com linhas, permitindo uma melhor visualização do que seria cada coluna. No gráfico a seguir, existem diversos histogramas de acordo com cada categoria da variável *cut* e cada cor de linha representa uma categoria da variável:

```

ggplot(data = smaller, mapping = aes(x = carat, colour = cut)) +
  geom_freqpoly(binwidth = 0.1)

```

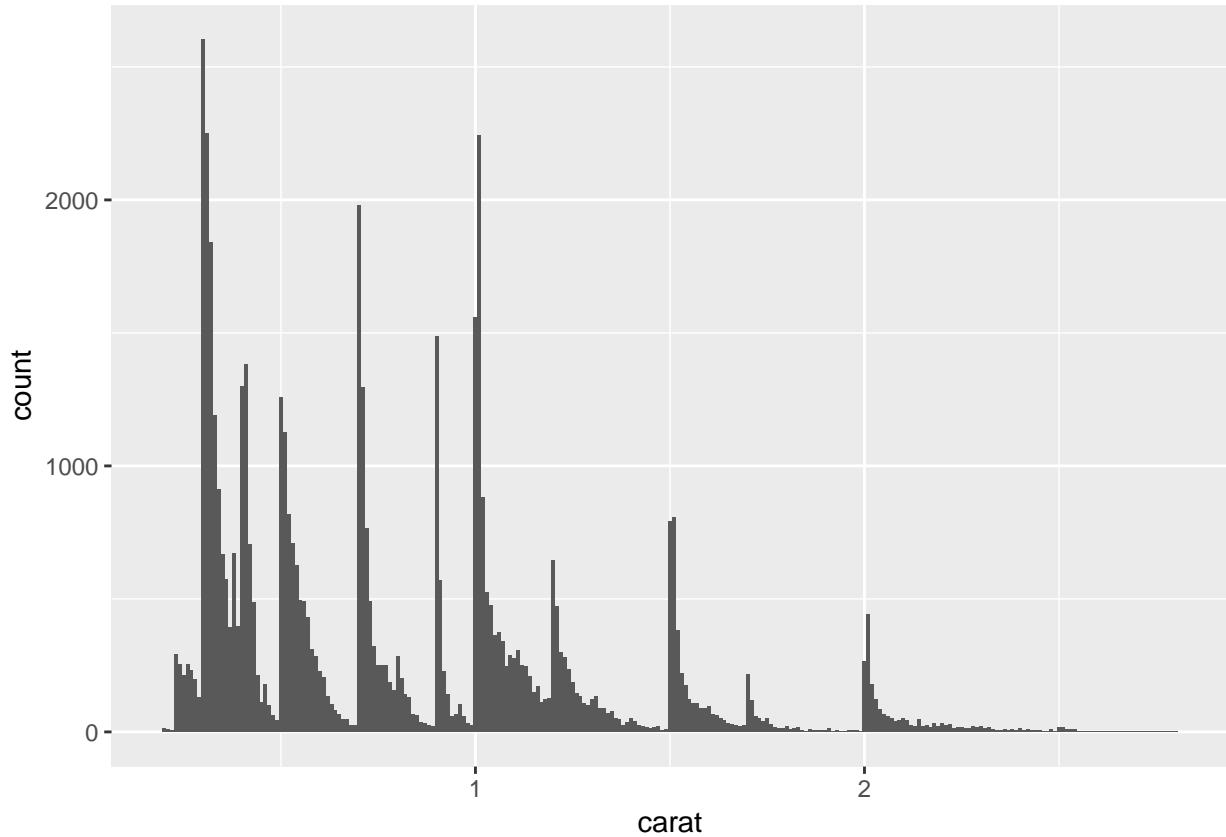


### Identificando valores típicos

Outro ponto fundamental no processo de análise exploratória de dados apresentado no capítulo envolve a identificação dos valores de uma distribuição. Isso busca responder perguntas como: Quais os valores mais comuns? ou Quais os valores mais distoantes da minha variável?

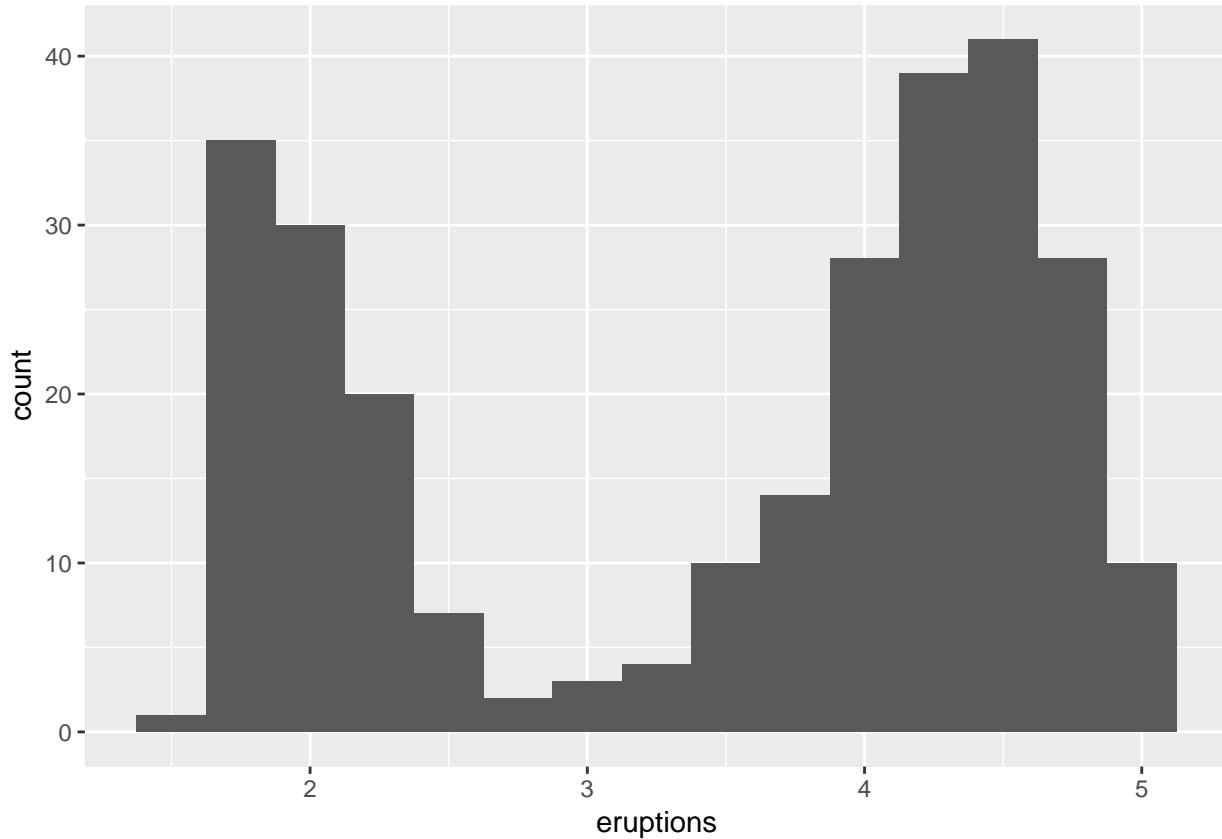
Primeiro é então apresentado um histograma de um banco envolvendo diamantes:

```
ggplot(data = smaller, mapping = aes(x = carat)) +
  geom_histogram(binwidth = 0.01)
```



O que é possível perceber ao analisar o gráfico é que existem grupos específicos na distribuição (*cluster*) e nenhum diamante com quilate (*carat*) maior que 3. Outro exemplo é apresentado envolvendo um banco com erupções e o tempo que estas duram:

```
ggplot(data = faithful, mapping = aes(x = eruptions)) +  
  geom_histogram(binwidth = 0.25)
```



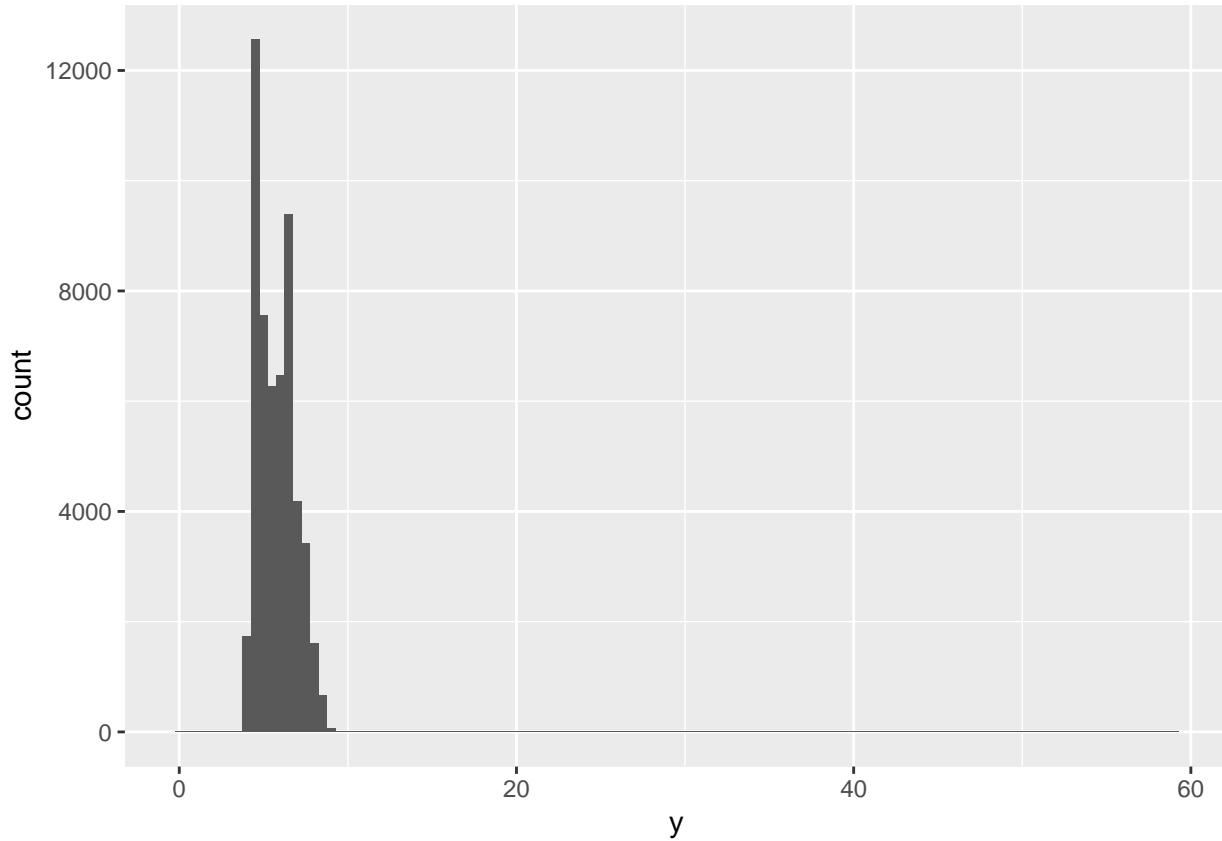
Podemos constatar então a presença de dois grupos no gráfico (uma distribuição bimodal). Ou seja, subgrupos na nossa distribuição envolvendo agrupamentos específicos em relação ao tempo da erupção.

### Identificando outliers

Um fator extremamente importante ao visualizar os seus dados é buscar verificar se existem ou não dados que destoam do resto da sua distribuição. Esses valores podem influenciar nos resultados dos modelos e análises posteriores.

