

Sexta Lista de Exercício

Caio Rios

19 de maio de 2019

1. Descreva os conceitos abaixo:

a) Variável dependente:

A variável dependente é o fenômeno que o pesquisador almeja explicar. Ela recebe este nome pois ela depende de outros fatores para existir ou para variar. Na literatura americana também é conhecida como “outcome”.

b) Variável independente:

A variável independente consiste no fenômeno que explica a variação da variável dependente. Ela recebe este nome pois ela independente de outras variáveis no modelos para acontecer.

c) Apresente qual a relação existente entre variáveis independentes e dependente:

Se alguém, por exemplo, deseja explicar o porquê em alguns lugares a taxa de comparecimento é maior (ou menor) o pesquisador precisa pensar no que estaria causando essa variação de comparecimento. Uma hipótese seria que a competição política influencia na decisão do eleitor de comparecer às urnas. Neste exemplo, competição política explica comparecimento, logo competição política é minha variável independente e, por sua vez, o comparecimento eleitoral é sua variável dependente. A relação entre elas é de causação. VI causa VD.

2. Em análise de dados, qual o nome dado à equação abaixo?

$$Y = \alpha + \beta X + \mu$$

Esta equação representa uma regressão bivariada.

3. Com suas palavras, apresente uma definição para cada um dos componentes da equação apresentada no exercício 2.

Y -> Valor observado da variável dependente.

α -> Intercepto. Isto é, o valor estimado de Y , quando os outros componentes se igualam a 0. Em um gráfico, seria o valor de Y quando X é 0.

β -> Coeficiente de variação. O efeito de X em Y . Mais especificamente, seria o quanto Y varia ao acrescentar uma unidade da variável X . Calculado pela covariância de X e Y dividido pela variação de X ao quadrado.

X -> Valor observado da variável independente.

μ -> Erro amostral ou componente estocástico. Seria o erro do modelo. A distância entre os valores observados e o modelo.

4. Apresente o componente sistemático da equação apresentada no exercício 2. Descreva por quê é sistemático.

O componente sistemática da equação é $\alpha + \beta X$. Este é o componente que vai predizer o valor estimado de Y .

5. Apresente o componente estocástico da equação apresentada no exercício 2. Descreva por quê é estocástico.

O componente estocástico da equação é μ . Este componente é aleatório e responde pela adequação do modelo aos dados reais. Ou seja o seu valor é indeterminado. A soma das distâncias dos pontos reais e o modelo (reta de regressão) precisa se igualar a 0.

6. Descreva a diferença entre Y_i e Y (chapéu). Qual a relação desses dois componentes com μ ?

Y_i é o valor observado da variável dependente. E Y (chapéu) é o valor estimado da variável dependente. Em um gráfico de linha com pontos, os pontos seriam Y_i e os valores de Y que correspondem a reta seria o Y (chapéu). O Y (chapéu) não leva em conta o erro (μ) para sua estimativa, mas sim, apenas os componentes sistemáticos. Já Y_i seria a soma de Y (chapéu) e μ .

7. Com suas palavras, apresente o que é o modelo OLS e seu principal uso na análise de dados.

OLS significa Ordinary Least Square ou Mínimos Quadrados Ordinários. Ela consiste na escolha da reta de regressão para representar o modelo. A escolha é dada pela soma dos quadrados da distância entre a reta (o modelo) e os valores observados de Y . A reta será aquela que minimiza esse valor.

8. Com base no Google's R Style Guide (<https://google.github.io/styleguide/Rguide.xml#indentation>), apresente exemplos de boas práticas para os seguintes tópicos:

a) File name;

vote_growth.R

b) Identifiers;

Nome da Variável: avg.vote Nome da função: CalculateAvgVotes

c) Indentation;

Nunca usar tab para deixar separar o texto. Sempre dois espaços.

d) Spacing;

Sempre separa os argumentos da função por espaço

valor <- 5 + 7 - sqrt(220)

e) Assignment;

x <- 5

f) Commenting Guidelines;

```
# Separar o # do comentário com um espaço
x <- 2 + 2 * 15 # Comentário após linha de código
```

g) Function Definitions and Calls;

PredictCTR <- function(arg1, arg2, arg3, defaultArg = TRUE)

h) Function Documentation;

Logo abaixo da função de conter o comentário sobre o que está sendo calculado com a função. Depois explicar os argumentos abaixo da sessão criada #Arg:. Depois uma descrição do resultado

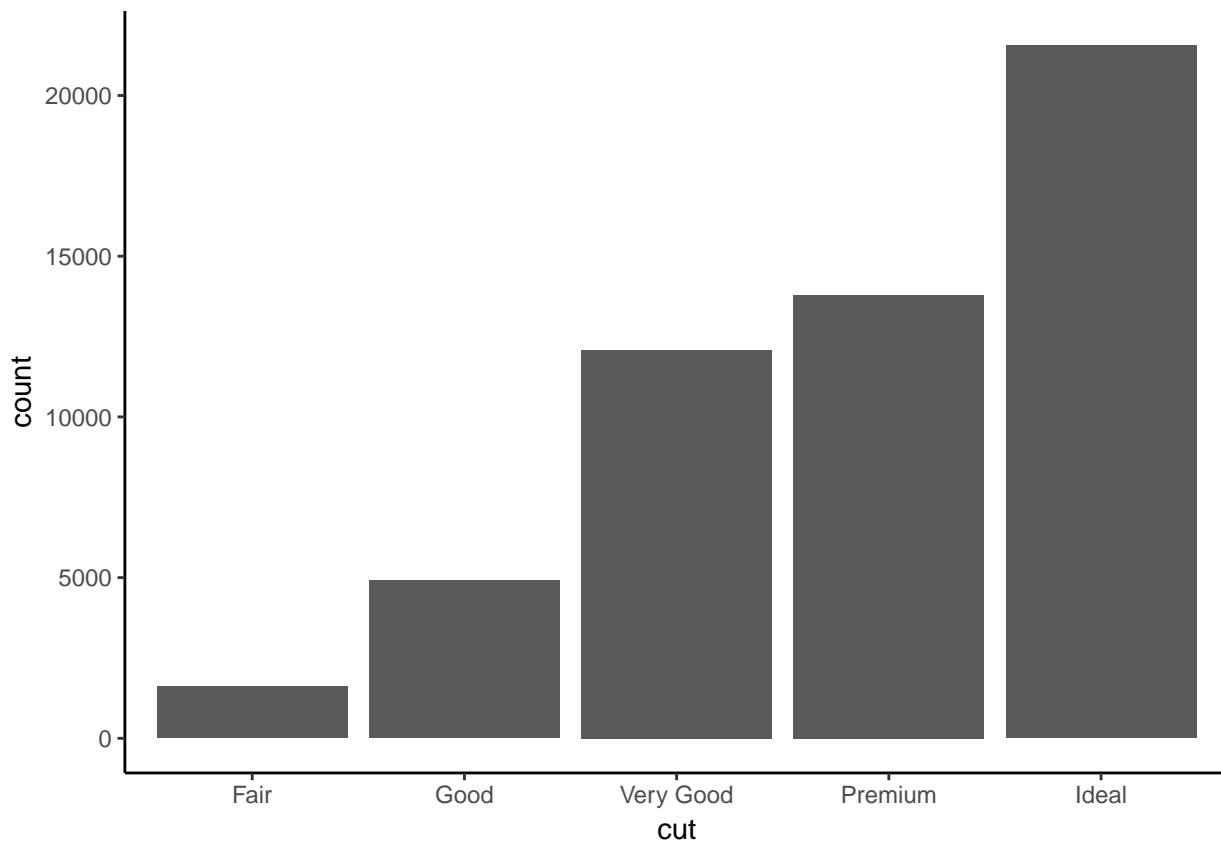
9. Leia o Capítulo 7 do livro R para Ciência de Dados (<http://r4ds.had.co.nz/exploratory-data-analysis.html>) e entregue script no R que reproduza os exemplos apresentados no capítulo. Comente seu código indicando o que está para ser realizado em cada etapa do seu script.

```
##
## 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
## -- Attaching packages ----- tidyverse 1.2.1 --
## v ggplot2 3.1.1     v readr    1.3.1
## v tibble   2.1.1     v purrr    0.3.2
## v tidyverse 0.8.3    v stringr  1.4.0
## v ggplot2 3.1.1     vforcats 0.4.0
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()
##
## Attaching package: 'magrittr'
## The following object is masked from 'package:purrr':
##   set_names
## The following object is masked from 'package:tidyverse':
##   extract
```

Variações

Visualizando distribuições a partir de gráfico de barra para variáveis categóricas.

