

Lab04

Vanderlin Amorim Júnior - Curso Verão 2016 - Curso R

25 de fevereiro de 2016

Aula 04 - Laboratório

Carregando as bibliotecas

```
library(ggplot2)
library(dplyr)

## 
## 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

library(tidyr)
```

Lendo a estrutura da base de dados diamonds e também de outras funções

```
help(diamonds)

## starting httpd help server ...

## done

?geom_point
?xlab()
```

1. Quais são os aspectos estéticos (aesthetics) exigidos (obrigatórios) da função geom_point()?

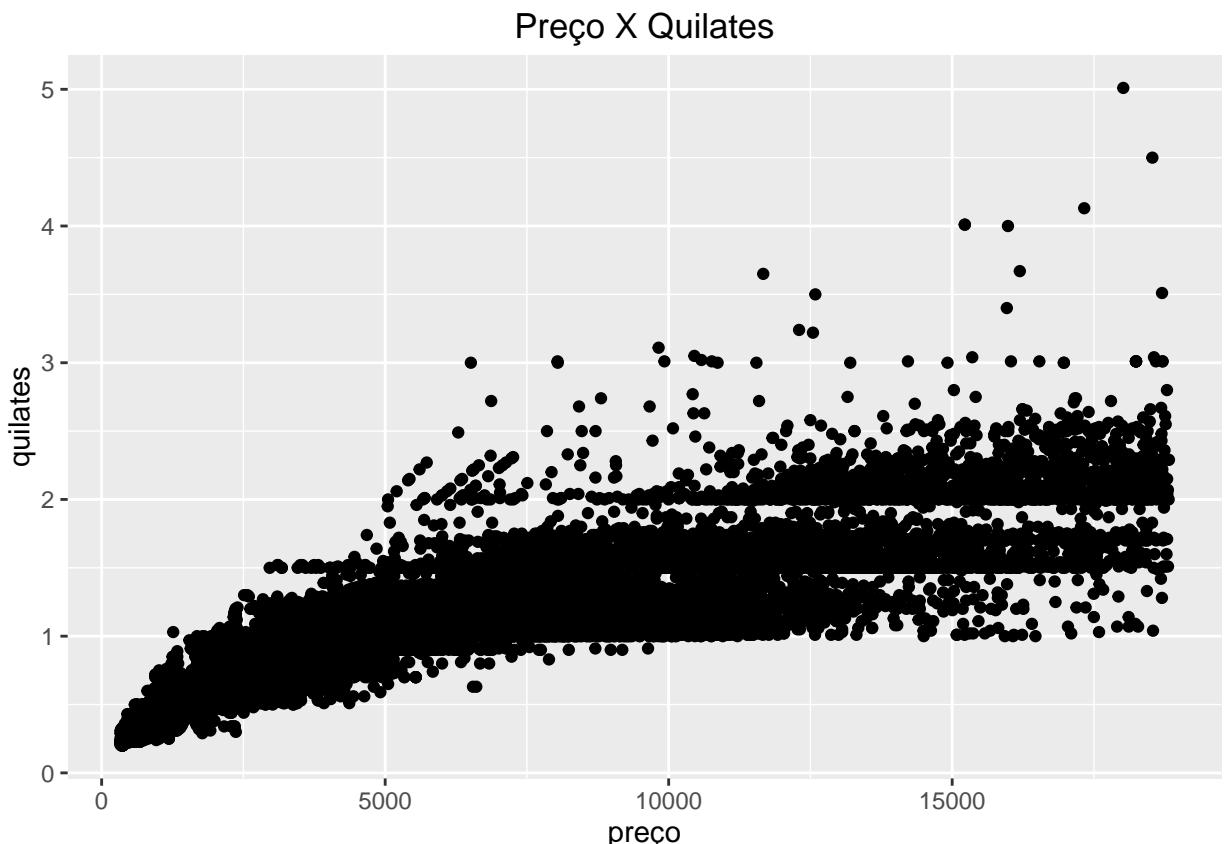
A primeira camada de um gráfico deve indicar a relação entre os dados e cada aspecto visual do gráfico, como qual variável será representada no eixo x, qual será representada no eixo y, a cor e o tamanho dos componentes geométricos etc. Deve ser fixado pelo menos uma cor(color) ou tamanho (size)

Dica: utilizar a função help().