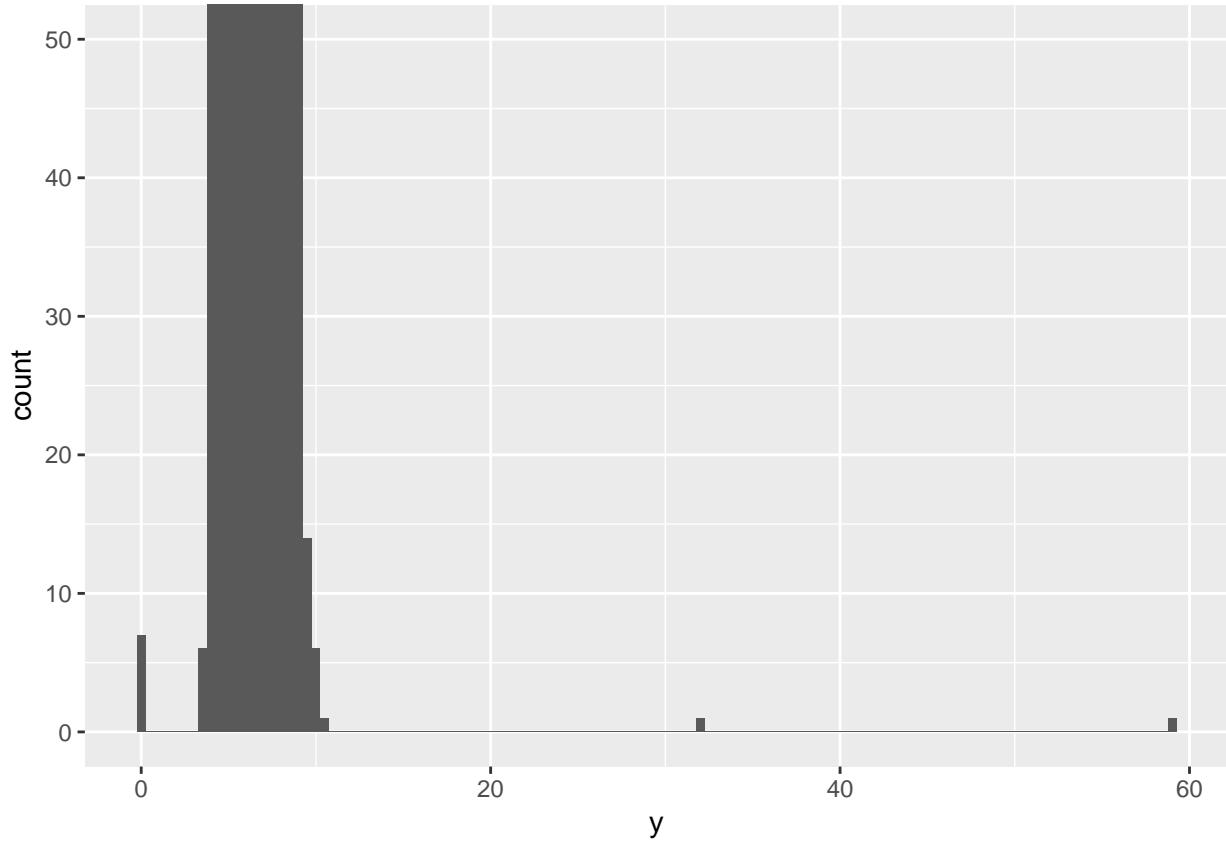
Com isso, o autor apresenta um exemplo envolvendo a identificação de casos destoantes:

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



Nesse primeiro gráfico, vemos o histograma mas não identificamos valores extremos na nossa distribuição. Qual pode ser o motivo? O intervalo do eixo Y do gráfico, que permite um alto número de observações em apenas uma coluna. Vamos ver o que acontece ao mudarmos os limites do eixo Y:

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



Agora, aplicamos uma espécie de *zoom* nos nossos dados e podemos perceber a presença de alguns valores que estão distantes da nossa distribuição ou destoam do padrão visualizado. Por meio do uso do `filter`, podemos remover esses valores:

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

```
## # A tibble: 9 x 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
```

Nesse caso, pegamos os valores maiores que 20 e menores que 3 e elaboramos um novo *data.frame* para visualização destes casos.

## Valores ausentes ou NA's

Ao analisar um determinado conjunto de dados, é possível se deparar com o problema de valores ausentes ou a necessidade de remover *outliers* da distribuição. Dito isso, o autor apresenta algumas soluções para esse clássico problema ao realizar o tratamento dos dados.

Primeiramente, é possível eliminar os valores extremos:

```
diamonds2 <- diamonds %>%
  filter(between(y, 3, 20))
```

Outra solução possível (e mais recomendada pelo autor) é substituir os valores por NA:

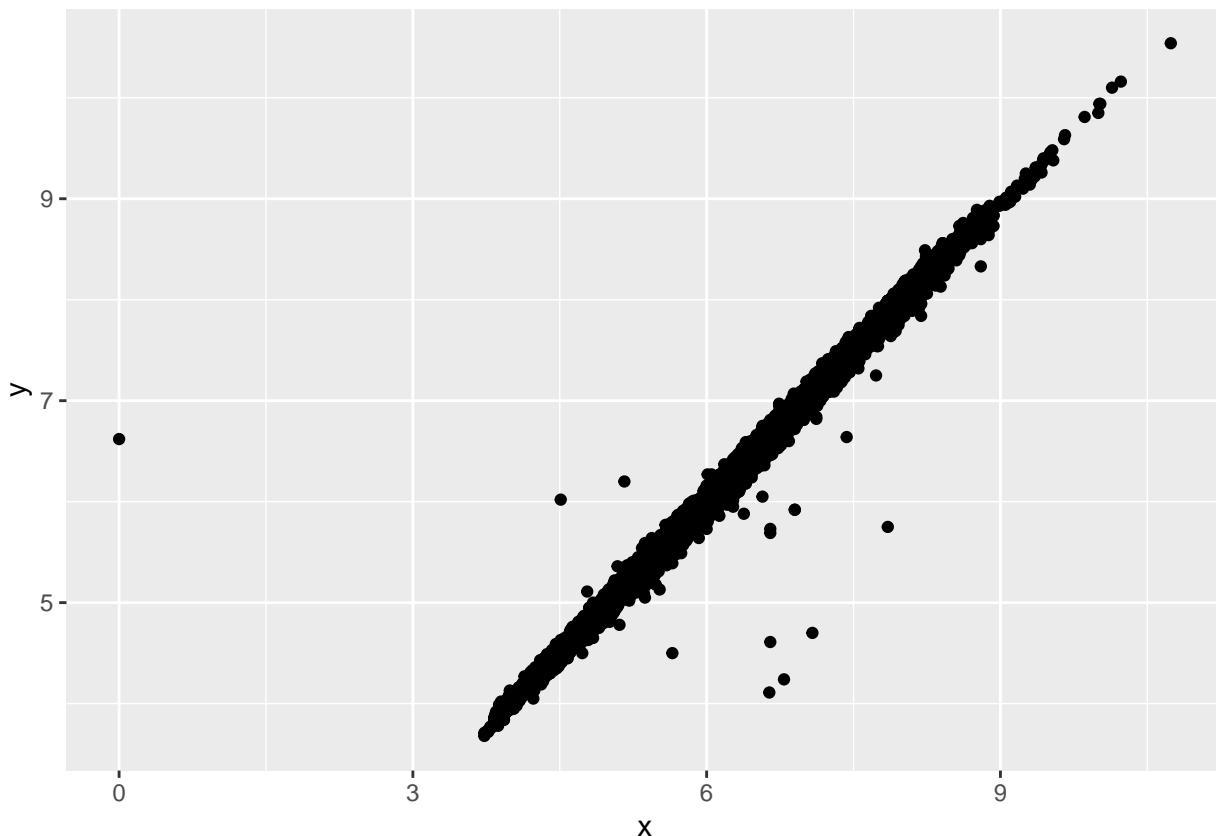
```
diamonds2 <- diamonds %>%
  mutate(y = ifelse(y < 3 | y > 20, NA, y))
```

Nesse caso, os valores de Y menores que 3 e maiores que 20 foram substituídos por NA na distribuição por meio do comando *ifelse*.

Ao realizar um gráfico no ggplot de uma variável contendo 'NA's', o ggplot avisa que esses valores não foram incluídos no gráfico:

```
ggplot(data = diamonds2, mapping = aes(x = x, y = y)) +
  geom_point()
```

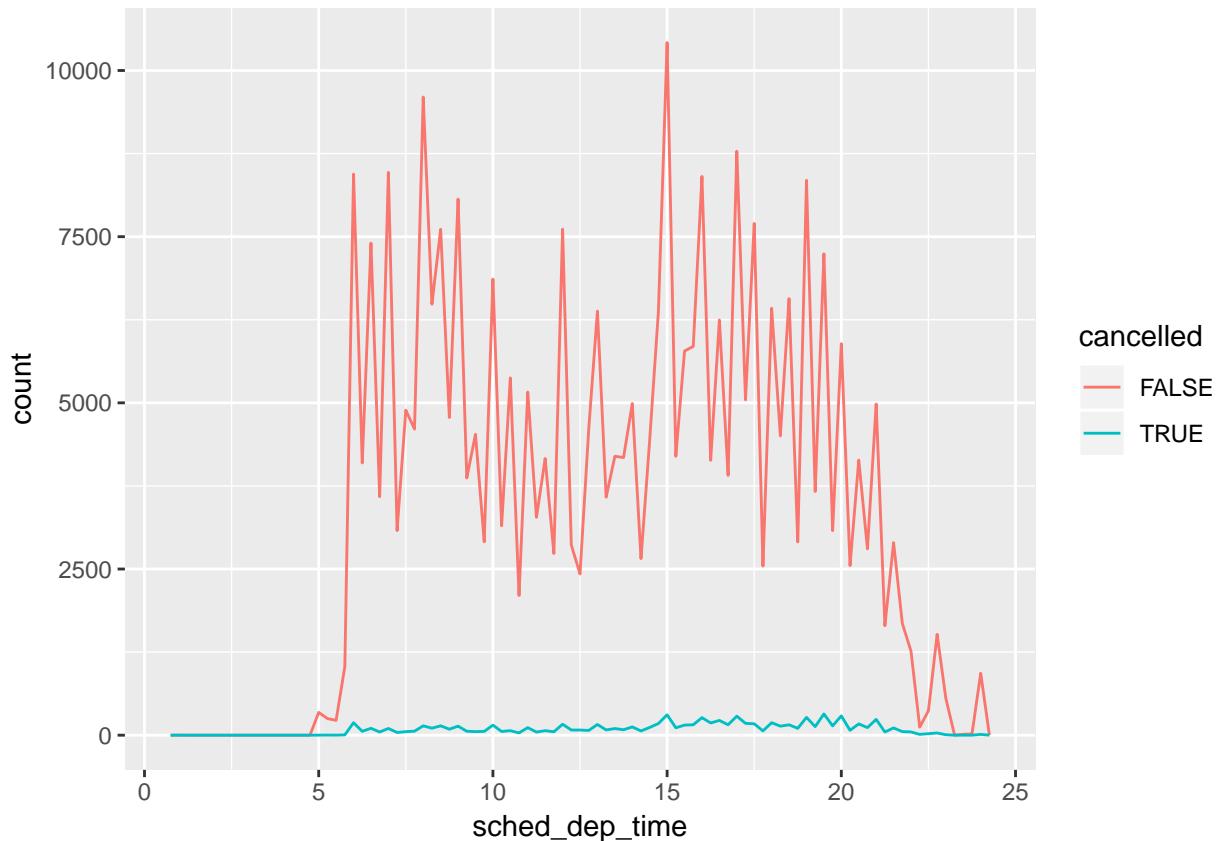
```
## Warning: Removed 9 rows containing missing values (geom_point).
```