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



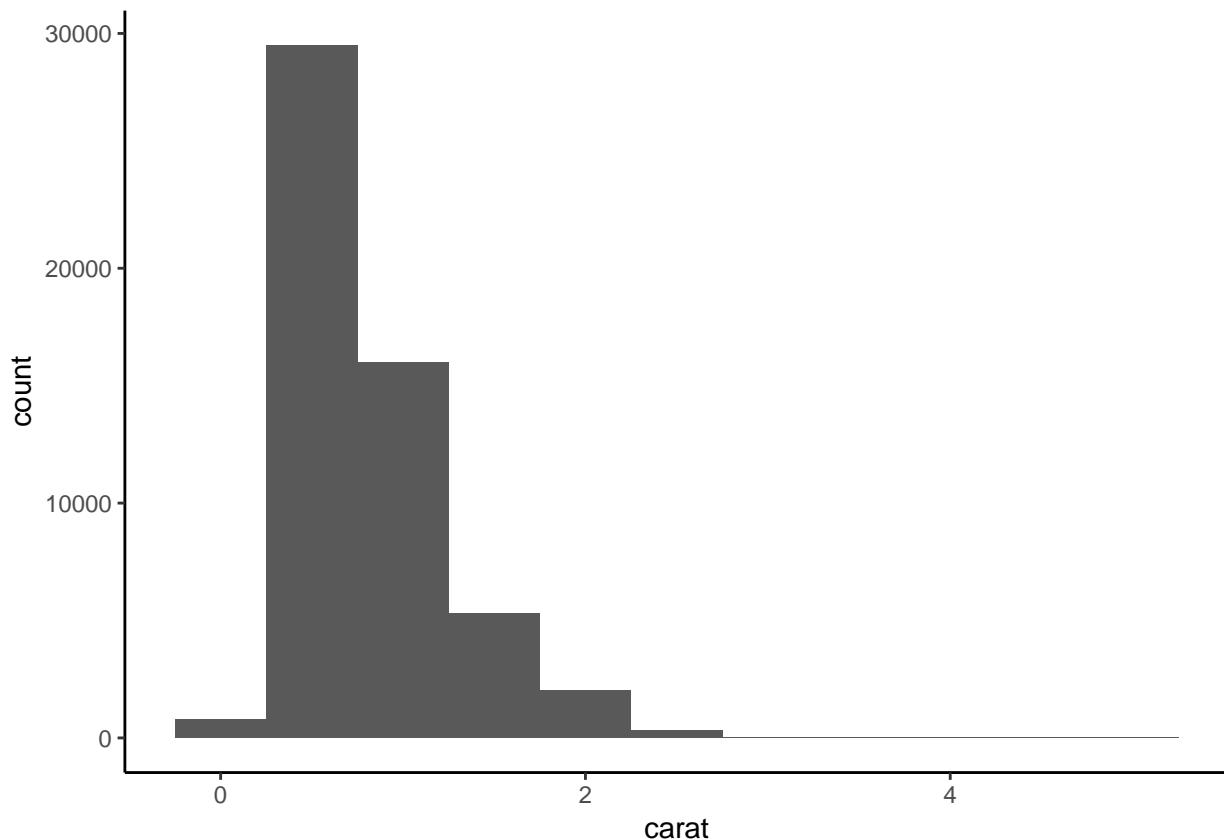
Mostrando os valores de cada barra

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

cut	n
Fair	1610
Good	4906
Very Good	12082
Premium	13791
Ideal	21551

Visualizando distribuições a partir de histogramas para variáveis contínuas

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



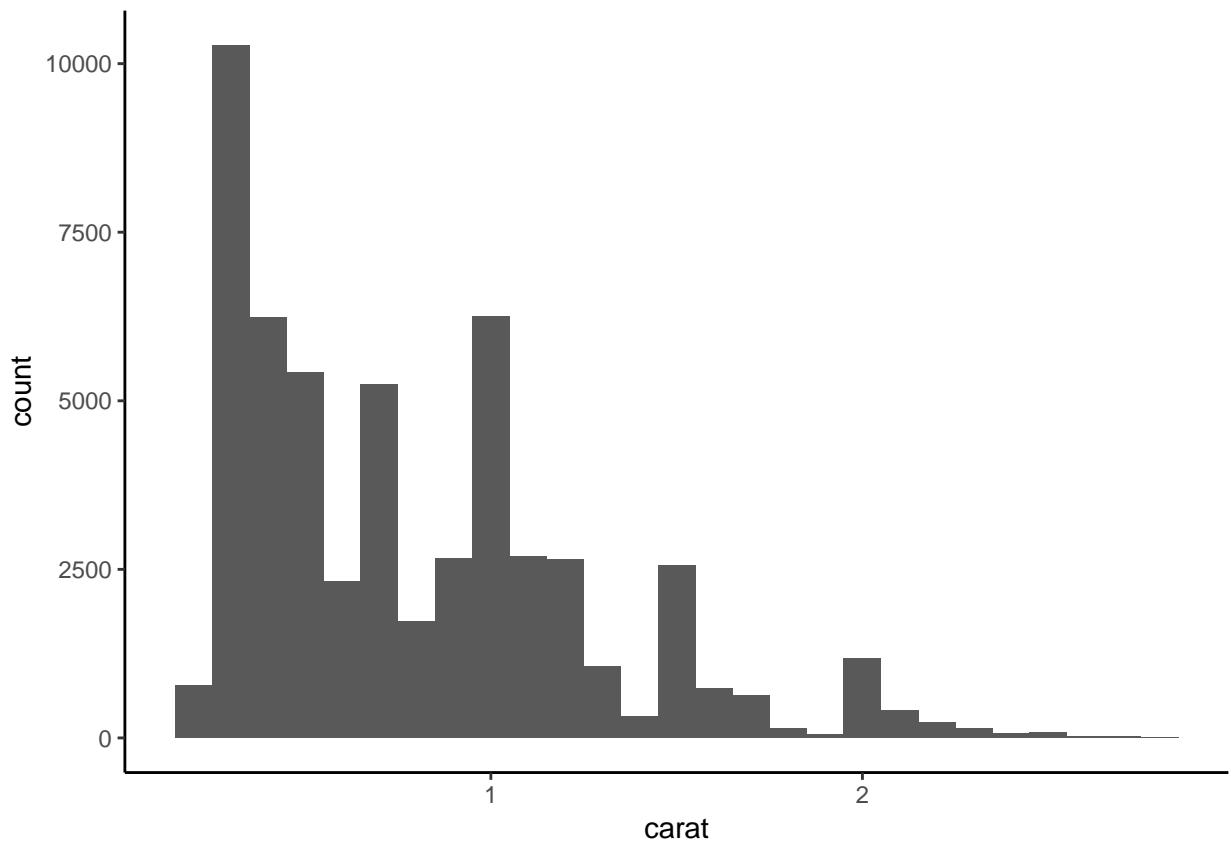
Mostrando os valores de cada “barra” em blocos de 0,5

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

cut_width(carat, 0.5)	n
[-0.25, 0.25]	785
(0.25, 0.75]	29498
(0.75, 1.25]	15977
(1.25, 1.75]	5313
(1.75, 2.25]	2002
(2.25, 2.75]	322
(2.75, 3.25]	32
(3.25, 3.75]	5
(3.75, 4.25]	4
(4.25, 4.75]	1
(4.75, 5.25]	1

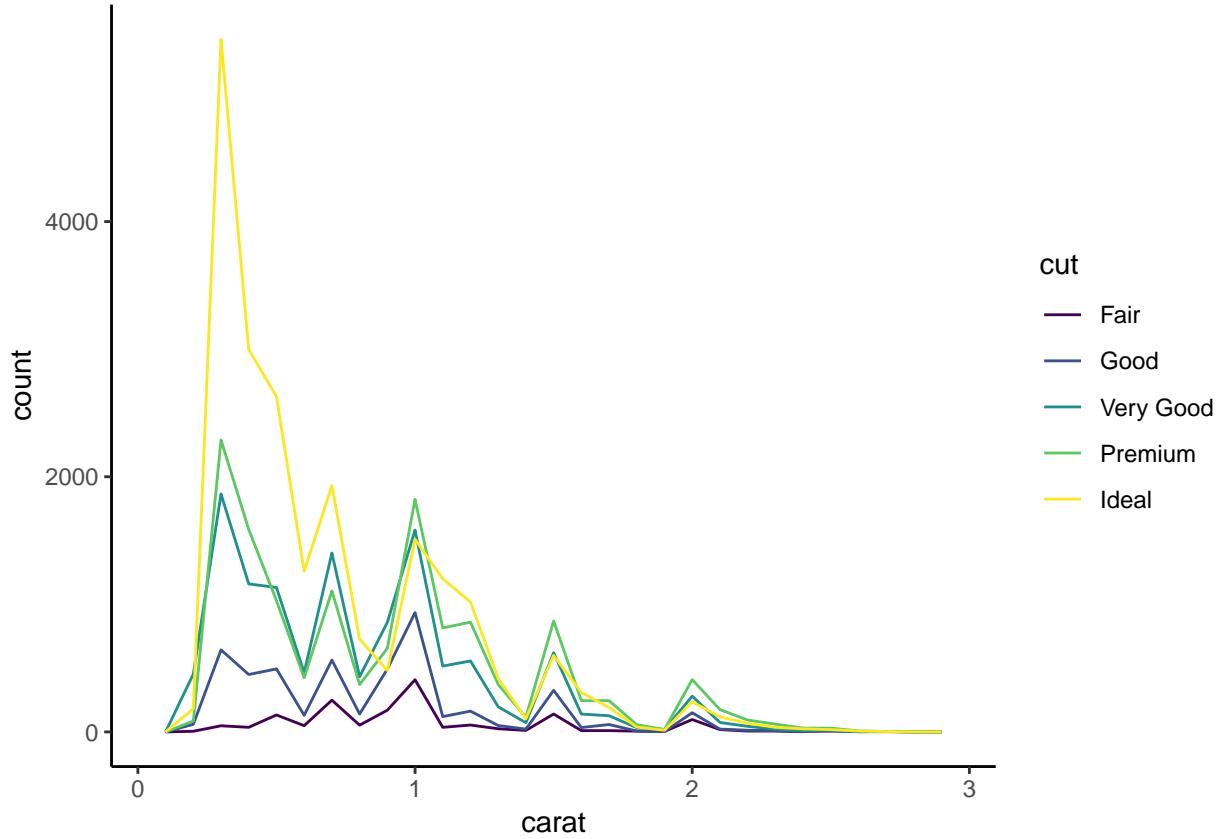
É importante olhar a distribuição em diferentes “bandwidths” variável de interesse. Vamos observar como a distribuição muda ao olharmos apenas os diamantes com carat < 3

```
smaller <- diamonds %>%
  filter(carat < 3)
ggplot(data = smaller, mapping = aes(x = carat)) +
  geom_histogram(binwidth = 0.1)
```



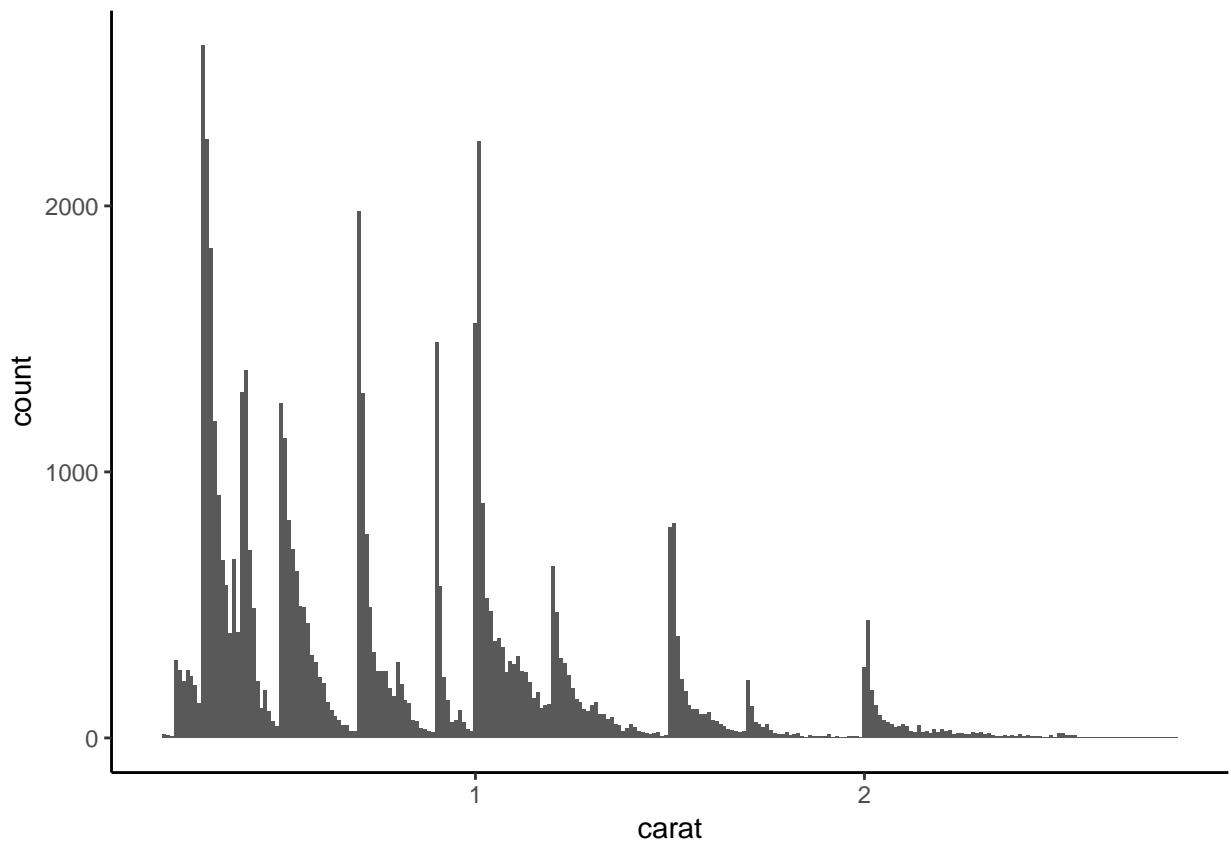
Para comparar múltiplos histogramas, utilizar geom_freqpoly.

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



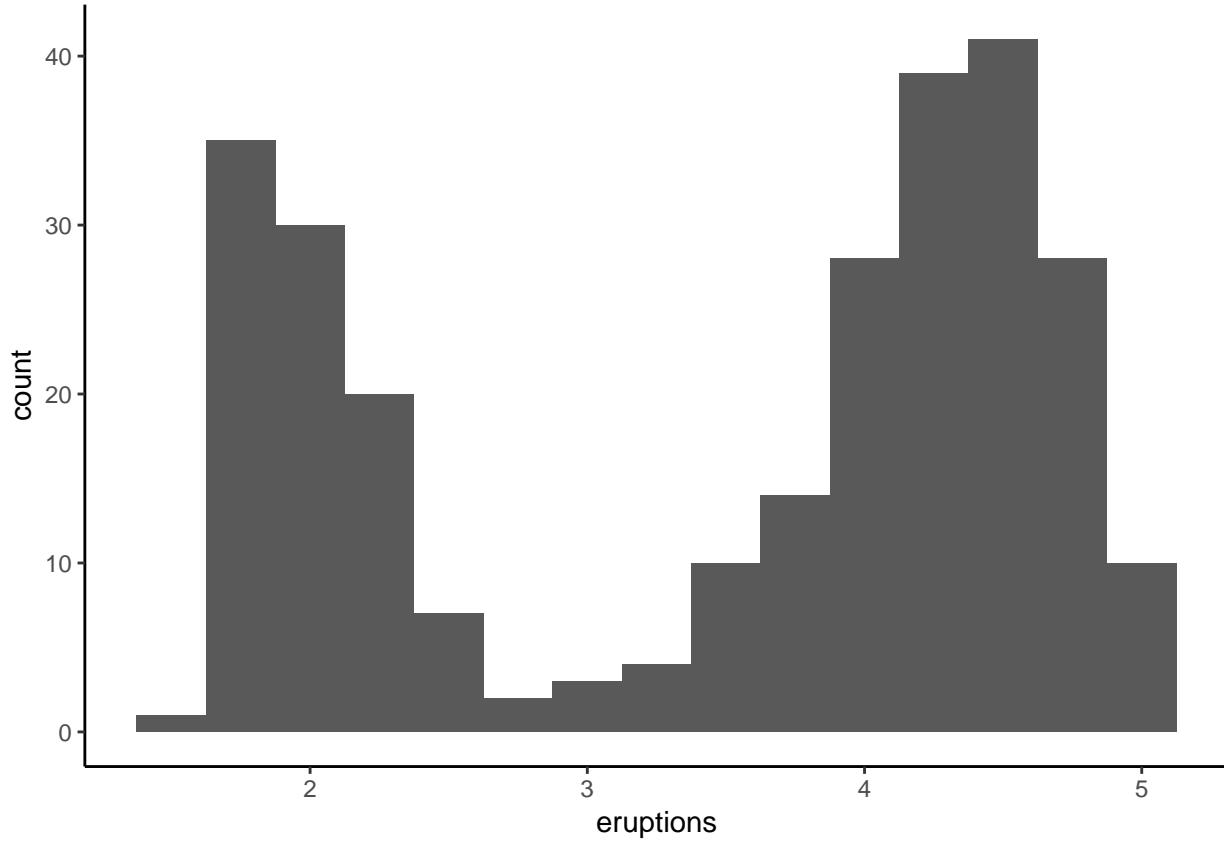
Plotando um histograma de carat para observar fatores interessantes dos dados. Como a quantidade de diamantes com carat inteiro.