2. Faça um gráfico de dispersão do preço (price) pela variável quilates (carat).

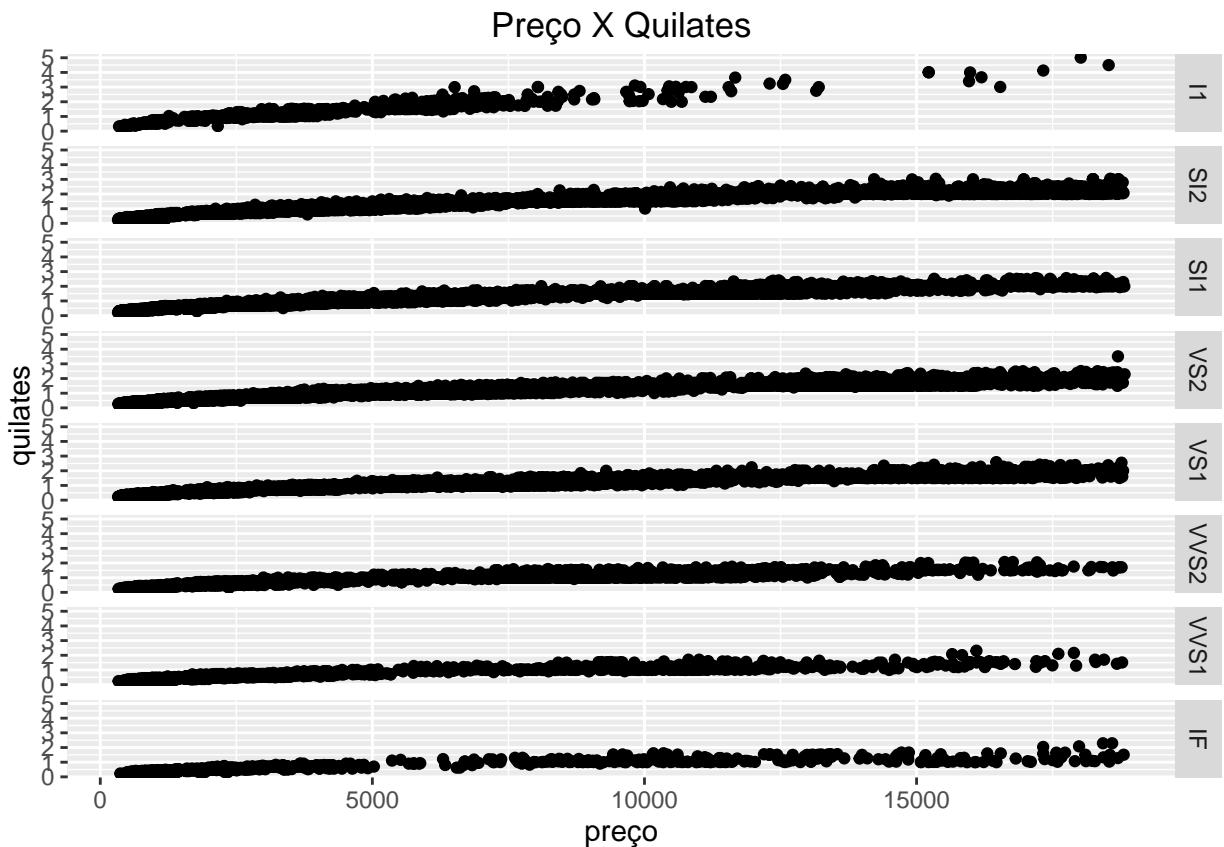
Utilize as funções xlab() e ylab() para trocar os labels dos eixos x e y, respectivamente.

```
ggplot(data = diamonds, aes(x = price, y = carat)) +  
  geom_point() + xlab("preço") + ylab("quilates") + labs(title = "Preço X Quilates")
```