Caso queira remover o aviso, altera-se a linha `geom_point()` para: `geom_point(na.rm = TRUE)`

Também é apresentado pelo autor como é possível comparar os casos ausentes no banco com os casos presentes. Em um banco envolvendo voos, existem casos que foram cancelados. É então elaborado um gráfico que contém ambas as informações:

```
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(mapping = aes(sched_dep_time)) +
  geom_freqpoly(mapping = aes(colour = cancelled), binwidth = 1/4)
```



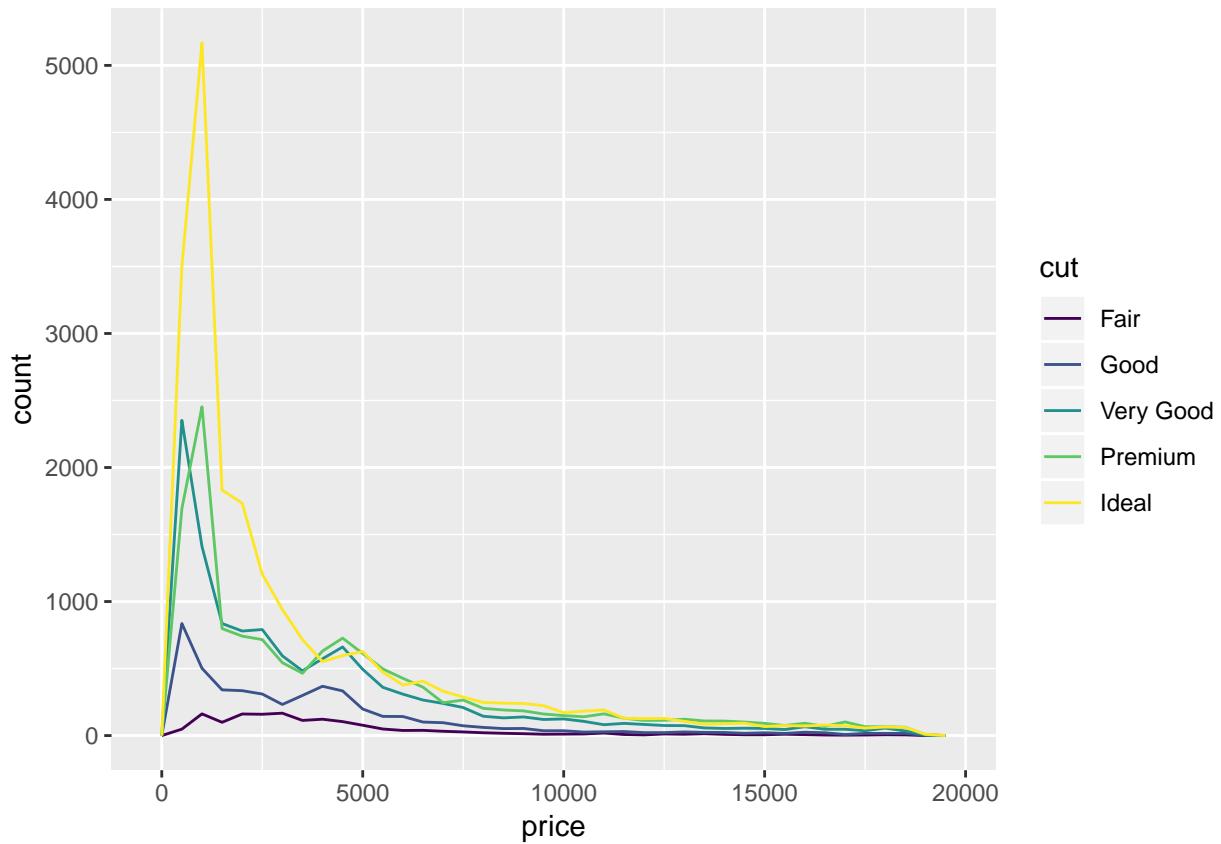
No gráfico, observamos o uso do comando `mutate` que modifica as informações do banco criando *labels* mais ajustados. Nesse caso: `cancelled` representa os casos ausentes da variável `dep_time`.

## Variação

Mais um ponto importante apresentado no capítulo é relacionado a análise de variações de uma ou mais variáveis. Um exemplo é verificar a distribuição de uma variável categórica de acordo com cada categoria da variável.

Nesse caso, é apresentado a variação do preço de um diamante de acordo com a sua qualidade:

```
ggplot(data = diamonds, mapping = aes(x = price)) +
  geom_freqpoly(mapping = aes(colour = cut), binwidth = 500)
```

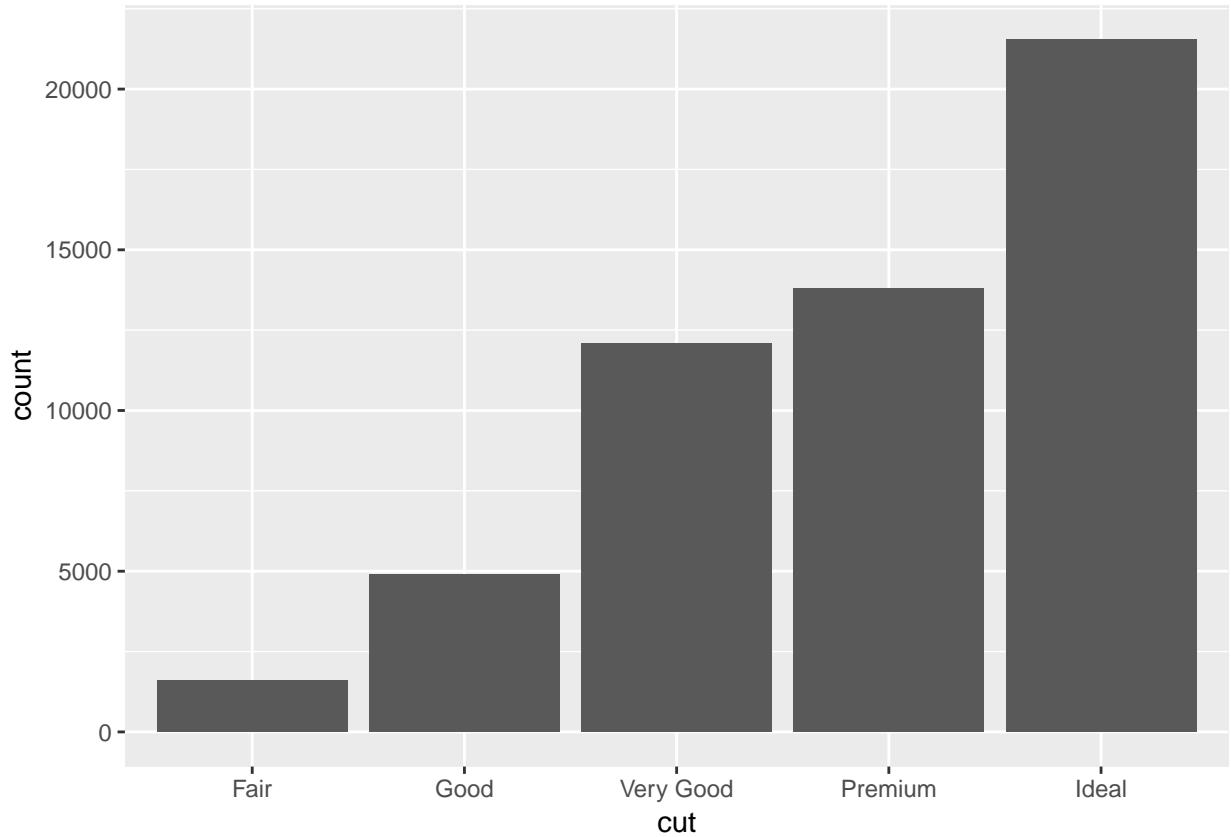


```
geom_freqpoly(mapping = aes(colour = cancelled), binwidth = 1/4)
```

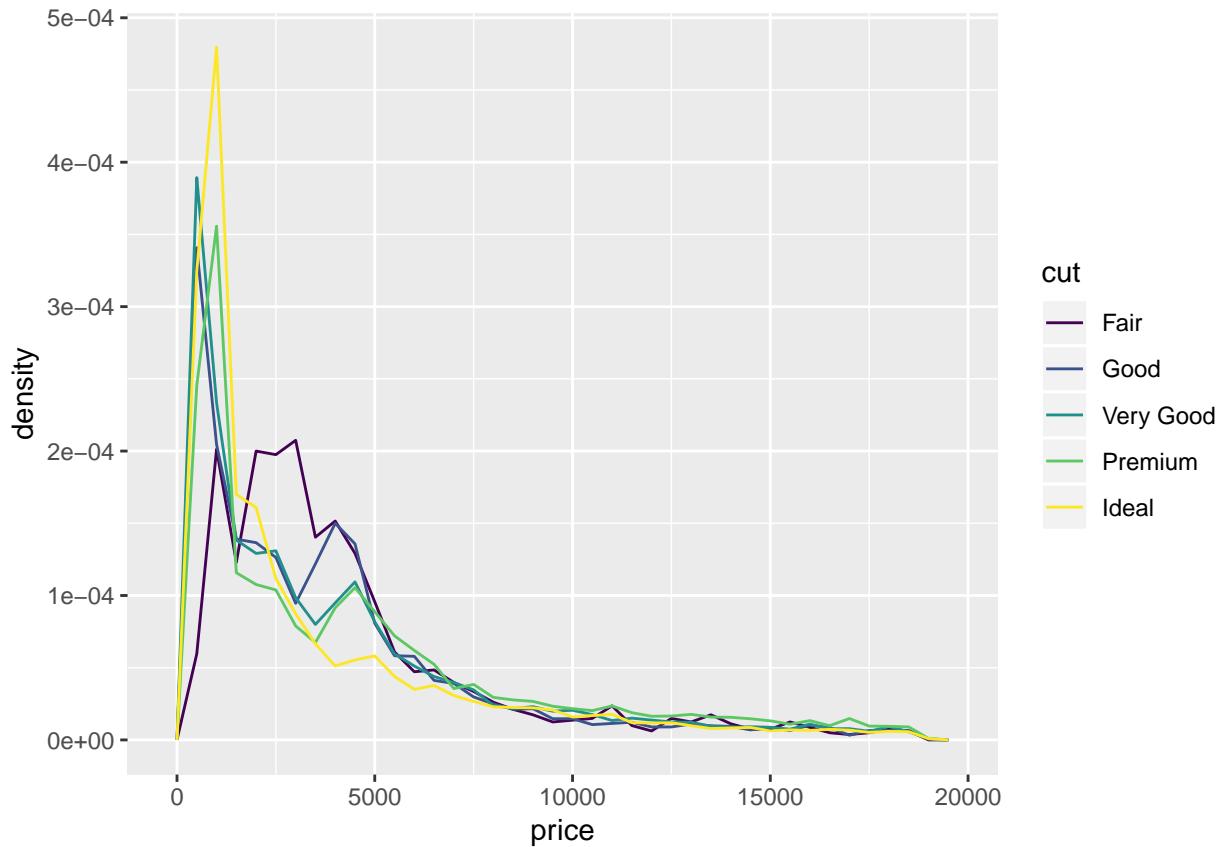
```
## mapping: colour = ~cancelled
## geom_path: na.rm = FALSE
## stat_bin: na.rm = FALSE, binwidth = 0.25, pad = TRUE
## position_identity
```