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



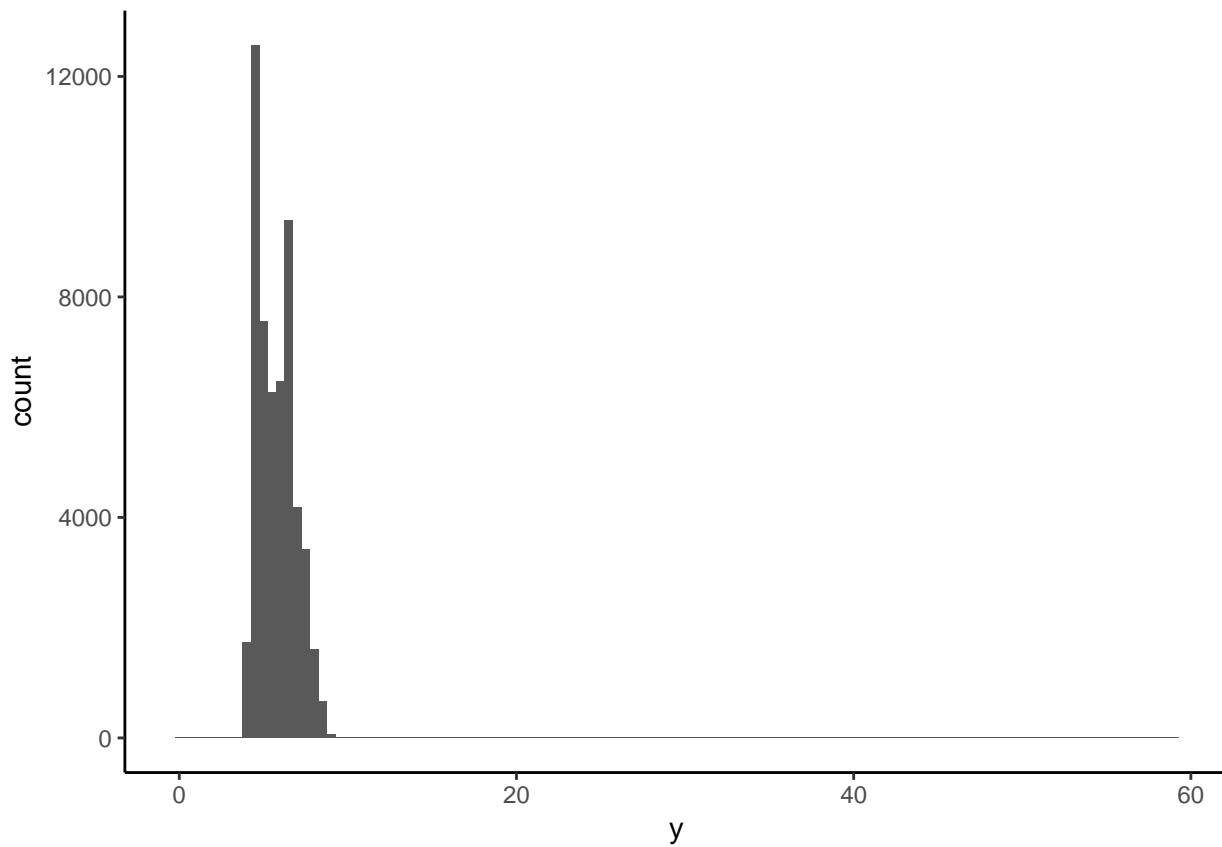
Plotando outro histograma para observar como os dados se comportam. Neste caso, observam-se dois clusters.

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



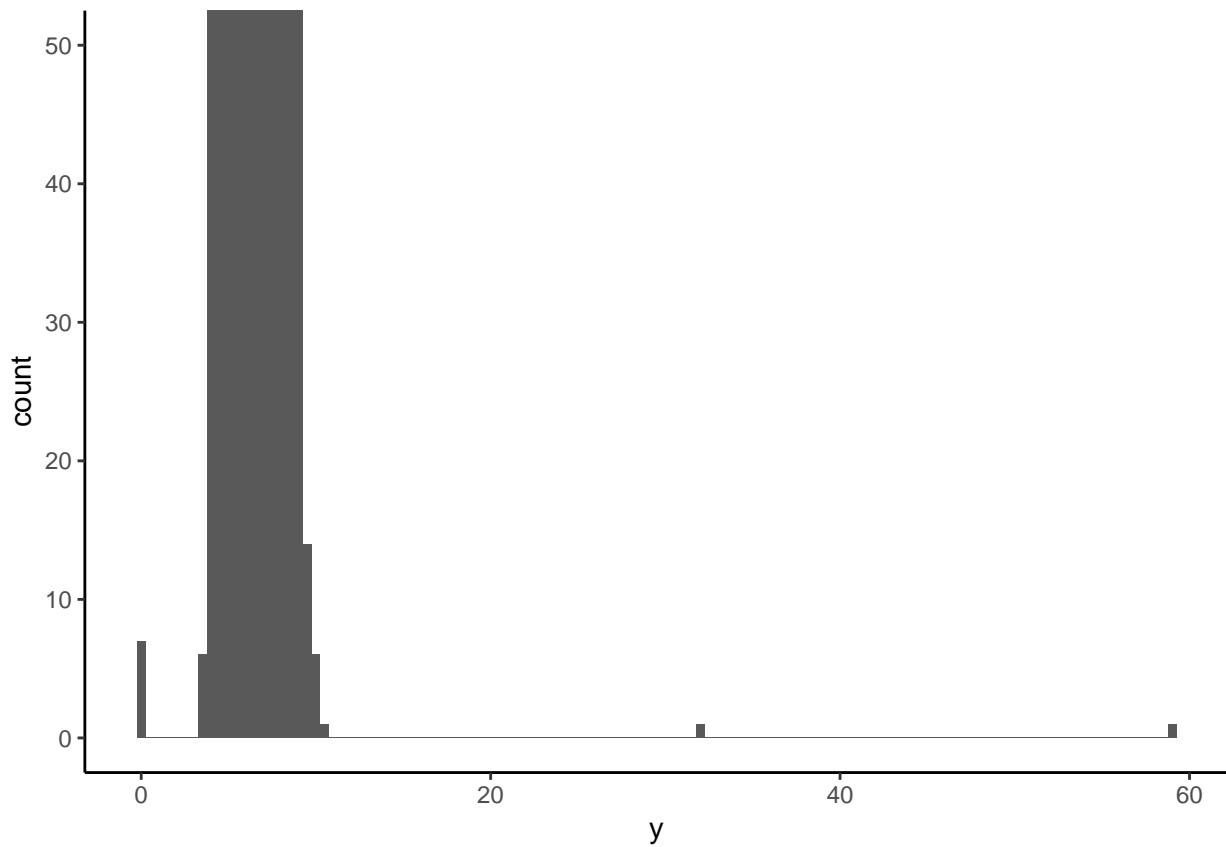
Quando há valores extremos ou outliers, o histograma inteiro fica concentrado em umas parte do eixo x como mostrado no exemplo abaixo.

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



Não conseguimos ver os valores para além de $y=10$ pois existem muitas observações no início da variável. Se limitarmos o eixo y para plotar apenas os valores menores do que 60, poderemos ver esses valores raros.

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



Observa-se que existem três valores raros próximo a 0, 30 e 60'. Para observá-los utilizamos o seguinte código.

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

price	x	y	z
5139	0.00	0.0	0.00
6381	0.00	0.0	0.00
12800	0.00	0.0	0.00
15686	0.00	0.0	0.00
18034	0.00	0.0	0.00
2130	0.00	0.0	0.00
2130	0.00	0.0	0.00
2075	5.15	31.8	5.12
12210	8.09	58.9	8.06

Missing Values

Para melhor observar a distribuição, podemos retirar da base os valores raros.

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

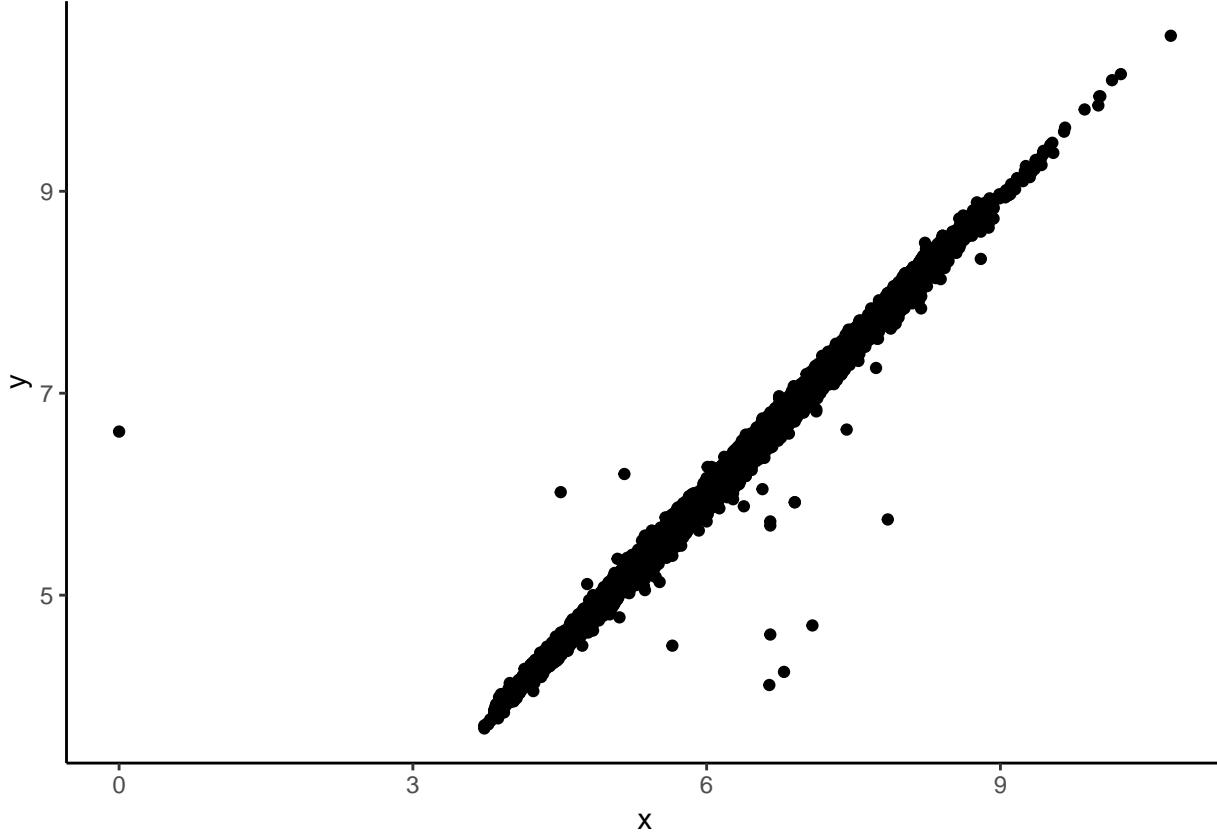
É mais recomendado substituir os valores raros por NA.

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

Ao plotar essa nova base, o ggplot2 remove os NAs mas, pelo menos avisa quantos dados foram removidos.

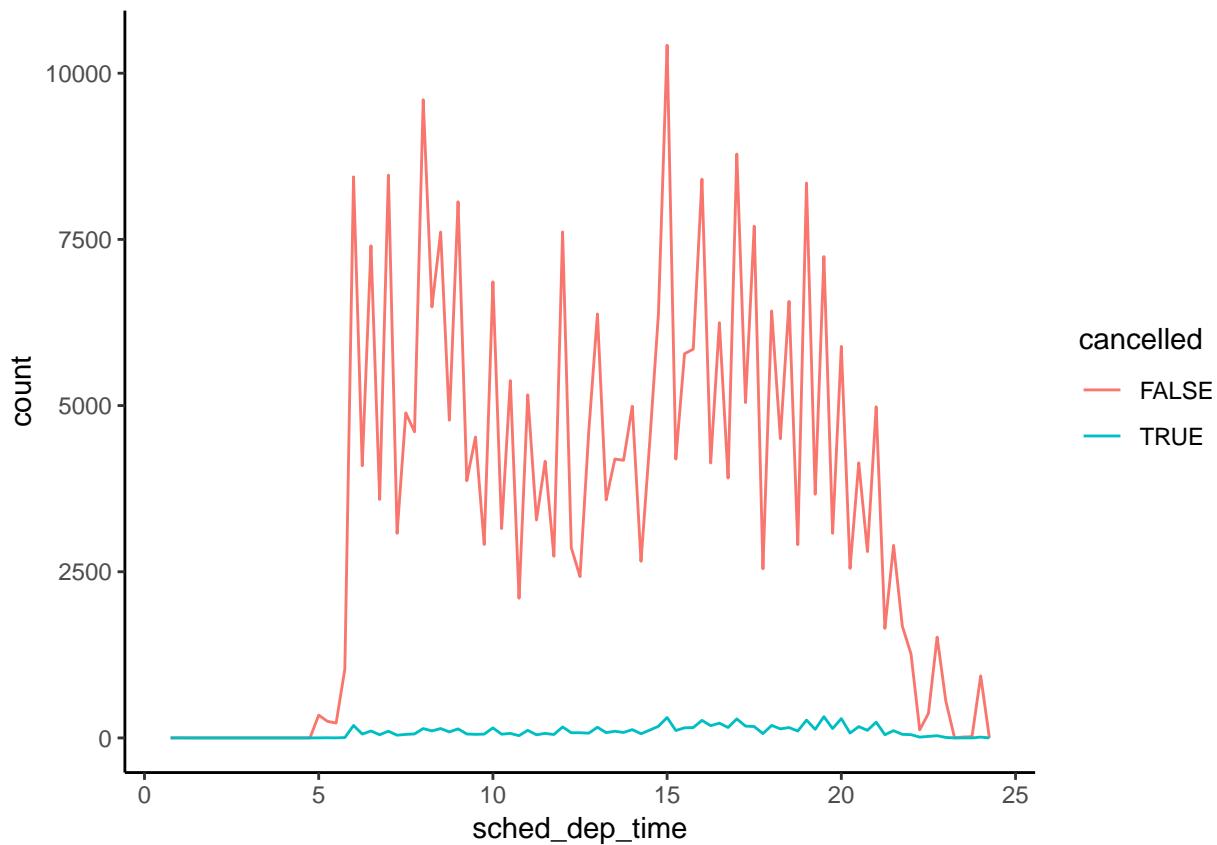
```
ggplot(data = diamonds2, mapping = aes(x = x, y = y)) +
  geom_point()
```

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



Nem sempre NA significa missing values. No exemplo abaixo, a base registra os voos feitos e NA significa voos cancelados. Para observá-los podemos utilizar a função is.na.

```
library(nycflights13)
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)
```

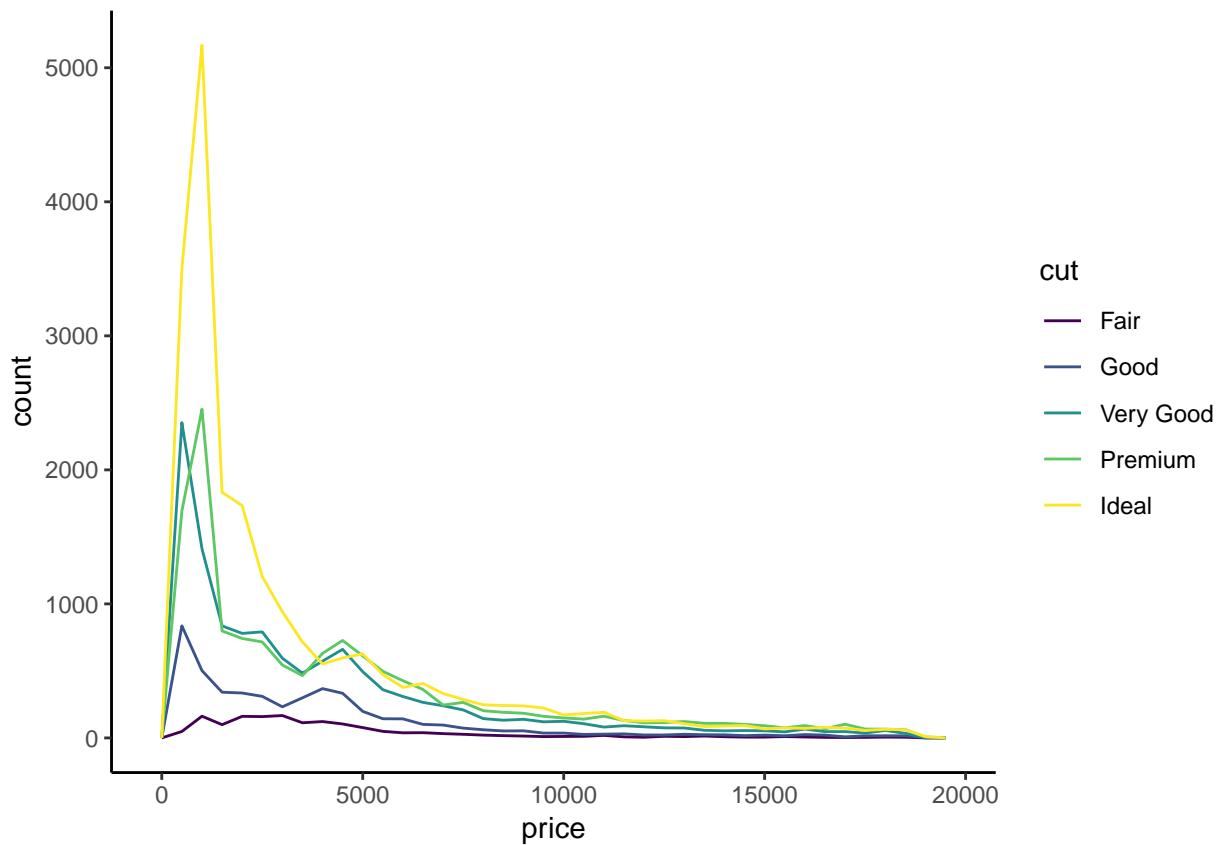


Covariação

Variáveis Categóricas e Ordinais

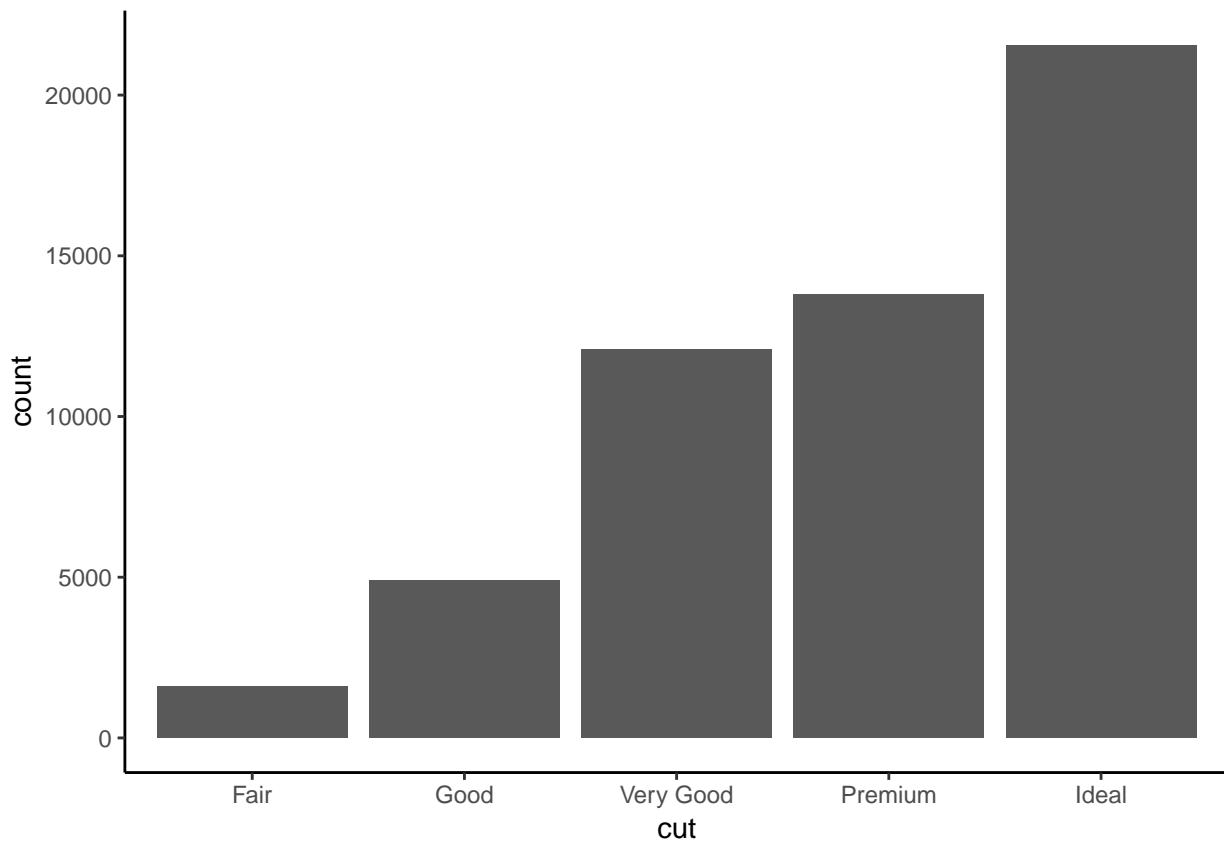
Analizando covariação entre variáveis categóricas e contínuas. Vamos observar a covariação do preço do diamante e a qualidade.

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



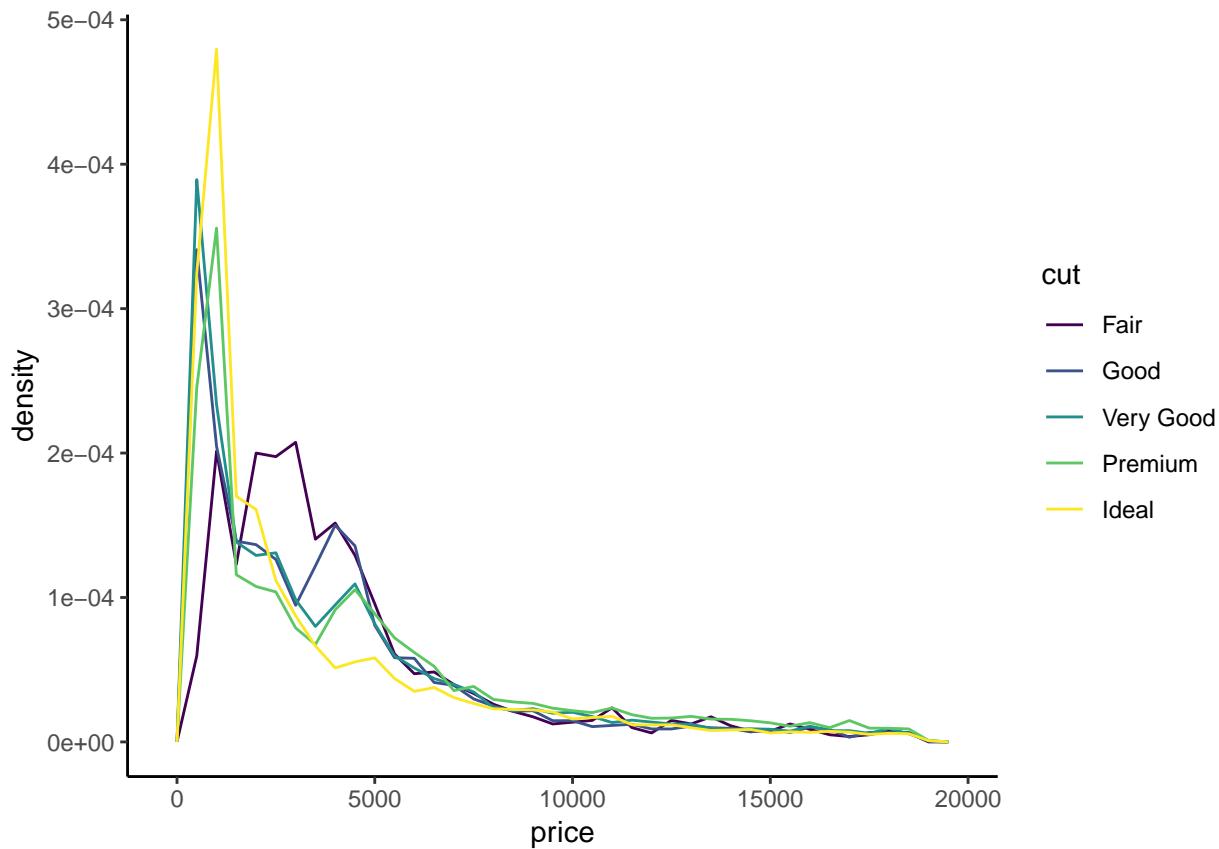
Difícil observar pois a quantidade varia muito.

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



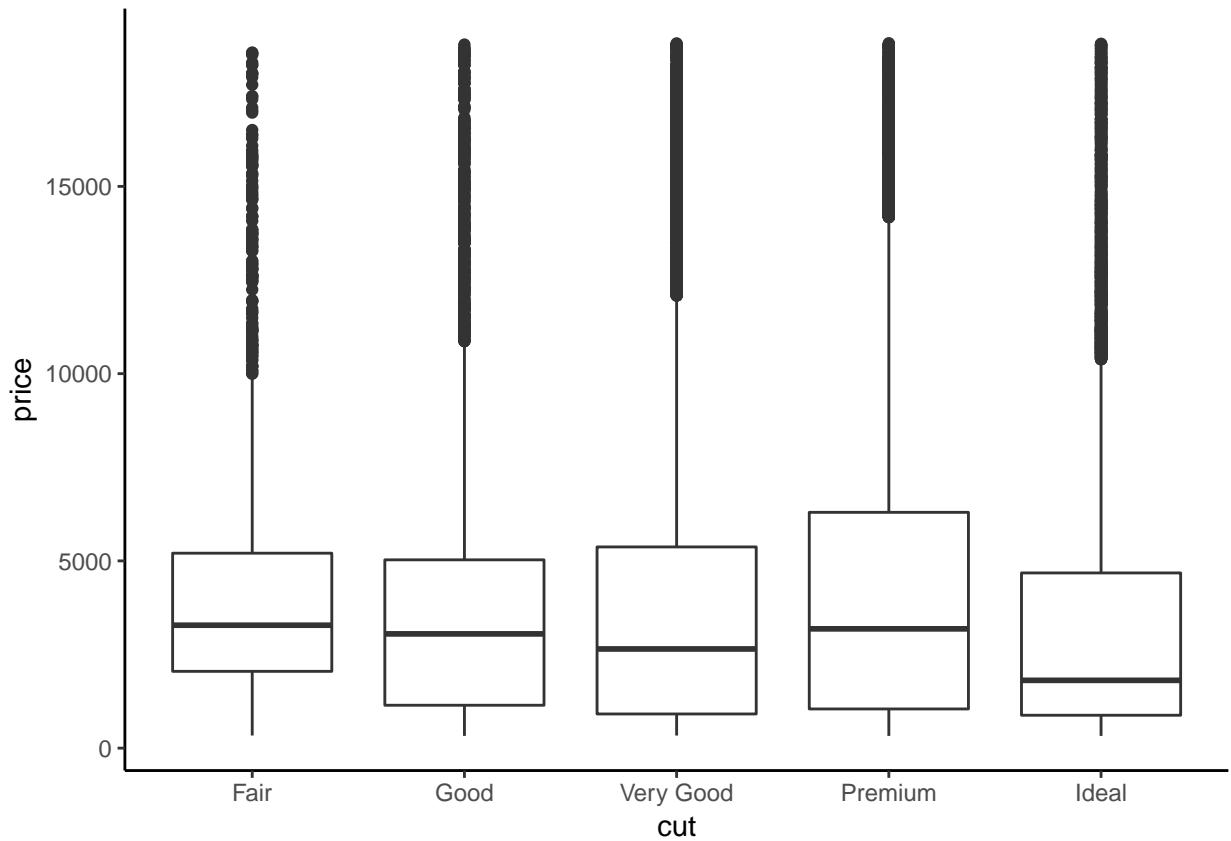
Vamos trocar o y pela densidade ao em vez de contagem.