Nesse gráfico, a visualização acaba ficando comprometida devido a sobreposição de linhas. Outras alternativas são apresentadas, como um gráfico de barras e um gráfico de densidade:

```
ggplot(diamonds) +
  geom_bar(mapping = aes(x = cut))
```

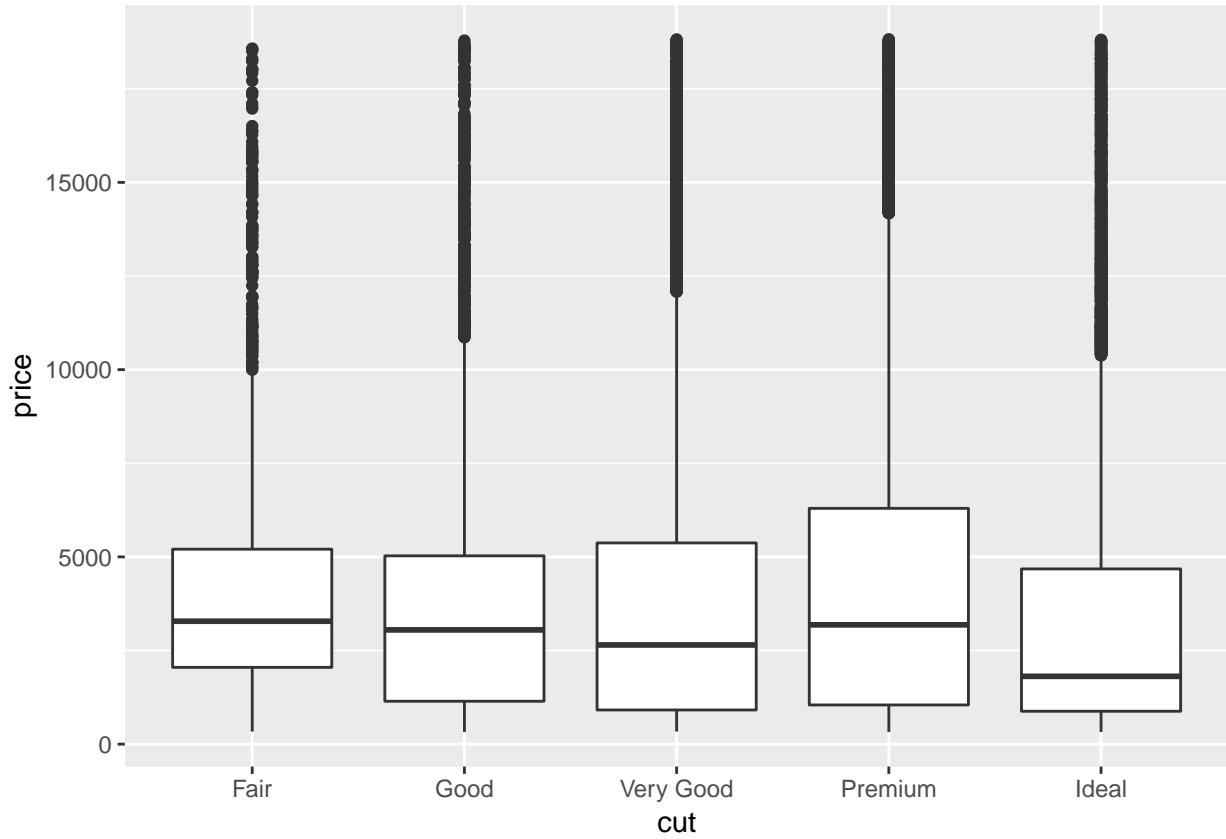


```
ggplot(data = diamonds, mapping = aes(x = price, y = ..density..)) +  
  geom_freqpoly(mapping = aes(colour = cut), binwidth = 500)
```



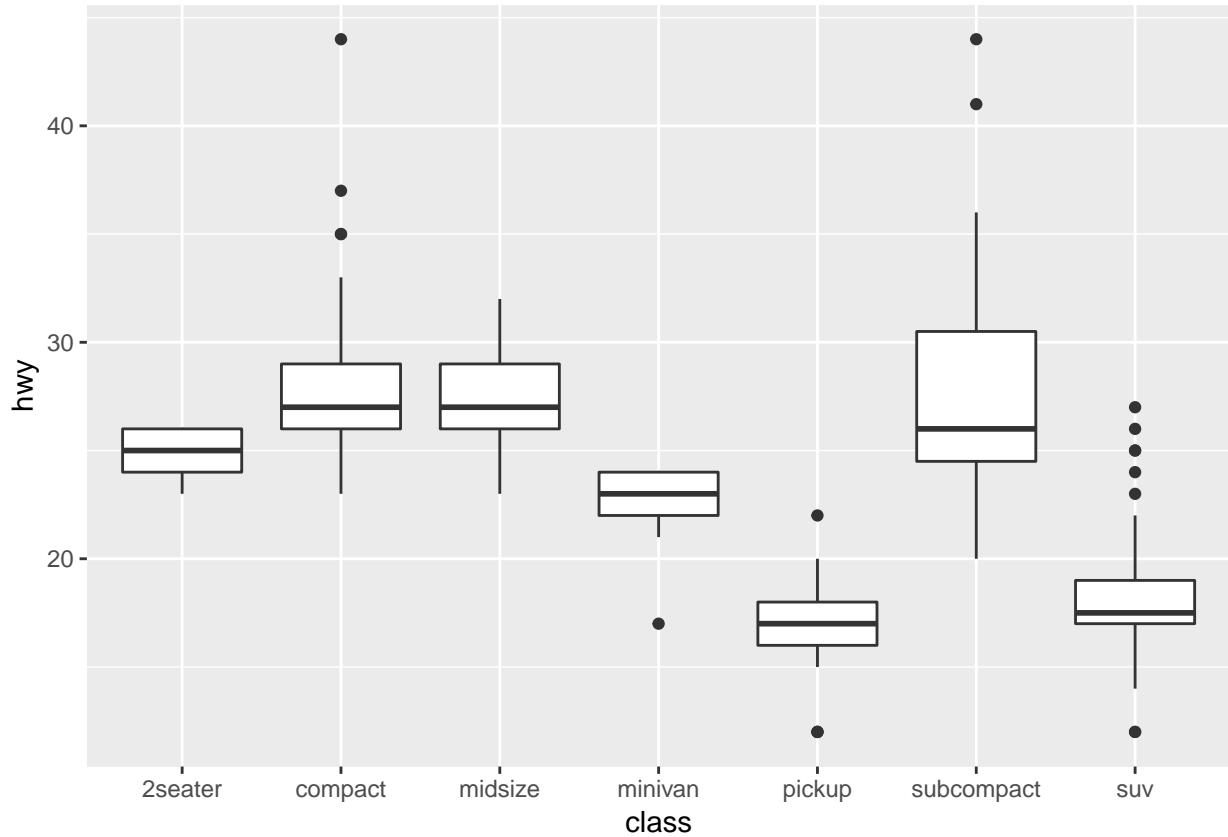
Ainda assim, a visualização não é a melhor!! O mais apropriado então é visualizar a relação entre qualidade do diamante e preço por meio de um boxplot:

```
ggplot(data = diamonds, mapping = aes(x = cut, y = price)) +
  geom_boxplot()
```



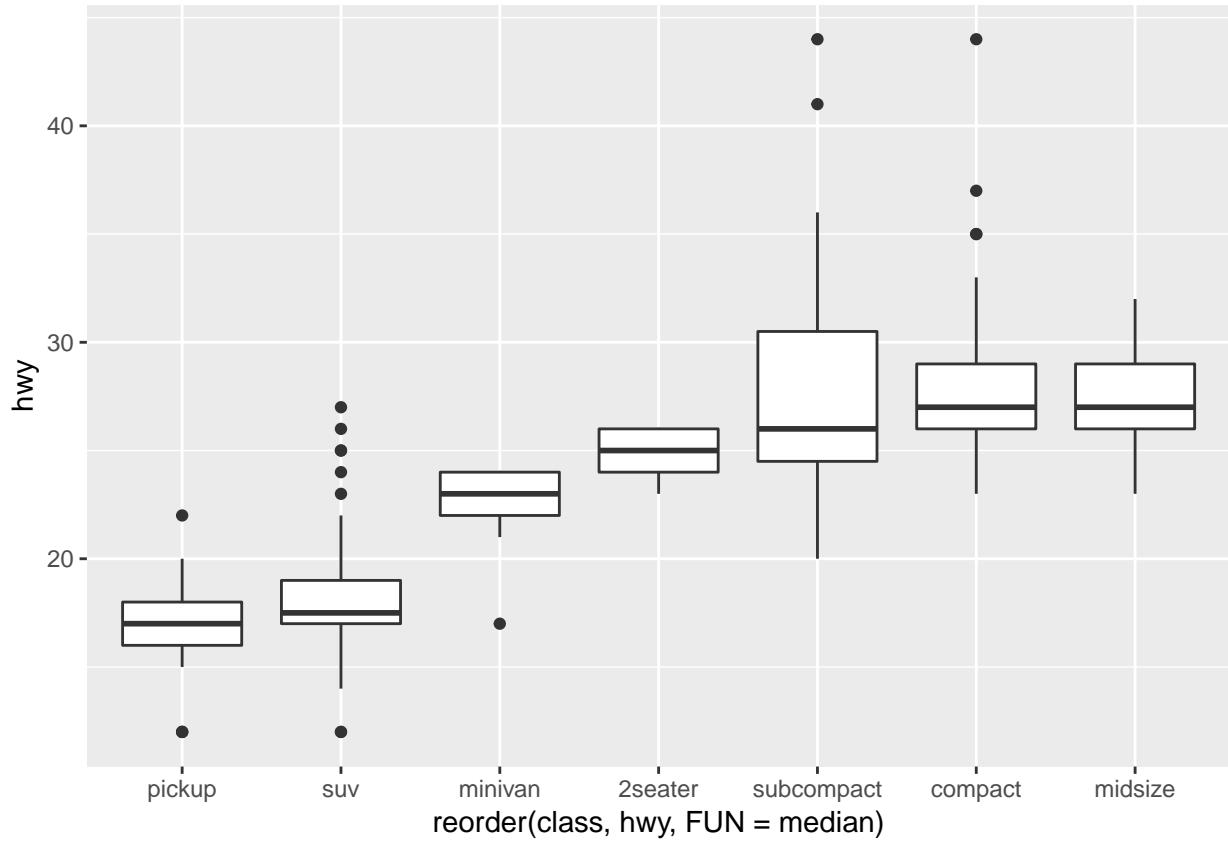
Apesar de apresentar menos informações, o boxplot permite uma melhor comparação entre as categorias. Outro exemplo é apresentado a relação entre a média de km/h e modelo do carro:

```
ggplot(data = mpg, mapping = aes(x = class, y = hwy)) +  
  geom_boxplot()
```