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



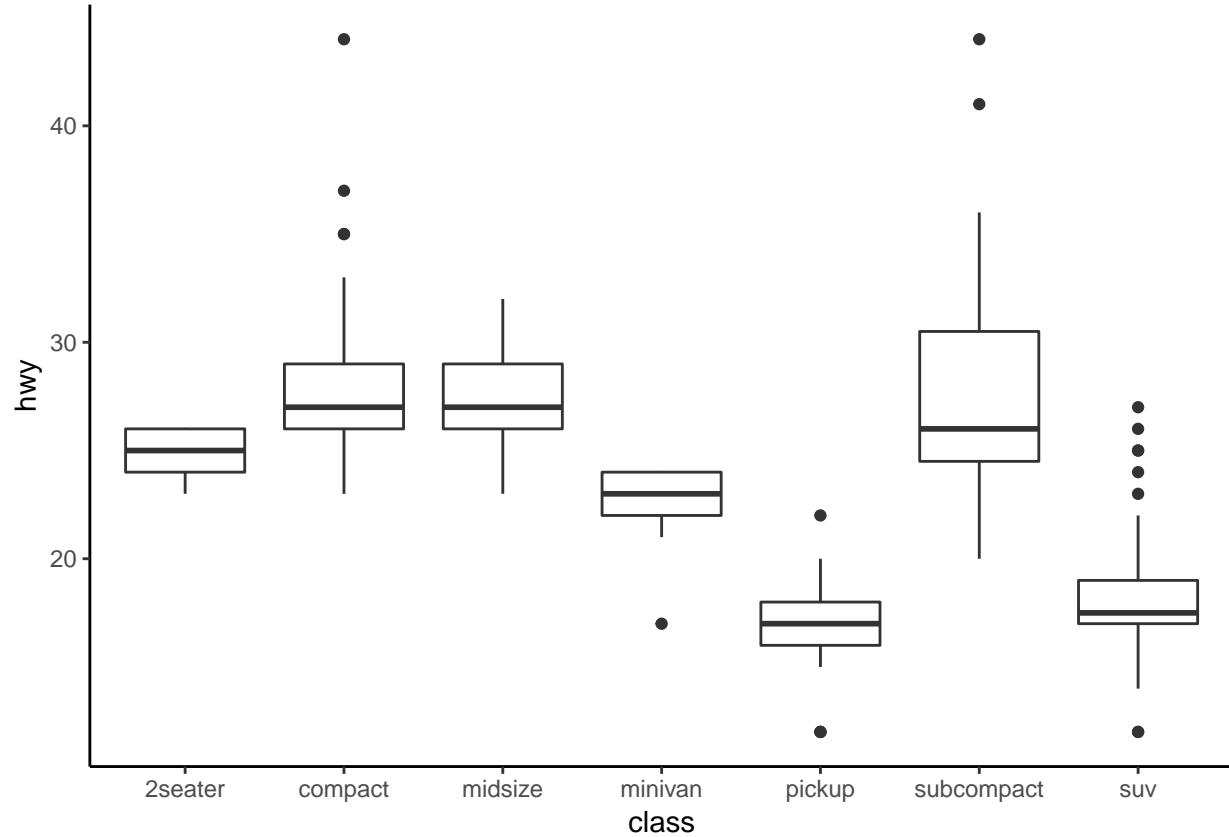
Outra maneira de observar essa relação é por boxplot.

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



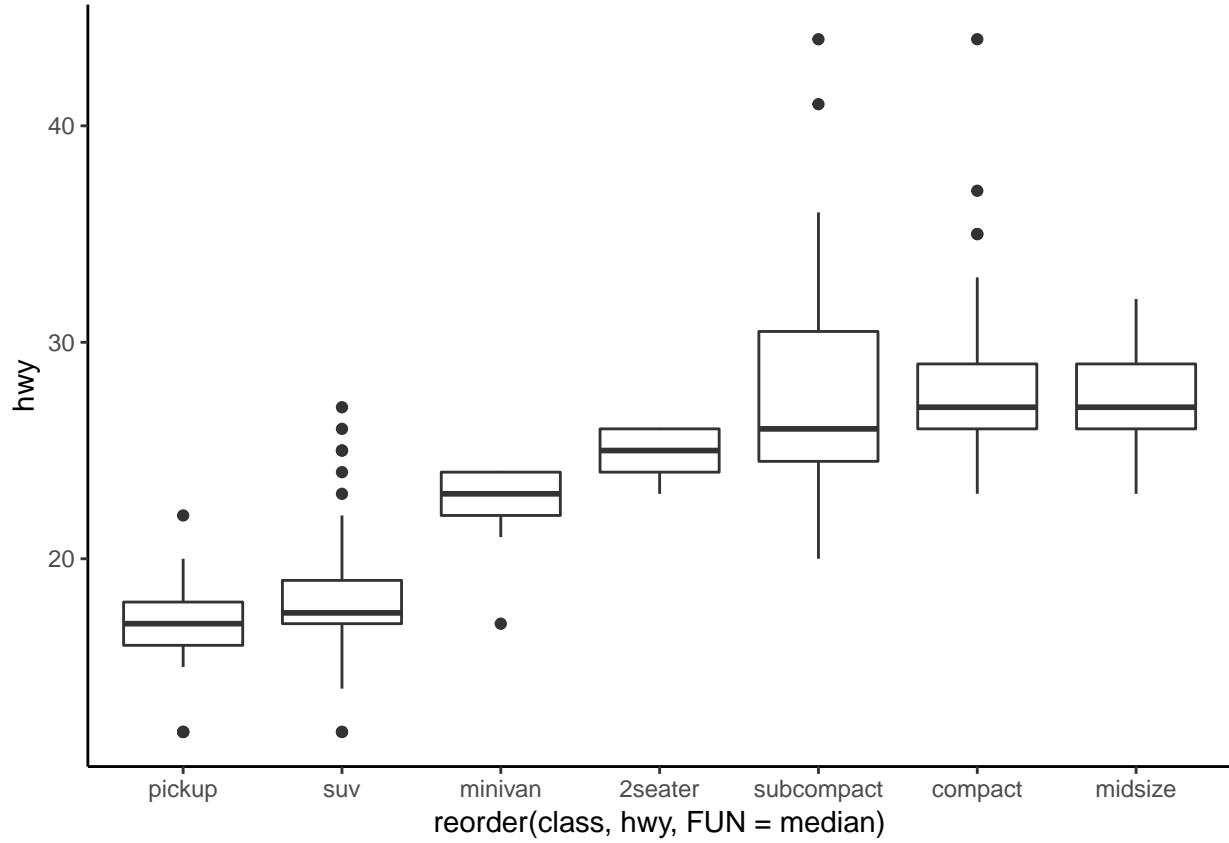
Vamos observar agora uma variável categórica que não é ordinal.

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



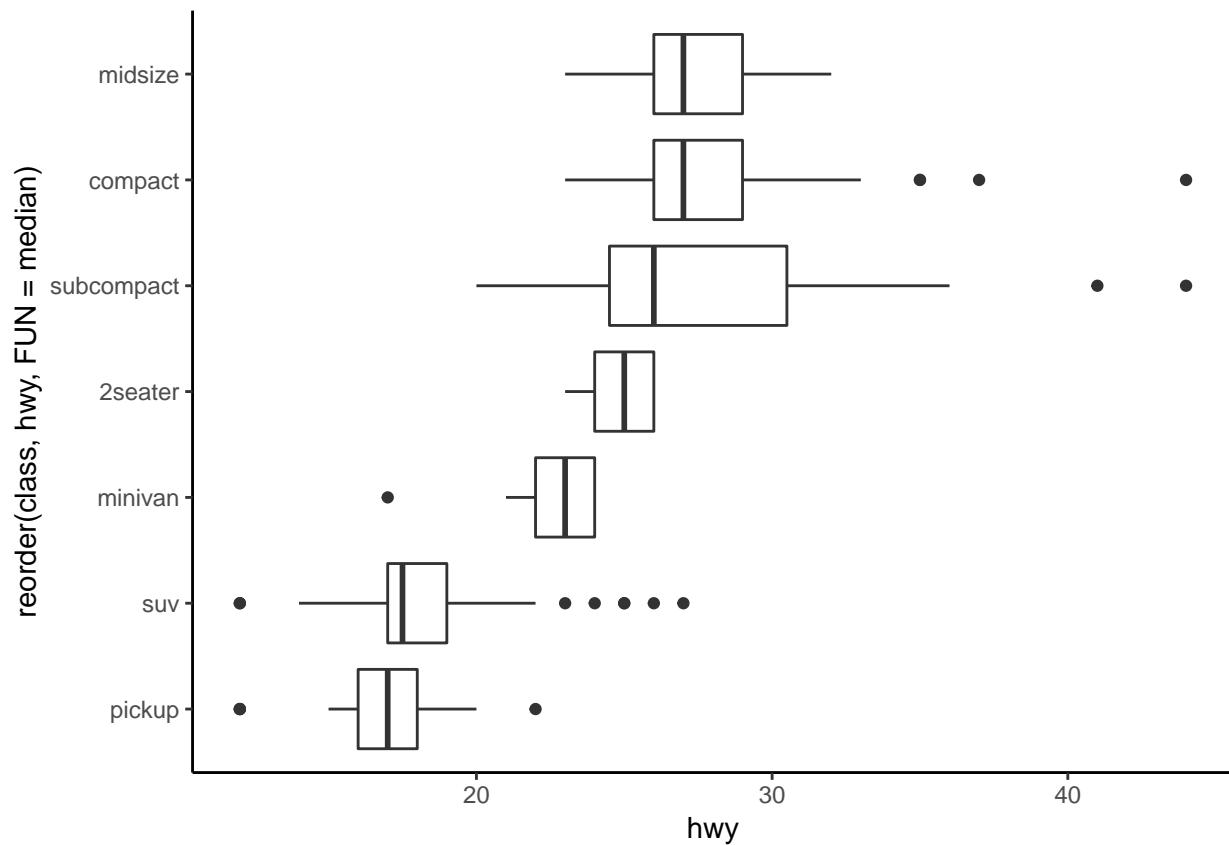
Para melhorar a visualização podemos reordenar o eixo x.

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



Caso o nome das variáveis sejam longas, podemos inverter os eixos.

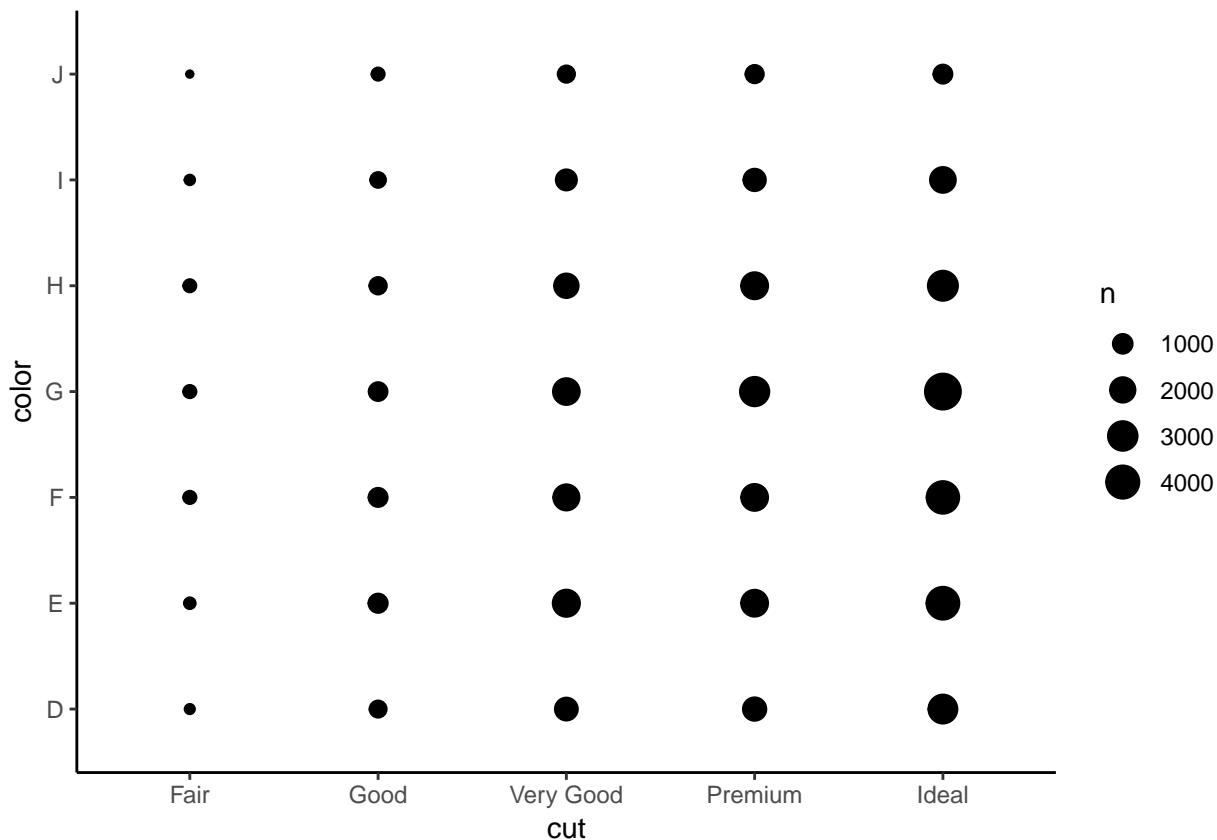
```
ggplot(data = mpg) +
  geom_boxplot(mapping = aes(x = reorder(class, hwy,
                                         FUN = median),
                             y = hwy)) +
  coord_flip()
```



Duas variáveis categóricas

Para observar duas variáveis categóricas precisamos contar o número de observações para cada categoria.

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



Cada círculo mostra quantas ocorrências combinadas houveram.

Podemos, também, calcular cada ocorrência combinada com dplyr.

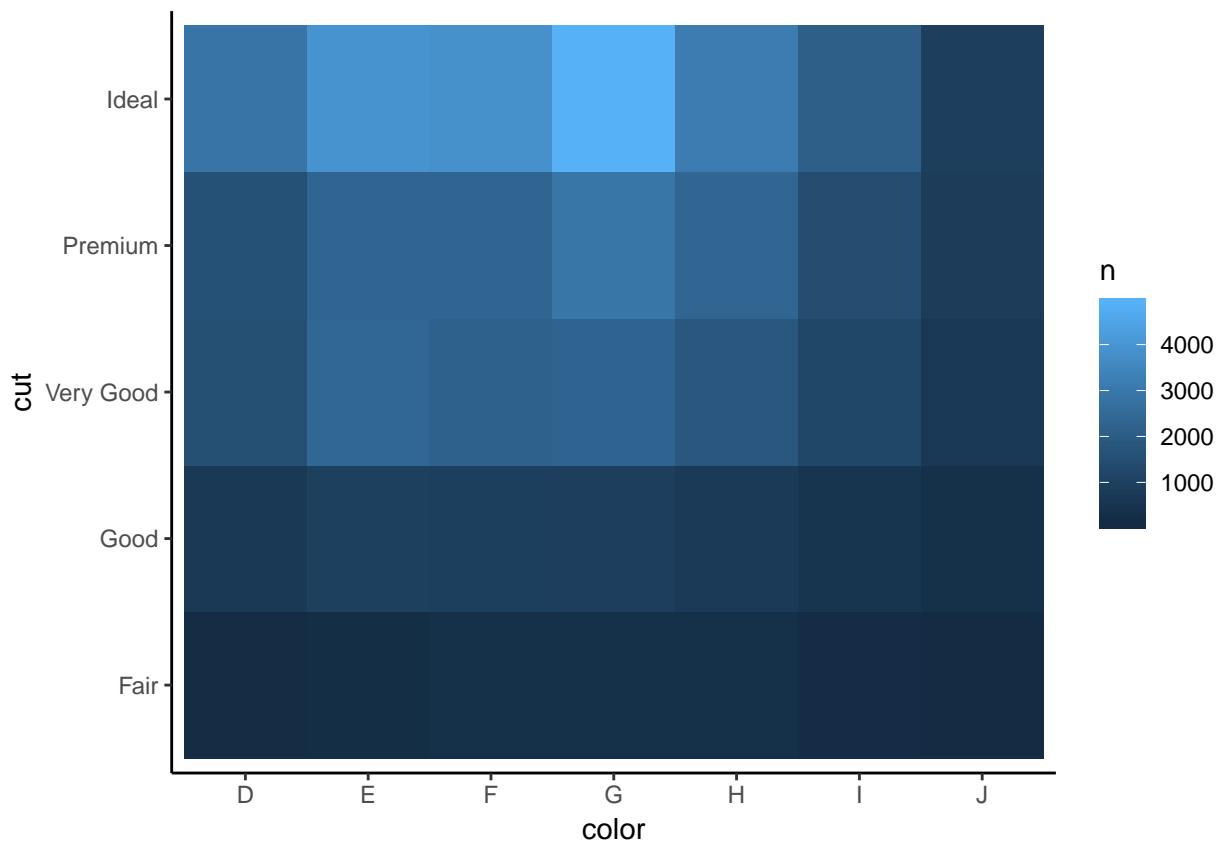
```
kable(diamonds %>%
      count(color, cut))
```

color	cut	n
D	Fair	163
D	Good	662
D	Very Good	1513
D	Premium	1603
D	Ideal	2834
E	Fair	224
E	Good	933
E	Very Good	2400
E	Premium	2337
E	Ideal	3903
F	Fair	312
F	Good	909
F	Very Good	2164
F	Premium	2331
F	Ideal	3826
G	Fair	314
G	Good	871
G	Very Good	2299
G	Premium	2924

color	cut	n
G	Ideal	4884
H	Fair	303
H	Good	702
H	Very Good	1824
H	Premium	2360
H	Ideal	3115
I	Fair	175
I	Good	522
I	Very Good	1204
I	Premium	1428
I	Ideal	2093
J	Fair	119
J	Good	307
J	Very Good	678
J	Premium	808
J	Ideal	896

Outra maneira seria com uma especie de mapa de calor. O geom_tile.

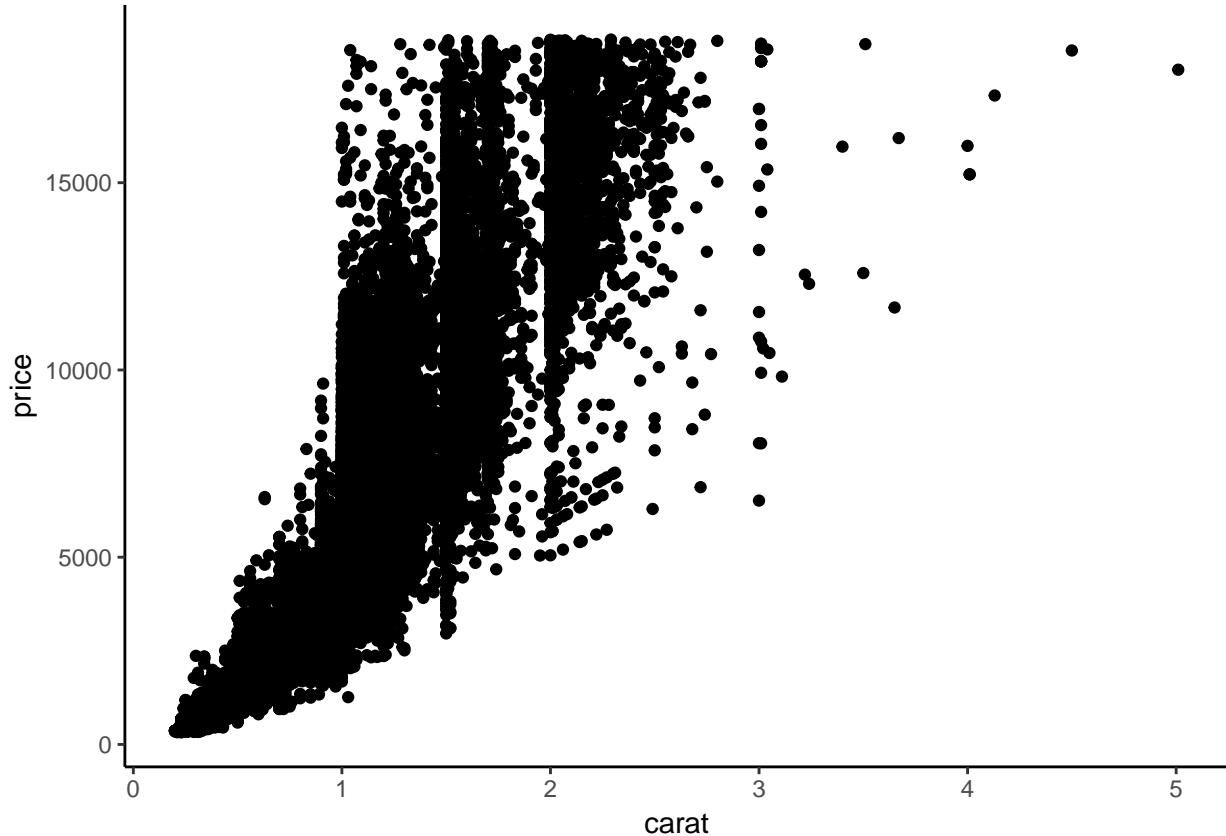
```
diamonds %>%
  count(color, cut) %>%
  ggplot(mapping = aes(x = color, y = cut)) +
  geom_tile(mapping = aes(fill = n))
```



Duas variáveis contínuas

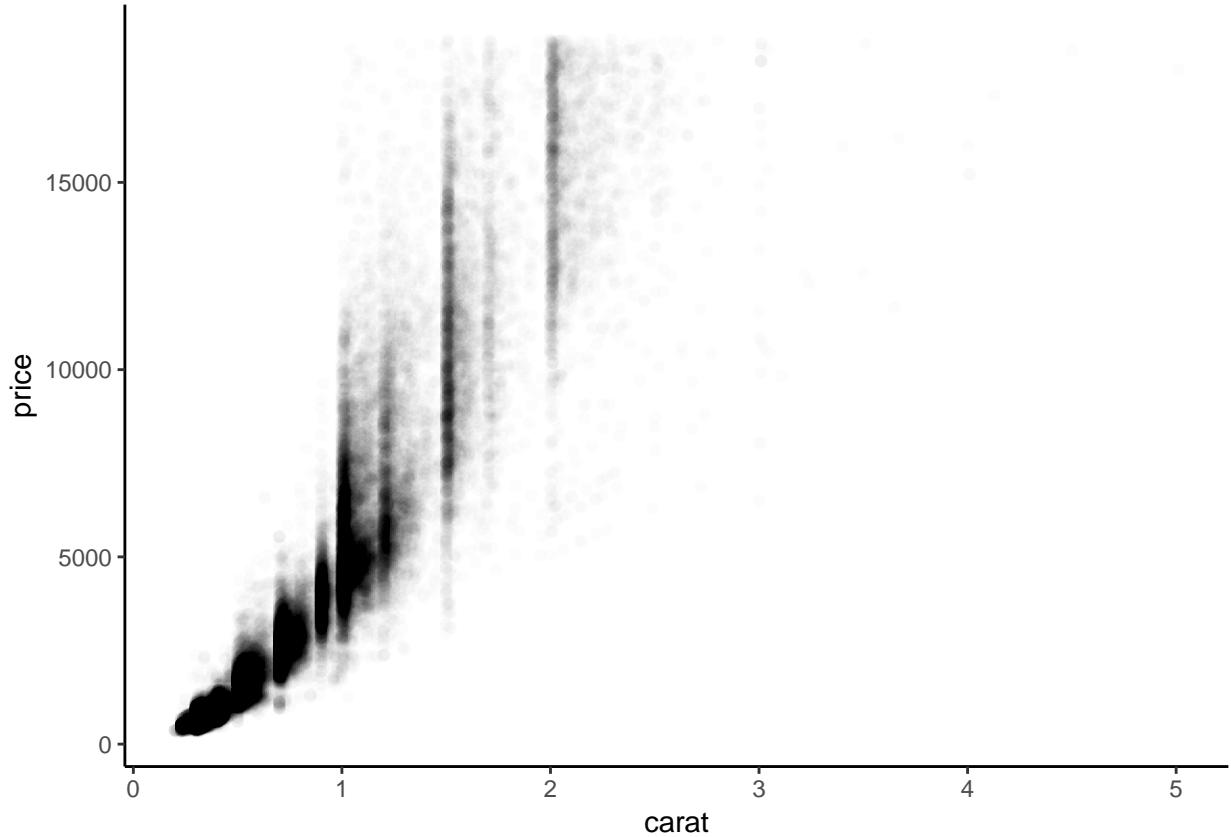
Para duas variáveis contínuas podemos observar um gráfico de dispersão de pontos.