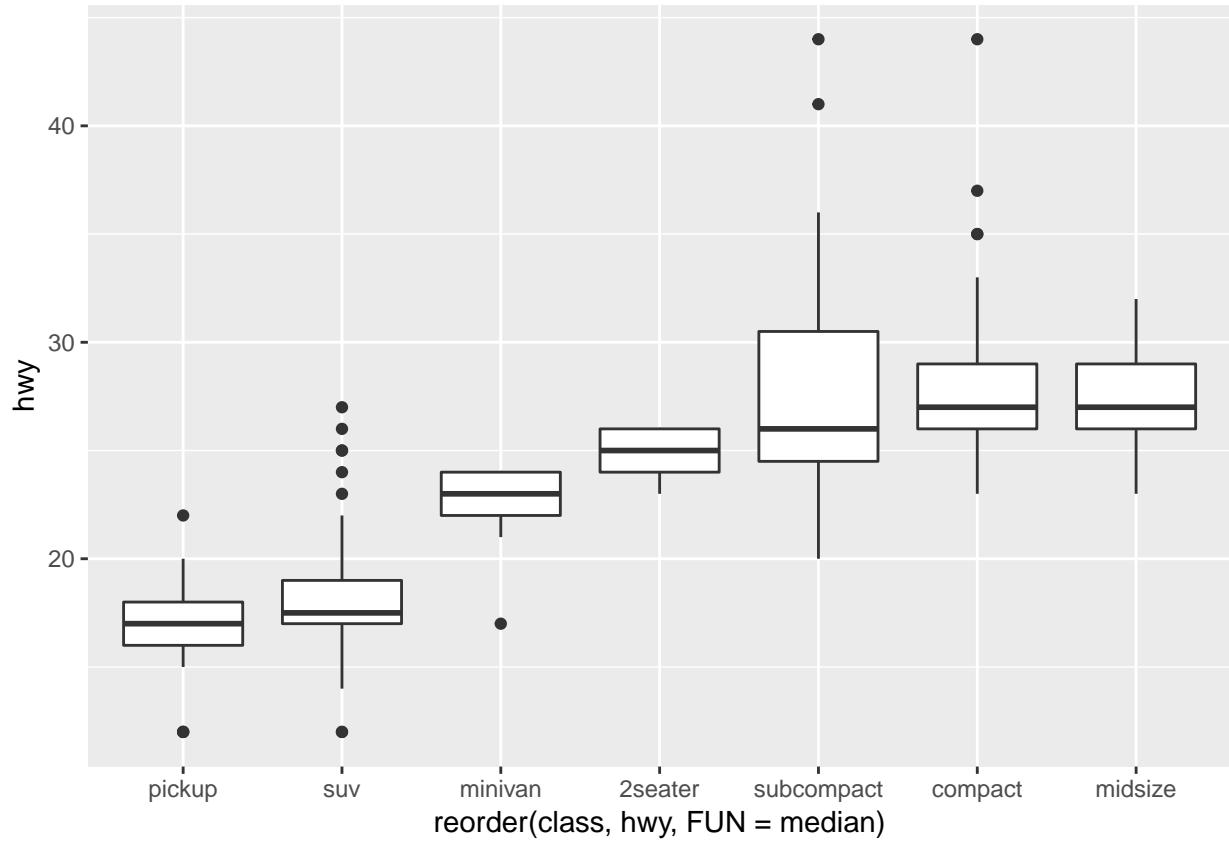
Para uma melhor visualização, é mais apropriado ordenar as categorias em ordem descendente por meio do `reorder`:

```
ggplot(data = mpg) +
  geom_boxplot(mapping = aes(x = reorder(class, hwy, FUN = median), y = hwy))
```



Também é possível inverter a localização dos eixos do gráfico caso os nomes de X sejam grandes:

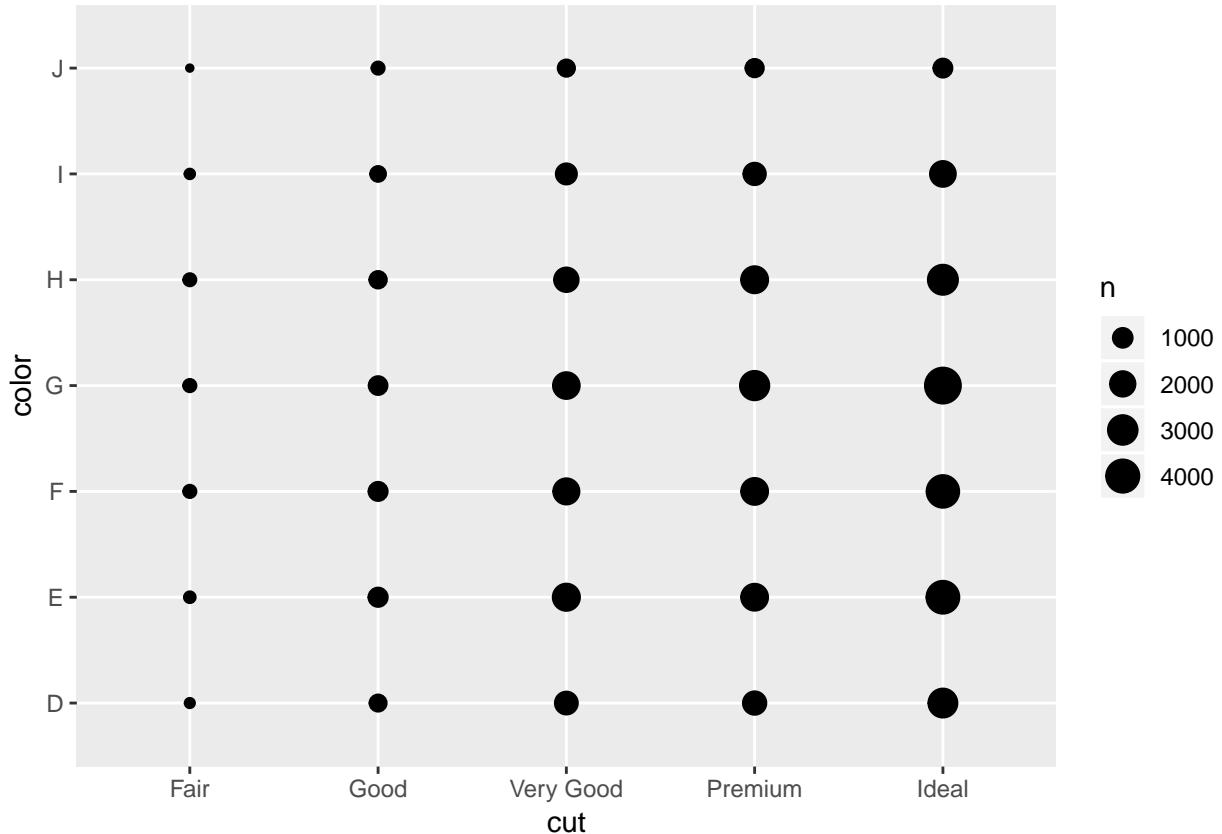
```
ggplot(data = mpg) +
  geom_boxplot(mapping = aes(x = reorder(class, hwy, FUN = median), y = hwy))
```



### Variação entre duas variáveis categóricas

Para visualizar a variação entre duas variáveis categóricas, o mais apropriado de acordo com o autor é apresentar um gráfico com a contagem de cada grupo em uma categoria:

```
ggplot(data = diamonds) +
  geom_count(mapping = aes(x = cut, y = color))
```



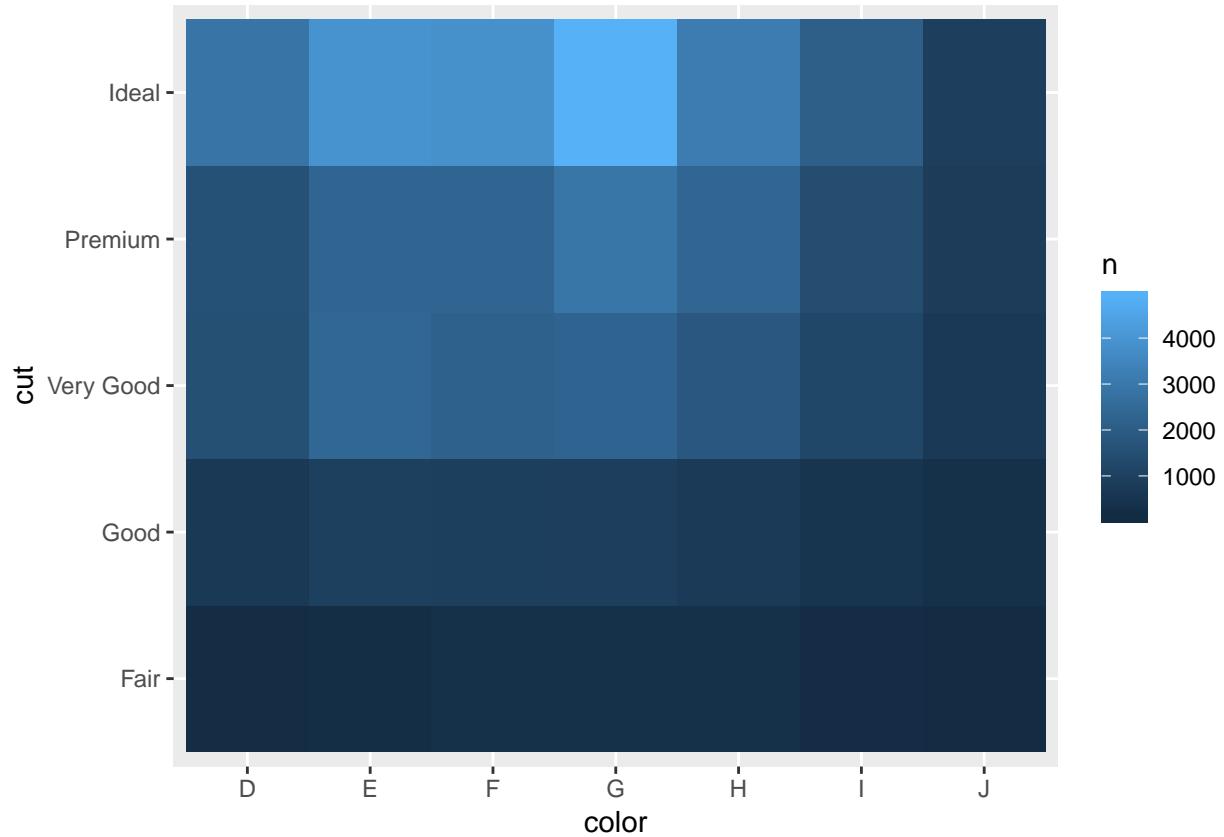
Assim como com os dados iniciais, é possível calcular essa contagem e apresentar em forma de tabela:

```
diamonds %>%
  count(color, cut)
```

```
## # A tibble: 35 x 3
##   color cut     n
##   <ord> <ord> <int>
## 1 D     Fair    163
## 2 D     Good    662
## 3 D     Very Good 1513
## 4 D     Premium 1603
## 5 D     Ideal   2834
## 6 E     Fair    224
## 7 E     Good    933
## 8 E     Very Good 2400
## 9 E     Premium 2337
## 10 E    Ideal   3903
## # ... with 25 more rows
```

Ou visualizar a relação entre as variáveis por meio de um gráfico de calor:

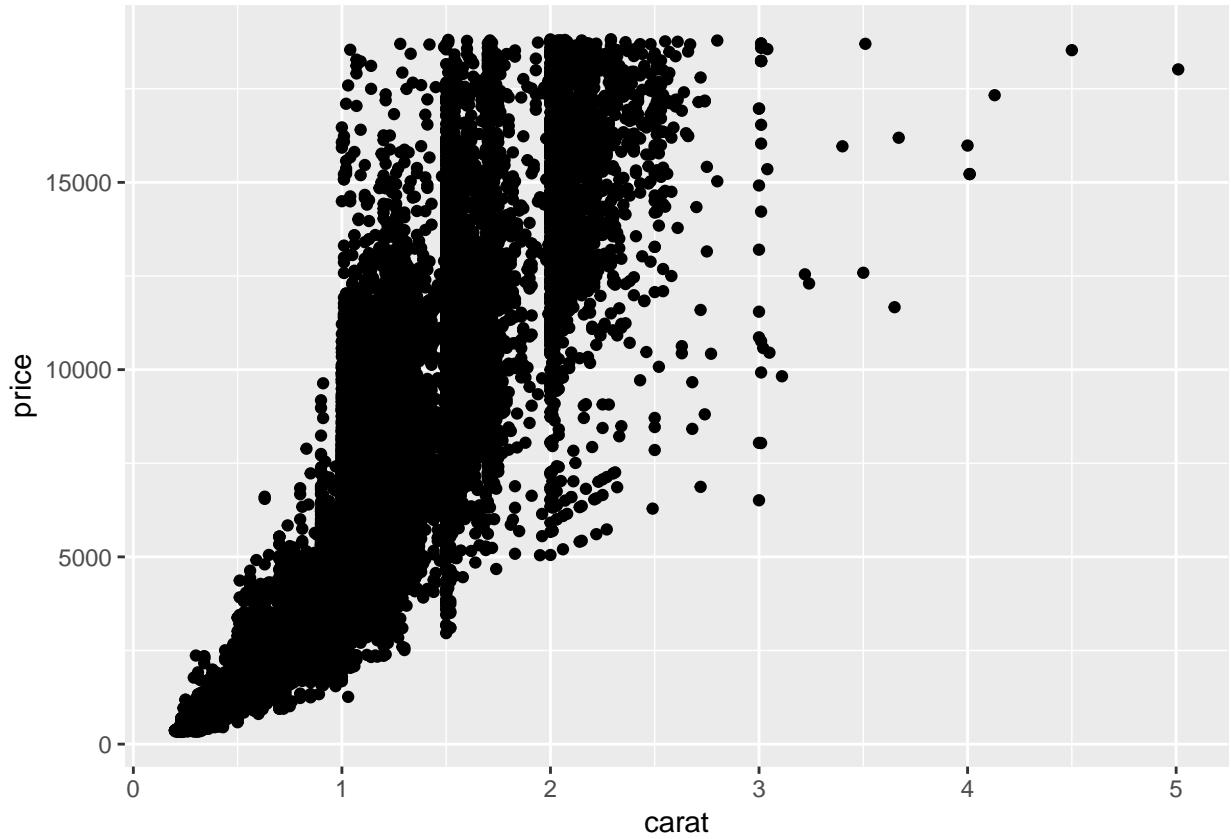
```
diamonds %>%
  count(color, cut) %>%
  ggplot(mapping = aes(x = color, y = cut)) +
  geom_tile(mapping = aes(fill = n))
```



### Variação entre duas variáveis contínuas

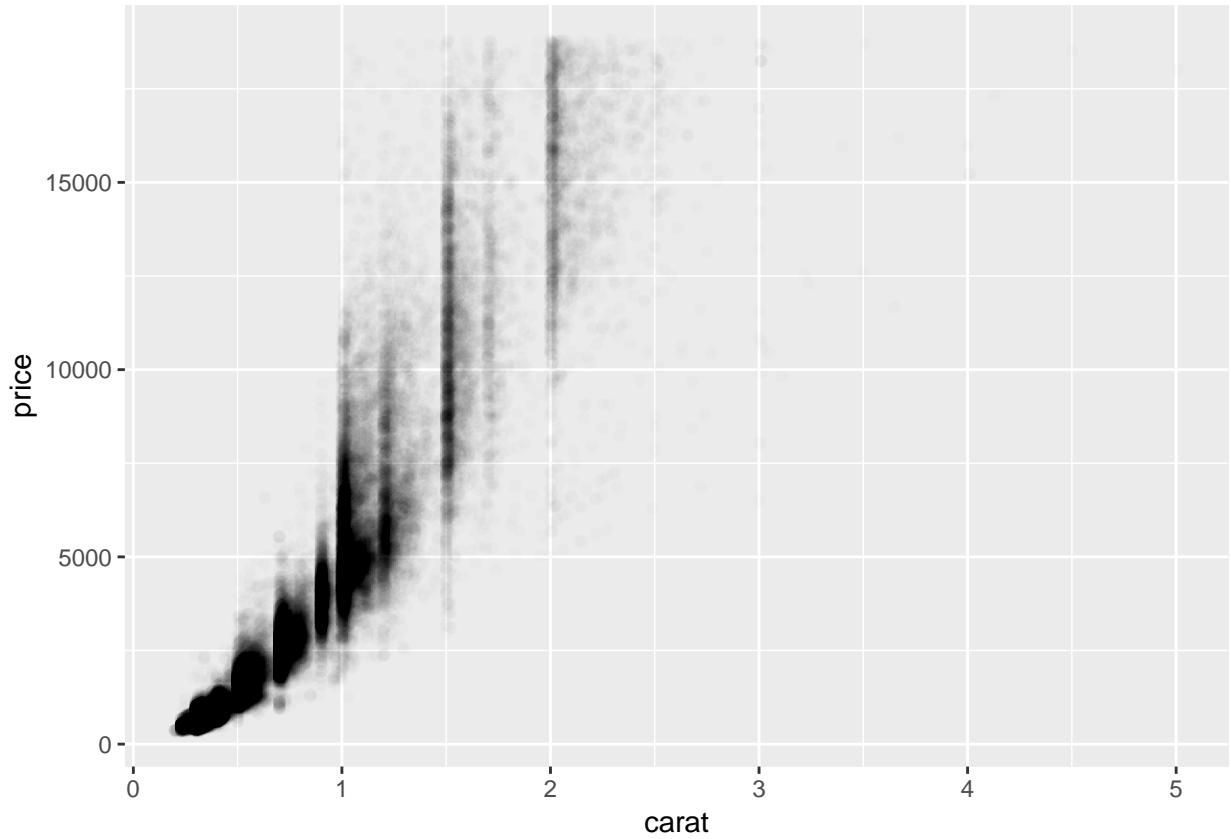
Quando envolve duas variáveis contínuas, o mais apropriado é um gráfico de dispersão:

```
ggplot(data = diamonds) +
  geom_point(mapping = aes(x = carat, y = price))
```



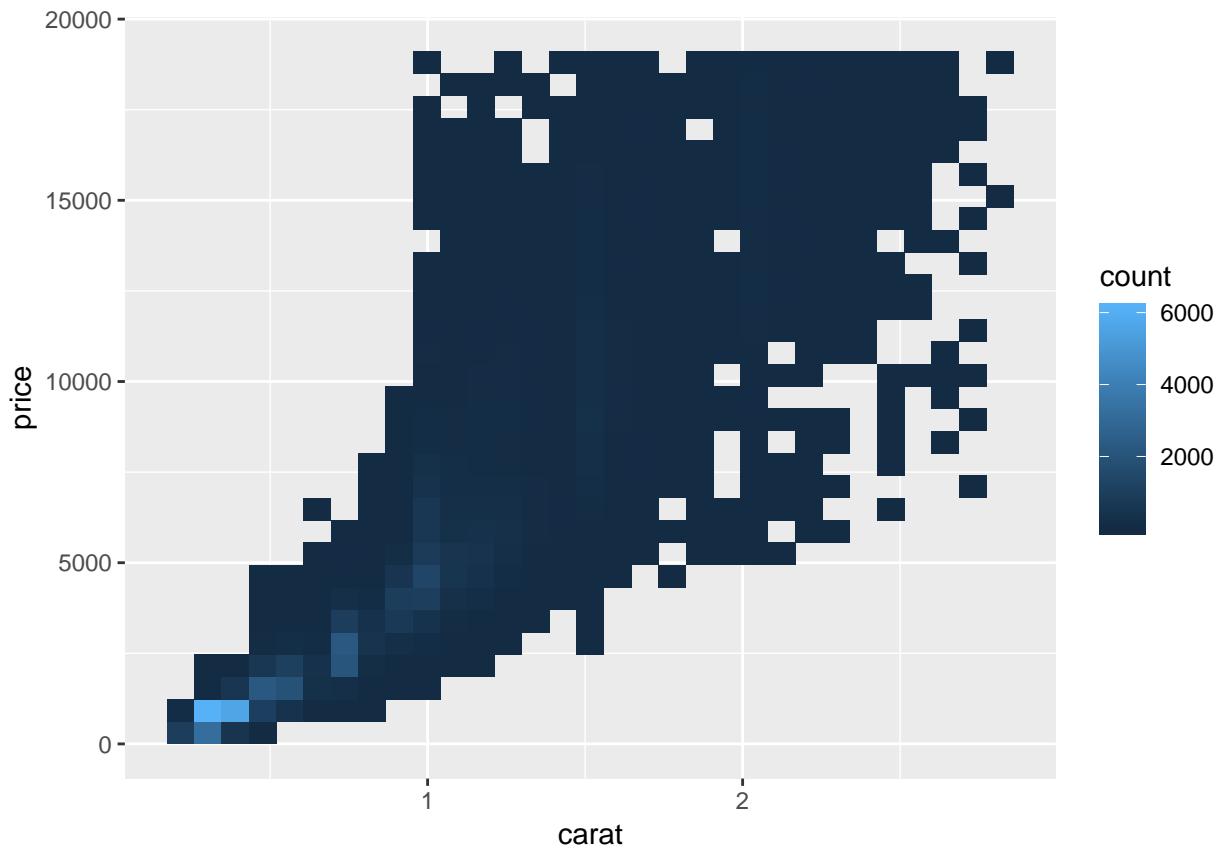
Adiconando transparência ao gráfico facilita a visualização:

```
ggplot(data = diamonds) +  
  geom_point(mapping = aes(x = carat, y = price), alpha = 1 / 100)
```



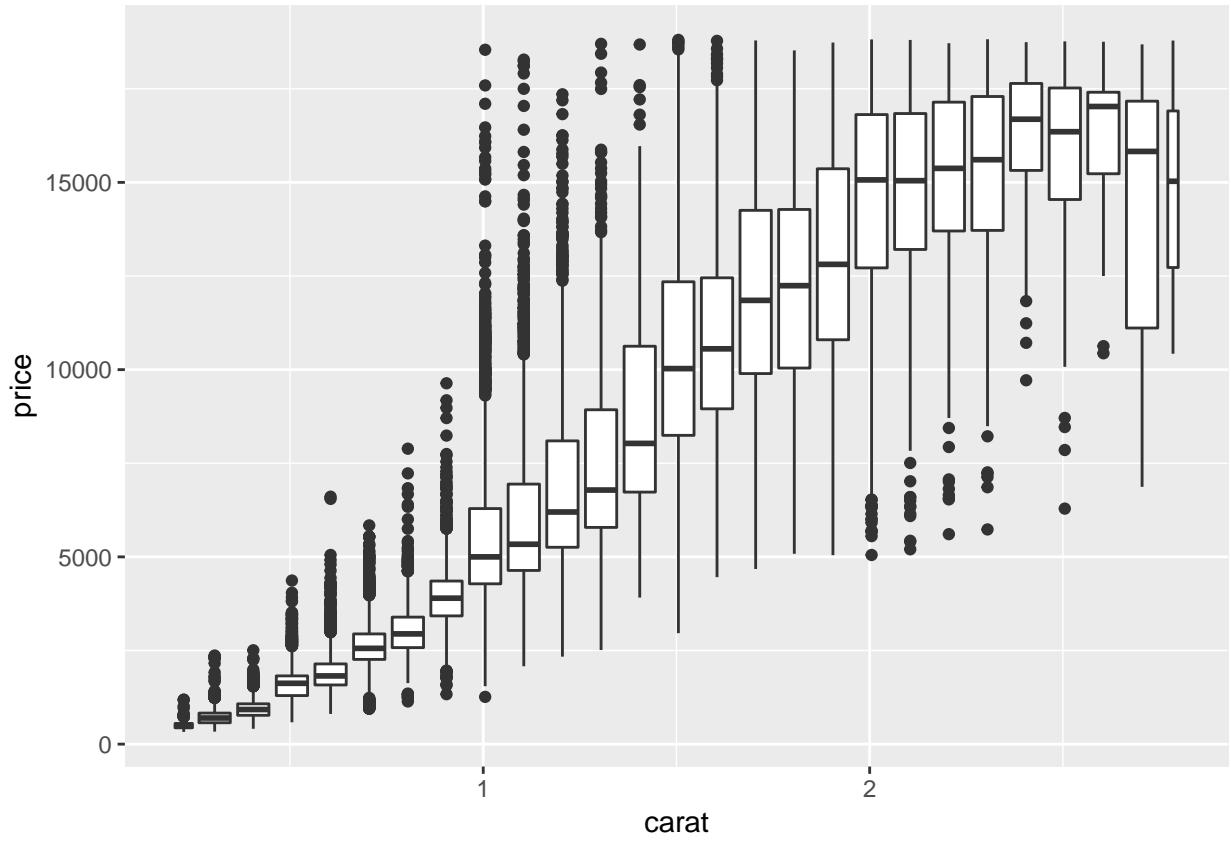
Outro ponto possível é criar gráficos com ‘caixinhas’ que agrupam os valores. Esse tipo de gráfico é uma alternativa para visualizar grandes bancos:

```
ggplot(data = smaller) +  
  geom_bin2d(mapping = aes(x = carat, y = price))
```