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



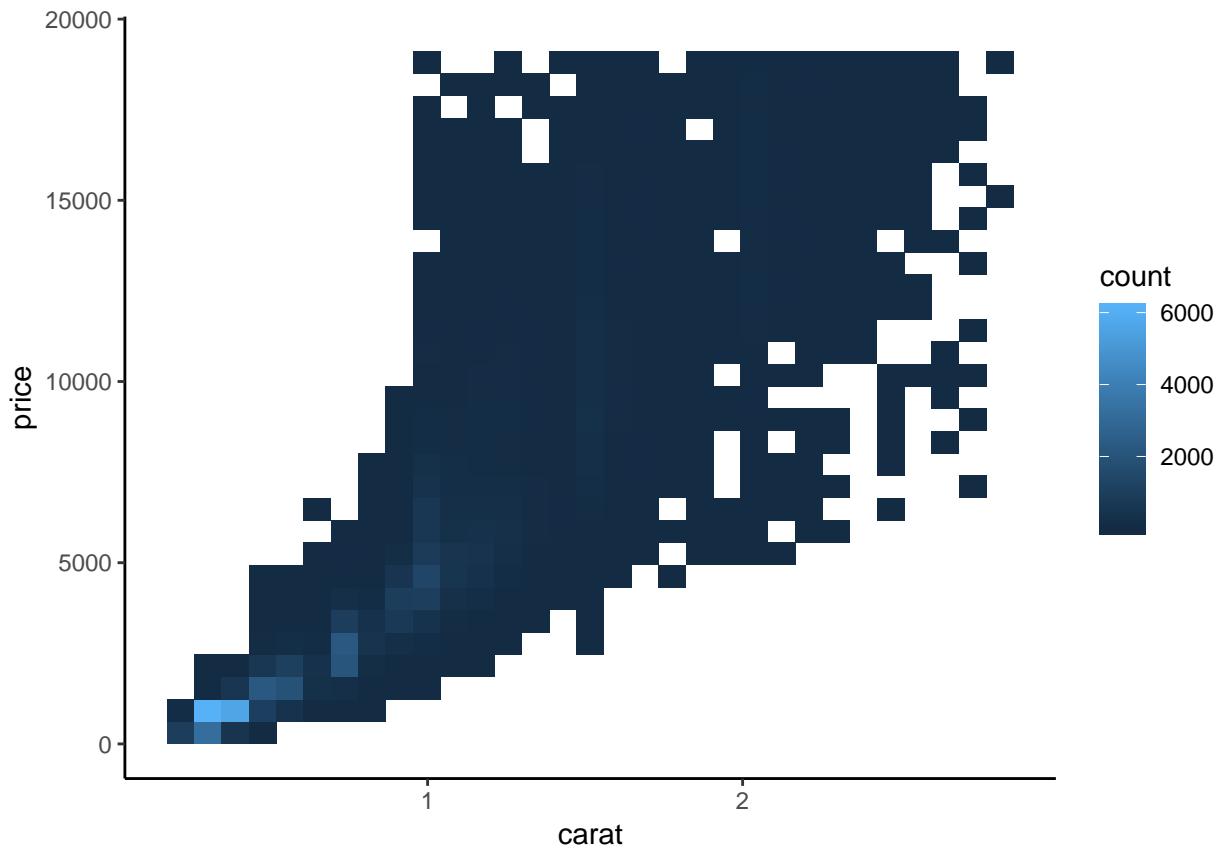
Esse gráficos se tornam menos úteis quanto maior for a quantidade de dados. Podemos adicionar transparência aos pontos para o gráfico ficar mais interessante.

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

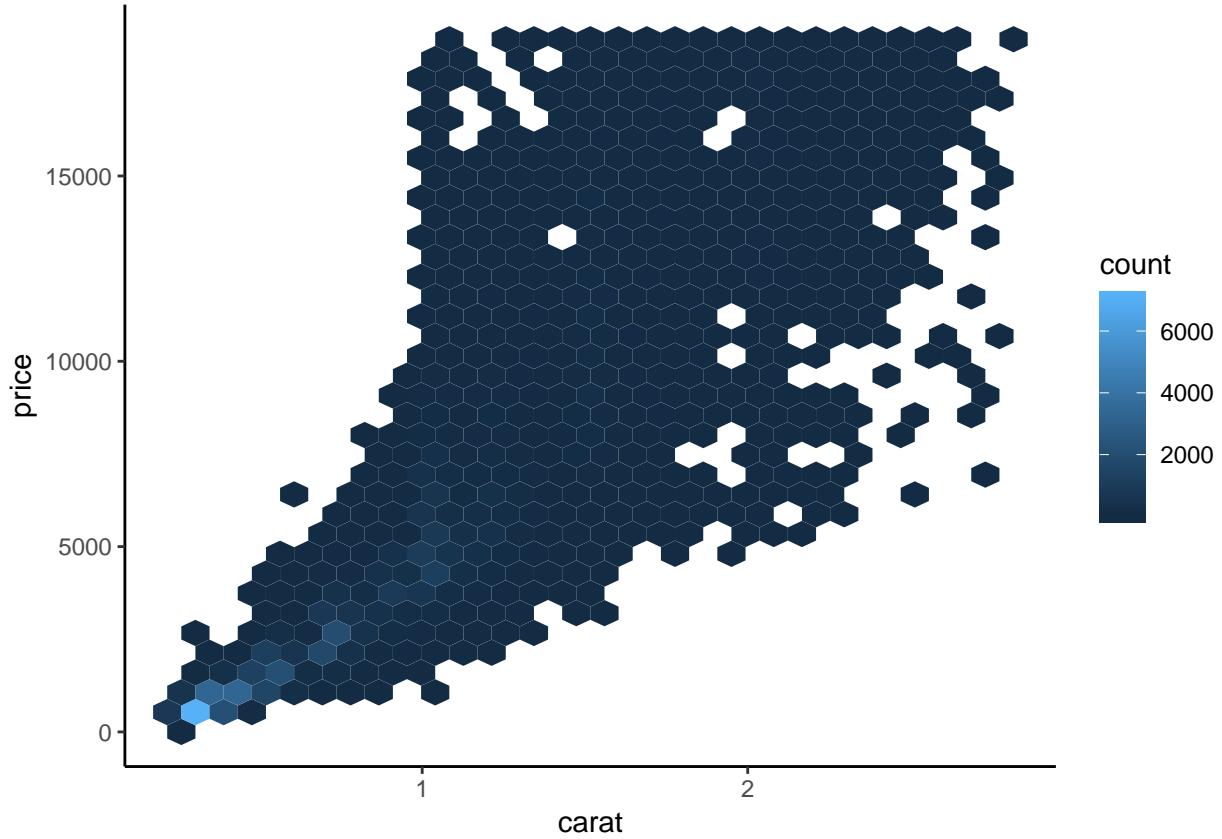


Uma maneira mais interessante ainda de observar grandes bases de dados é utilizar o geom_bin2d e o geom_hex. Com ele, os pontos que se sobrepõem recebem cores diferentes.

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

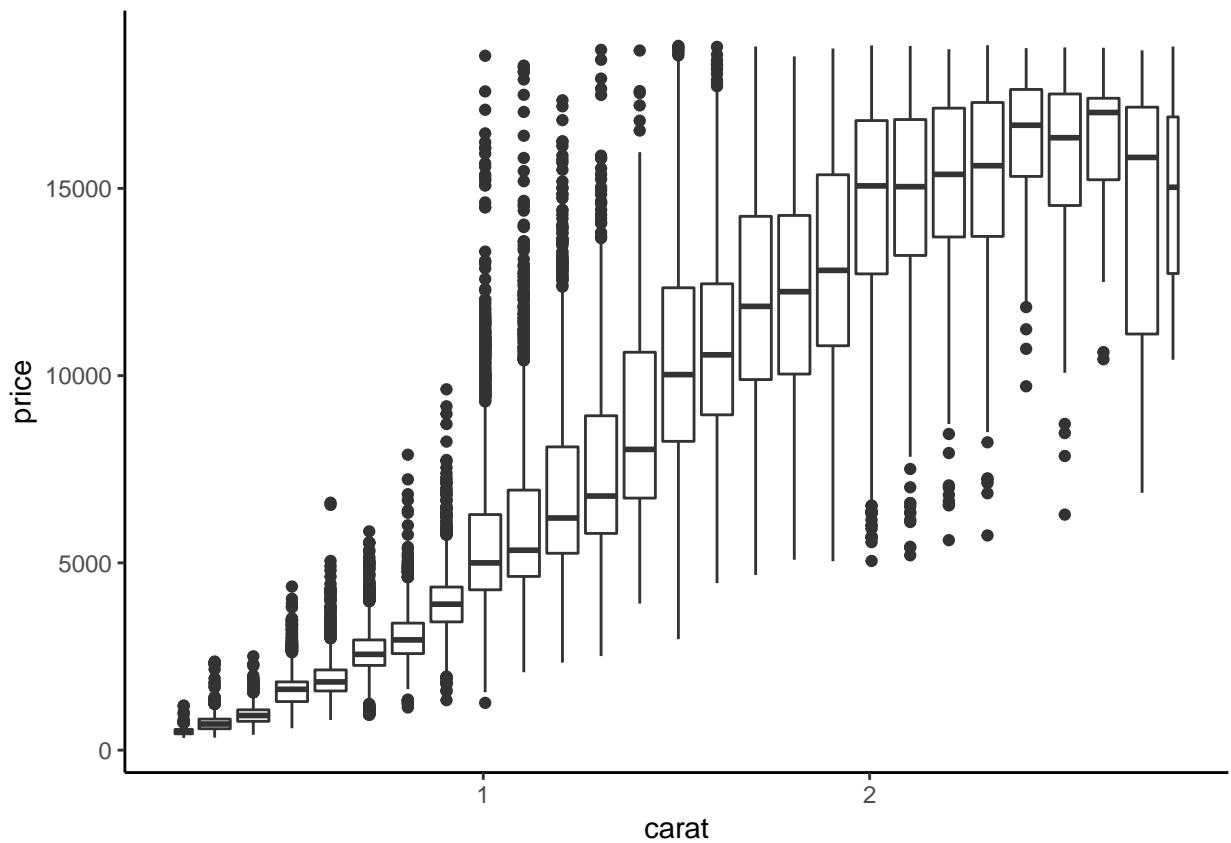


```
library("hexbin")  
  
ggplot(data = smaller) +  
  geom_hex(mapping = aes(x = carat, y = price))
```



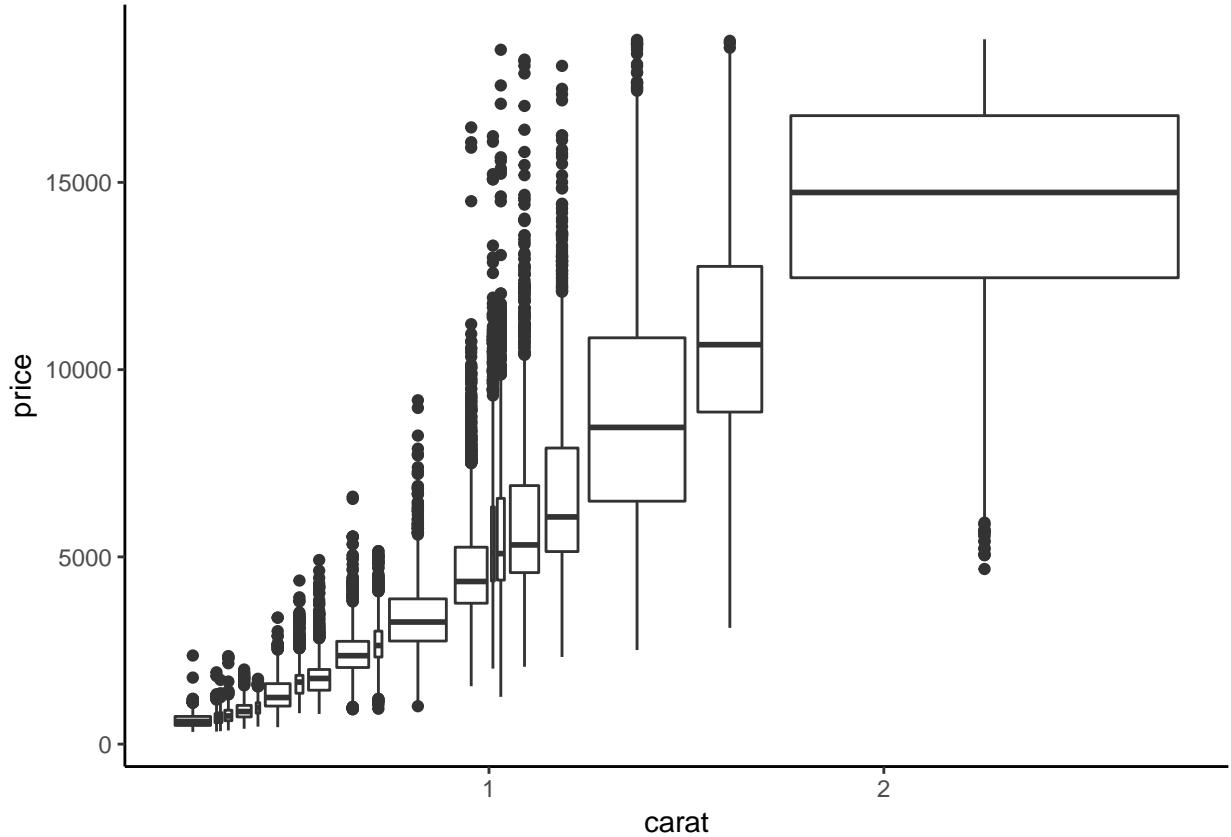
Podemos, ainda, tratar uma variável contínua como uma categórica e plotar boxplots. Com o `cut_width` podemos agrupar os dados.