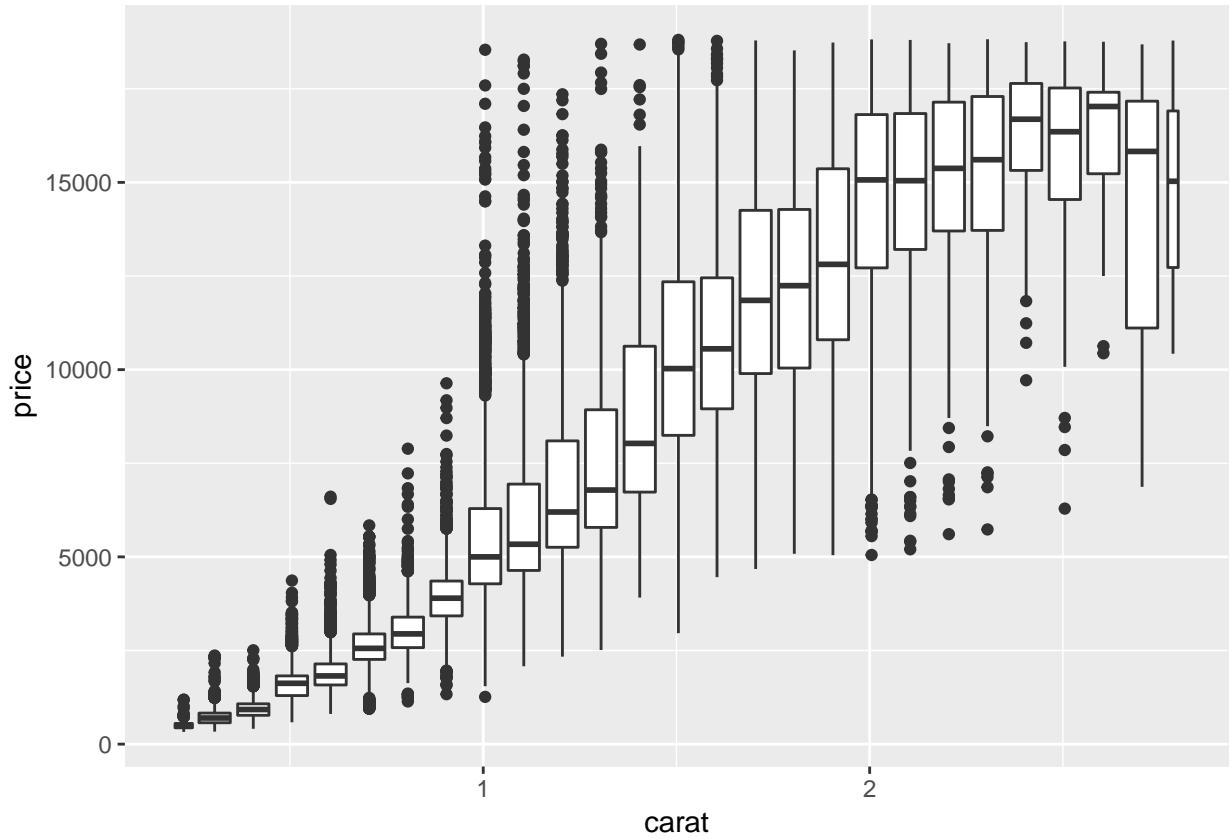
Uma outra maneira de visualizar a relação entre uma variável continua e categórica é fazer com que a variável continua ‘pareça’ uma variável categórica. Com isso é possível utilizar um boxplot, por exemplo:

```
ggplot(data = smaller, mapping = aes(x = carat, y = price)) +  
  geom_boxplot(mapping = aes(group = cut_width(carat, 0.1)))
```



Também é possível selecionar a quantidade de observações em cada caixa:

```
ggplot(data = smaller, mapping = aes(x = carat, y = price)) +  
  geom_boxplot(mapping = aes(group = cut_width(carat, 0.1)))
```

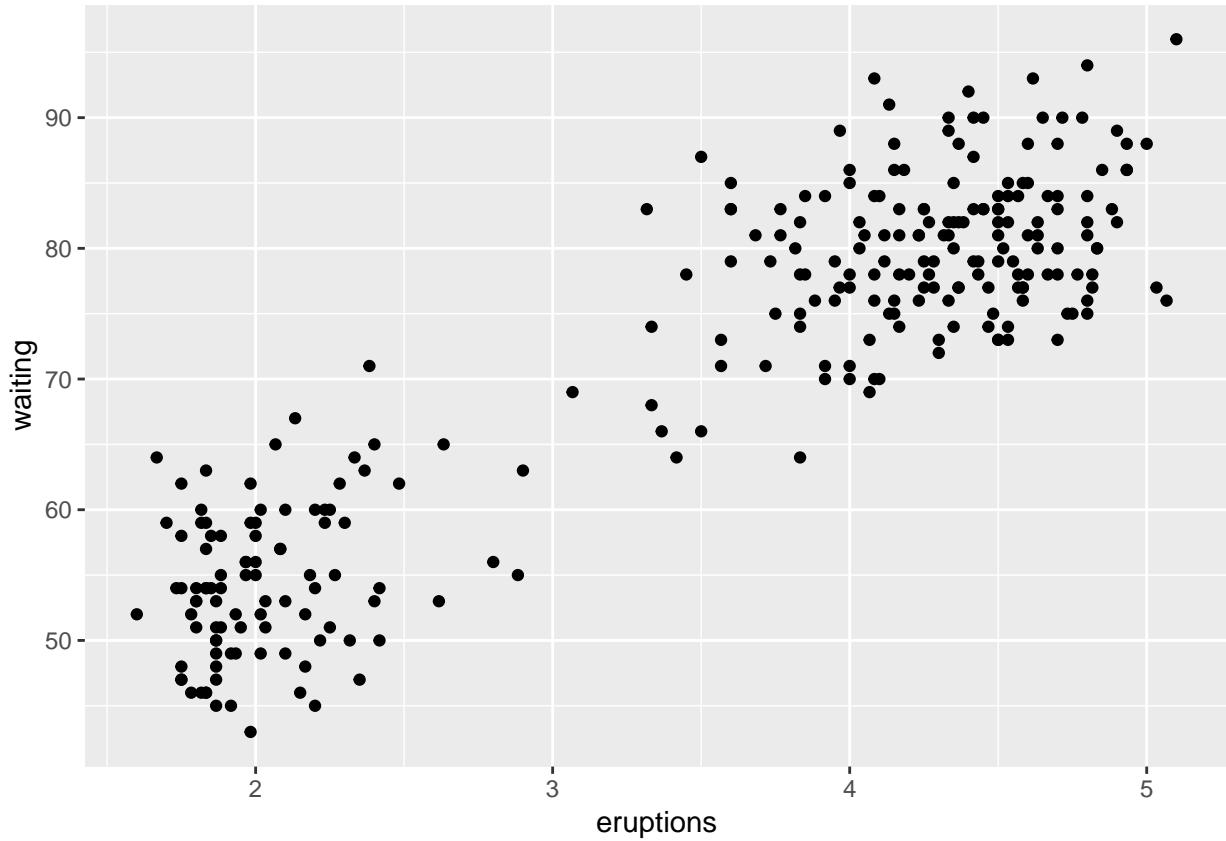


## Identificando padrões e modelos

Ainda no capítulo, é apresentado como identificar padrões envolvendo as variáveis, envolvendo a relação entre duas variáveis, a força dessa relação, etc.

O exemplo apresentado envolve a relação entre erupções e o tempo de espera entre erupções. Por meio de um gráfico de dispersão é possível visualizar o padrão entre estas duas variáveis:

```
ggplot(data = faithful) +
  geom_point(mapping = aes(x = eruptions, y = waiting))
```



Por meio da covariância entre duas variáveis, é possível reduzir o grau de incerteza envolvendo a predição dos valores de uma das variáveis por meio do controle exercido pela primeira. Envolve a elaboração de modelos, é possível extrair padrões presentes nos dados.

Dito isso, o autor apresenta um modelo que prediz o valor do preço por quilate do banco contendo informações sobre diamantes e computa os resíduos (o que permite verificar o preço do diamante dado que o efeito do quilate foi removido)

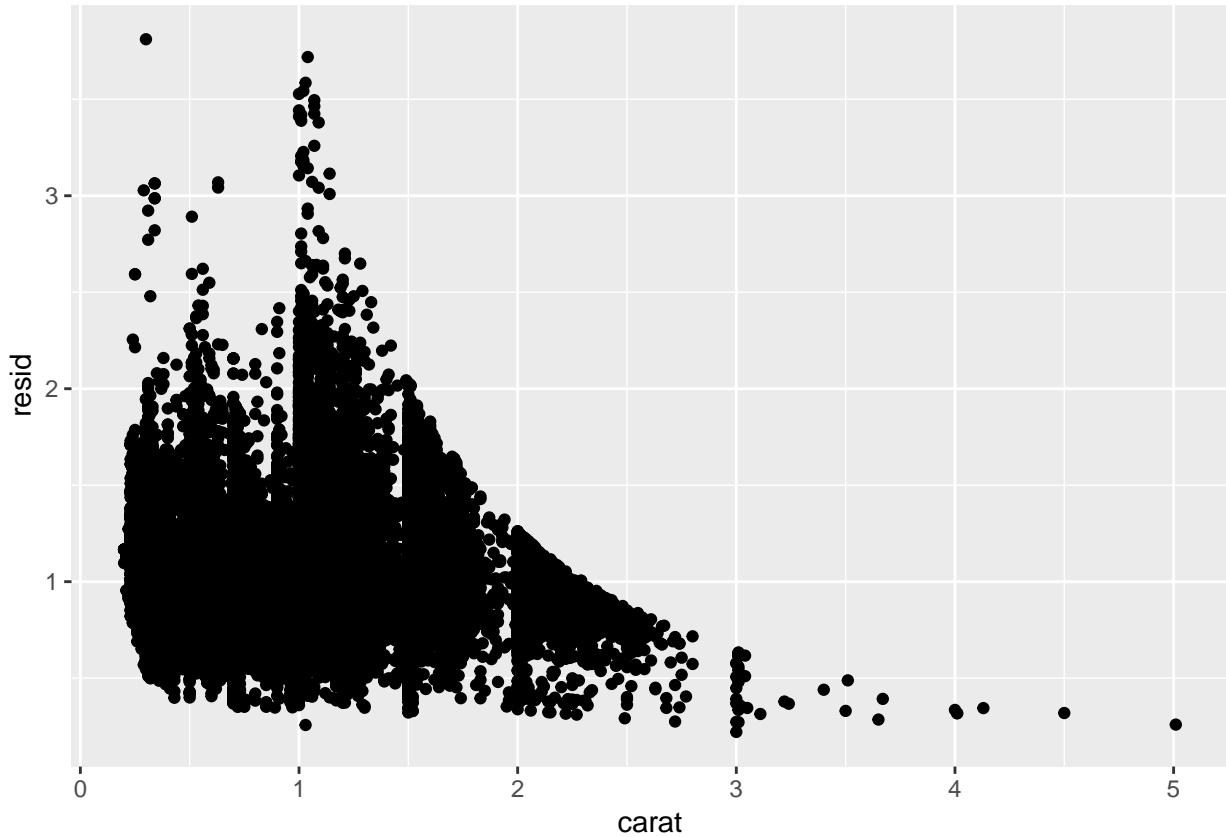
```
library(modelr)

## Warning: package 'modelr' was built under R version 3.5.3

mod <- lm(log(price) ~ log(carat), data = diamonds)

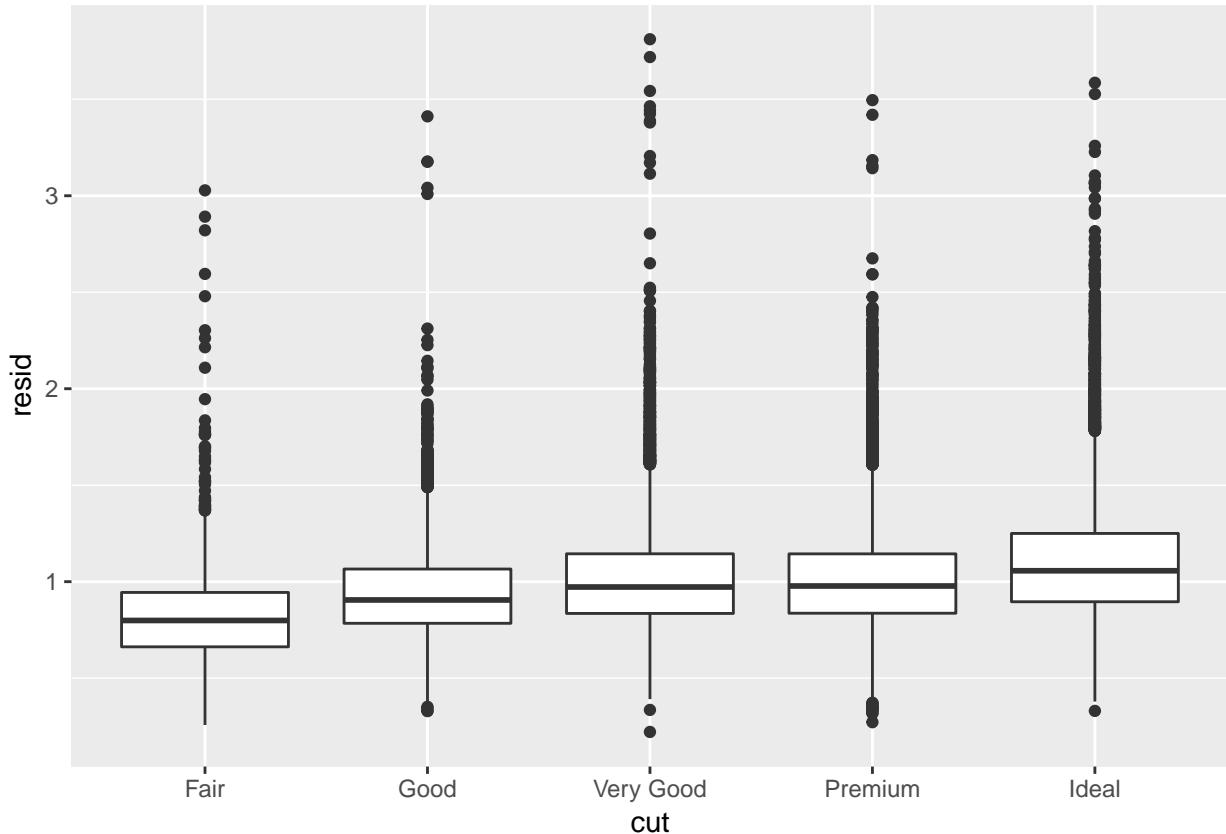
diamonds2 <- diamonds %>%
  add_residuals(mod) %>%
  mutate(resid = exp(resid))

ggplot(data = diamonds2) +
  geom_point(mapping = aes(x = carat, y = resid))
```



Nesse gráfico, podemos verificar a relação envolvendo o quilate e o resíduo do modelo da regressão linear envolvendo o efeito do quilate sobre o preço. A relação entre essas duas variáveis é forte e removendo o efeito de uma variável sobre a outra é possível verificar a relação entre o preço do diamante e o seu corte, por exemplo:

```
ggplot(data = diamonds2) +  
  geom_boxplot(mapping = aes(x = cut, y = resid))
```



Nesse gráfico temos a relação entre o resíduo (a parte não explicada do modelo, ou seja, o que não é explicado da variação do preço em relação ao quilate do diamante) e o tipo de corte.

Esses são os exemplos apresentados pelo autor envolvendo a análise exploratória de dados.

## Questão X

Na questão 10, é solicitado para executar um modelo de regressão linear envolvendo a variável voto no partido do incumbente e crescimento econômico. Ou seja: Qual o impacto da variação no crescimento econômico na quantidade de votos recebida pelo partido do incumbente?

```
load("C:/Users/Antonio/Documents/Dados>Listas/AD_5/Lista 6/vote_growth_usa.RData")
```

Após ler o banco, é executado o modelo de regressão linear. Os resultados estão abaixo:

```
reg <- lm(Vote ~ Growth, data = bd)

#obtendo resultado do modelo

summary(reg)

## 
## Call:
## lm(formula = Vote ~ Growth, data = bd)
##
```

```

## Residuals:
##      Min     1Q Median     3Q    Max
## -8.1968 -3.7667 -0.7972 3.1294 10.0107
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 51.5082    0.8569  60.110 < 2e-16 ***
## Growth      0.6249    0.1577   3.962  0.00039 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.955 on 32 degrees of freedom
## Multiple R-squared:  0.3291, Adjusted R-squared:  0.3081
## F-statistic:  15.7 on 1 and 32 DF,  p-value: 0.0003898

```

O *output* do modelo apresenta diversos resultados que serão analisados aqui:

- *Residuals*: Apresenta os valores do resíduo do modelo e a sua distribuição por meio dos valores mínimos, máximos, mediana, 1º e 3º quartis.
- *Coefficients*: Contém os valores dos coeficientes do modelo (a constante e a vi).

Nesse caso, a constante apresenta um efeito de 51.5 na VD. Ou seja, é o valor esperado da VD quando X é 0. O **erro padrão** apresenta o erro associado ao coeficiente, que no caso da constante é de 0.8569. O **valor T** é o teste utilizado para verificar se o coeficiente é significantemente diferente de 0. A última coluna é o **resultado do teste de hipótese** (de que o coeficiente é diferente de 0). Nesse caso, o valor apresentado é de que o valor do coeficiente é estatisticamente diferente de 0.

Em relação a VI, o resultado do coeficiente é de 0.6249. Ou seja, mantendo todo o resto constante, a variação de **1% na porcentagem do pib per capita** leva a uma **variação positiva de 0.6249** na porcentagem de votos do partido incumbente **mantendo todo o resto constante**. Em relação ao erro padrão do coeficiente, o valor obtido foi de 0.1577. Já no tocante ao valor T, o resultado foi de 3.962. Com esse valor obtido, o teste de hipóteses apresenta um p-valor < 0.005. Ou seja, de que o valor do coeficiente é estatisticamente diferente de 0.

- *Residual Standard Error*

Já o erro padrão do resíduo representa a distância média entre os pontos e a linha do modelo expressa na unidade de medida da VD. Nesse caso, o valor obtido é de 4,955%. Isso significa um valor alto dado a VD envolve percentual de votos recebidos.

- *Multiple R-Squared e Adjusted R-Squared*

O R<sup>2</sup> representa a proporção de variação da VD que é explicada pelas VIs presentes no modelo. Nesse caso, o modelo consegue explicar 0.3291 (quase 33%) da variação da VD por meio da variação no PIB per capita. Já o R<sup>2</sup> ajustado é uma medida que penaliza a inclusão variáveis preditoras no modelo.

Para identificar o intervalo de confiança do modelo (95%), executamos o seguinte código:

```
confint(reg)
```

```

##              2.5 %    97.5 %
## (Intercept) 49.7627069 53.2536139
## Growth      0.3036193  0.9461963

```

Ou seja, o intervalo de confiança do coeficiente da VI é localizado entre 0.30 (min) e 0.94 (max)

## Questão XI

Na questão 11, é executado o mesmo modelo com um intervalo de anos menor (1876 a 1932).

Primeiramente, convertemos os valores da coluna ano que estão em `factor` para `numeric` e depois selecionamos os anos específicos e executar o modelo:

```
bd$Year <- as.character(bd$Year) # Convertendo para caracter
bd$Year <- as.numeric(bd$Year) # Convertendo para numeric

vote <- bd[ which(bd$Year < 1933), ]

reg.1 <- lm(Vote ~ Growth, data = vote) #modelo

summary(reg.1)

##
## Call:
## lm(formula = Vote ~ Growth, data = vote)
##
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -9.7209 -3.5931  0.5013  3.1253  9.3127 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 51.9850    1.5371  33.821 4.66e-14 ***
## Growth      0.5336    0.2366   2.255   0.042 *   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.638 on 13 degrees of freedom
## Multiple R-squared:  0.2811, Adjusted R-squared:  0.2258 
## F-statistic: 5.083 on 1 and 13 DF,  p-value: 0.04205
```

Em comparação com o modelo executado na questão 10, percebemos que o valor do coeficiente da VI é menor ( $0.5336 < 0.6249$ ). Entretanto, ambos os valores são significativos. Apesar disso, a significância (p-valor) do modelo da questão 11 é menor que o valor do modelo da questão 10 ( $0.042 > 0.00$ ). Em relação ao valor do coeficiente da VI na questão 11, significa que o aumento de 1 unidade na VI (% variação PIB per capita) leva a um aumento de 0.53% de votos no partido do incumbente.

No tocante ao erro padrão do resíduo, o apresentado no modelo da questão 11 é maior do que o modelo executado na questão 10 ( $5.638 > 4.955$ ). Ou seja a adequação do segundo modelo acaba sendo pior.

O  $R^2$  do segundo modelo também é menor. Ou seja, o segundo modelo explica uma menor variação da VD (0.28) do que o primeiro modelo (0.33).

Por fim, em relação ao intervalo de confiança, executamos o seguinte comando:

```
confint(reg.1)
```

```
##           2.5 %    97.5 %
## (Intercept) 48.66440604 55.305605
## Growth      0.02230689 1.044796
```

E o resultado obtido mostra que o intervalo de confiança (95%) do  $\hat{\beta}$  vai de 0.022 (min) a 1.04 (max)