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



Podemos também plotar em cada boxplot o mesmo número de pontos.

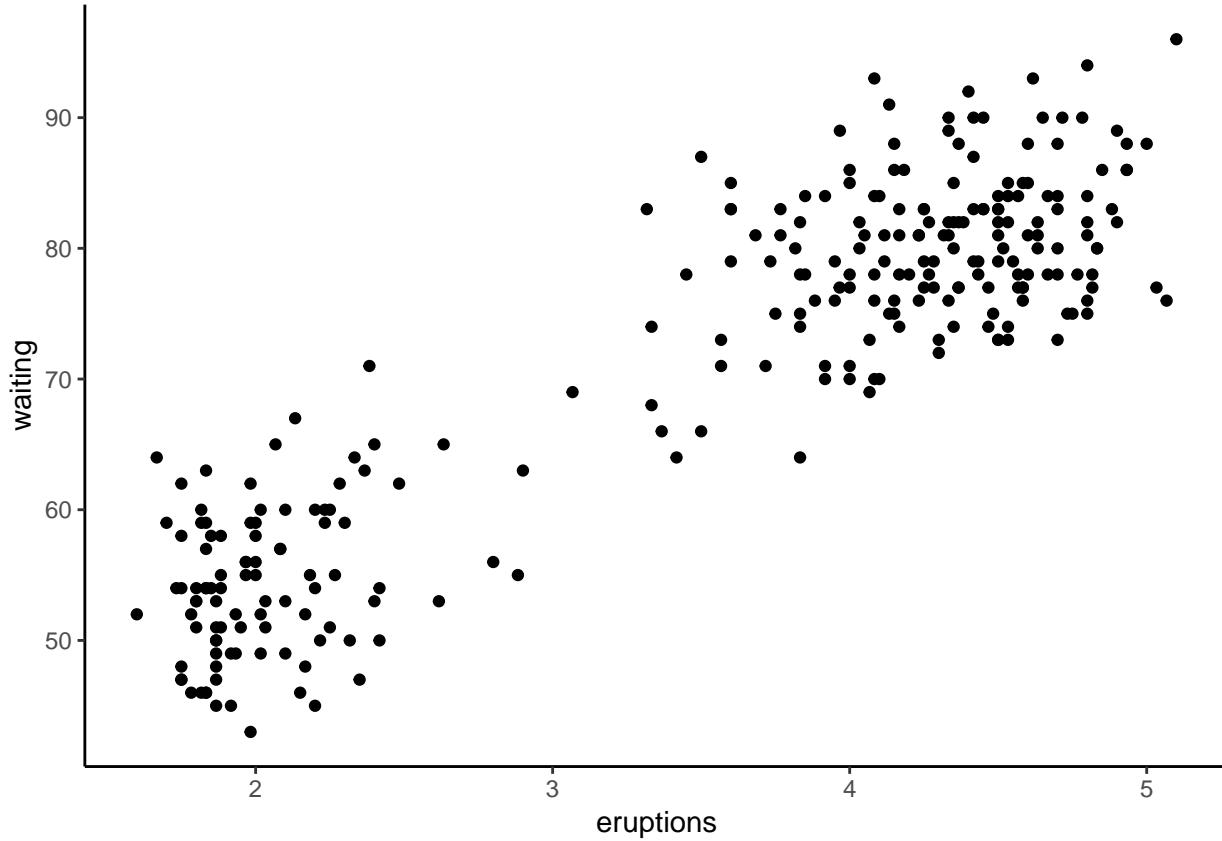
```
ggplot(data = smaller, mapping = aes(x = carat, y = price)) +  
  geom_boxplot(mapping = aes(group = cut_number(carat, 20)))
```



Padrões e Modelos

Padrões podem ser visualizados ao plotar os nossos dados. Neste exemplo, claramente observa-se dois clusters.

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



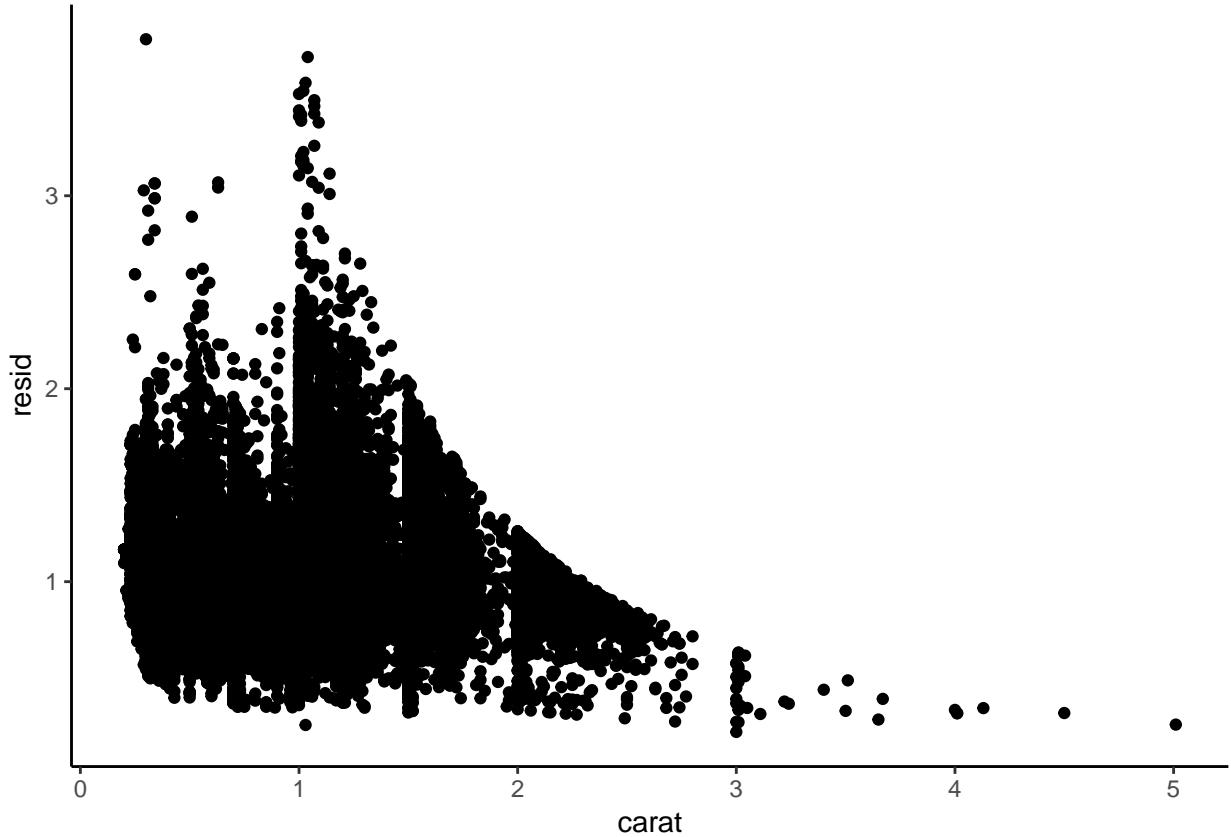
As vezes é difícil entender a relação entre duas variáveis, pois outra variável pode está correlacionada com ambas, variáveis de interesse. Um exemplo disso é a relação entre a qualidade do diamante e o preço pois a variável peso está relacionada tanto com o preço quanto com a qualidade do diamante. Podemos retirar a relação entre peso e preço do diamante ao substituímos essa relação pelo seu resíduo. Com os resíduos podemos observar o preço do diamante sem o efeito do peso dele.

```
library(modelr)

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

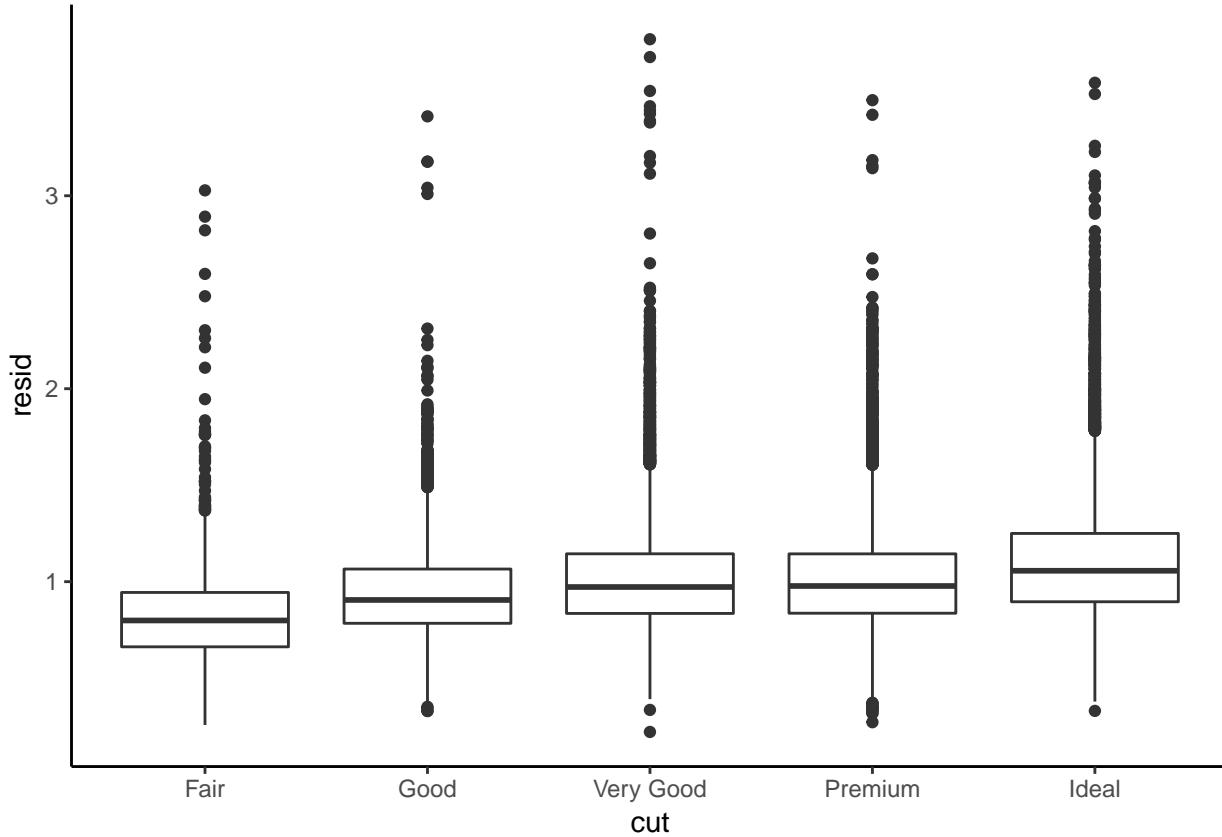
diamonds2 <- diamonds %>%
  add_residuals(mod) %>%
  mutate(resid = exp(resid))

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



Uma vez retirada essa relação do peso com o preço, podemos de fato observar a relação entre qualidade e preço dos diamantes.

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



10. Com os dados disponibilizados na plataforma (vote_growth_usa.RData), reproduza os resultados do livro Kellstedt, P. M., & Whitten, G. D. (2013) utilizando o código apresentado nos slides da aula.

```

load("vote_growth_usa.RData")
reg <- lm(Vote ~ Growth, data = bd)
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

```

11. Com os dados e as variáveis do exercício 10, realize uma análise de regressão considerando apenas o período de 1876 a 1932. Apresente os resultados e os compare quanto ao modelo completo (exercício 10) em relação a:

```

bd_select <- bd[1:15, ]
reg2 <- lm(Vote ~ Growth, data = bd_select)
summary(reg2)

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

```

a) Significância estatística dos resultados;

O nível de significância estatística medido pelo p-valor diminui do primeiro modelo (completo) para o segundo (1876-1932). No primeiro, o efeito de crescimento no comparecimento eleitoral foi de 0,62 com p-valor menor que 0,000. Já no segundo, o efeito foi menor, de 0,53 com p-valor de 0,042.

b) Intervalo de confiança para β ;

O intervalo de confiança é o Growth estimado (0,5336) mais ou menos o t valor * o erro padrão. Como em ambos os casos o intervalo não passa pelo 0, os dois modelos passam no nível de significância de 95% com p-valor menor que 0,05.

c) Medidas de ajuste do modelo;

O ajuste do modelo é medido pelo R^2 . O primeiro modelo é melhor ajustado do que o segundo pois o R^2 dele foi de 0,3291, enquanto do segundo foi de 0,2811. O R^2 ajustado é uma medida mais precisa pois não é tão sensível a quantidade de variáveis existente no modelo, sendo assim, uma medida melhor para comparar modelos diferentes. Analisando o R^2 ajustado, o primeiro modelo (0,3081) ainda é melhor do que o segundo modelo (0,2258).