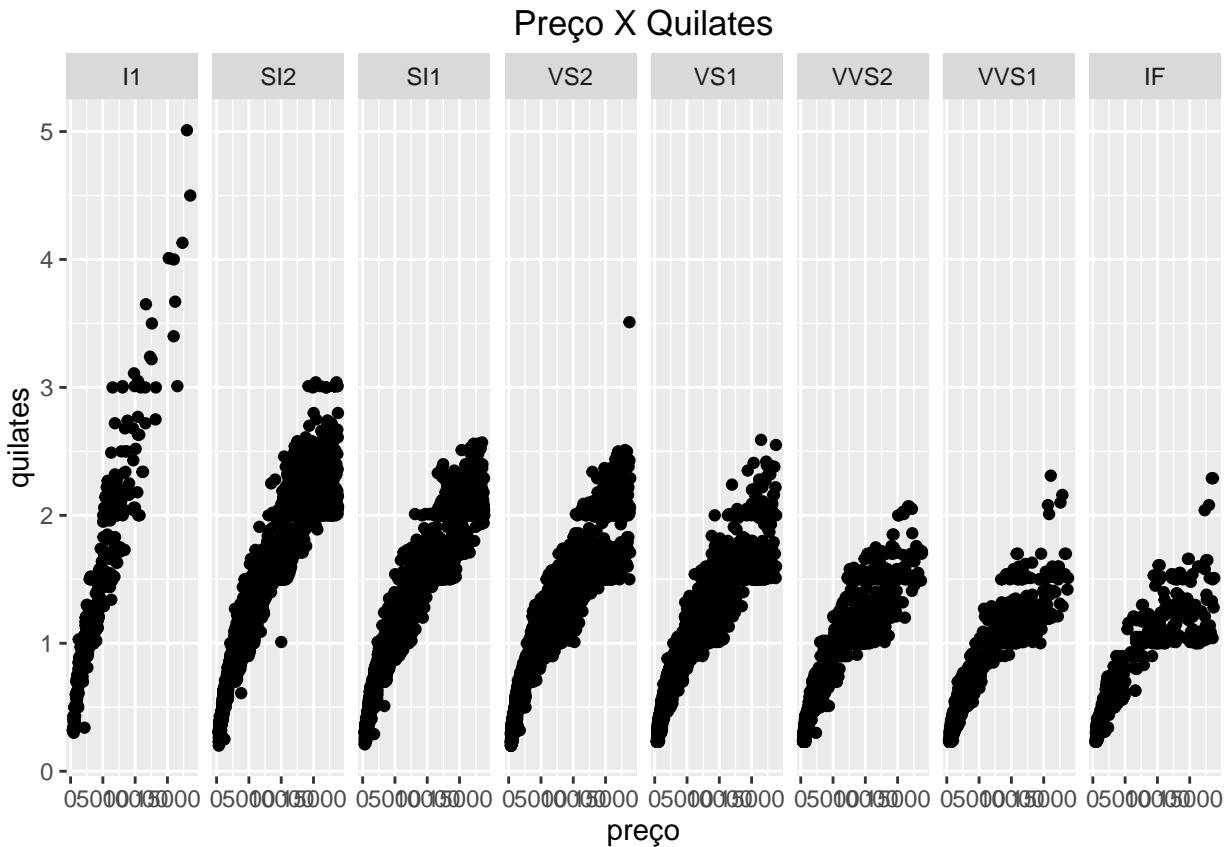


3. Utilize a facets para fazer gráficos de dispersão do preço pela variável quilate (o mesmo gráfico do exercício 1) para cada nível da variável claridade (clarity).

```
ggplot(data = diamonds, aes(x = price, y = carat)) +  
  geom_point() + xlab("preço") + ylab("quilates") + labs(title = "Preço X Quilates") +  
  facet_grid(clarity ~ .)
```



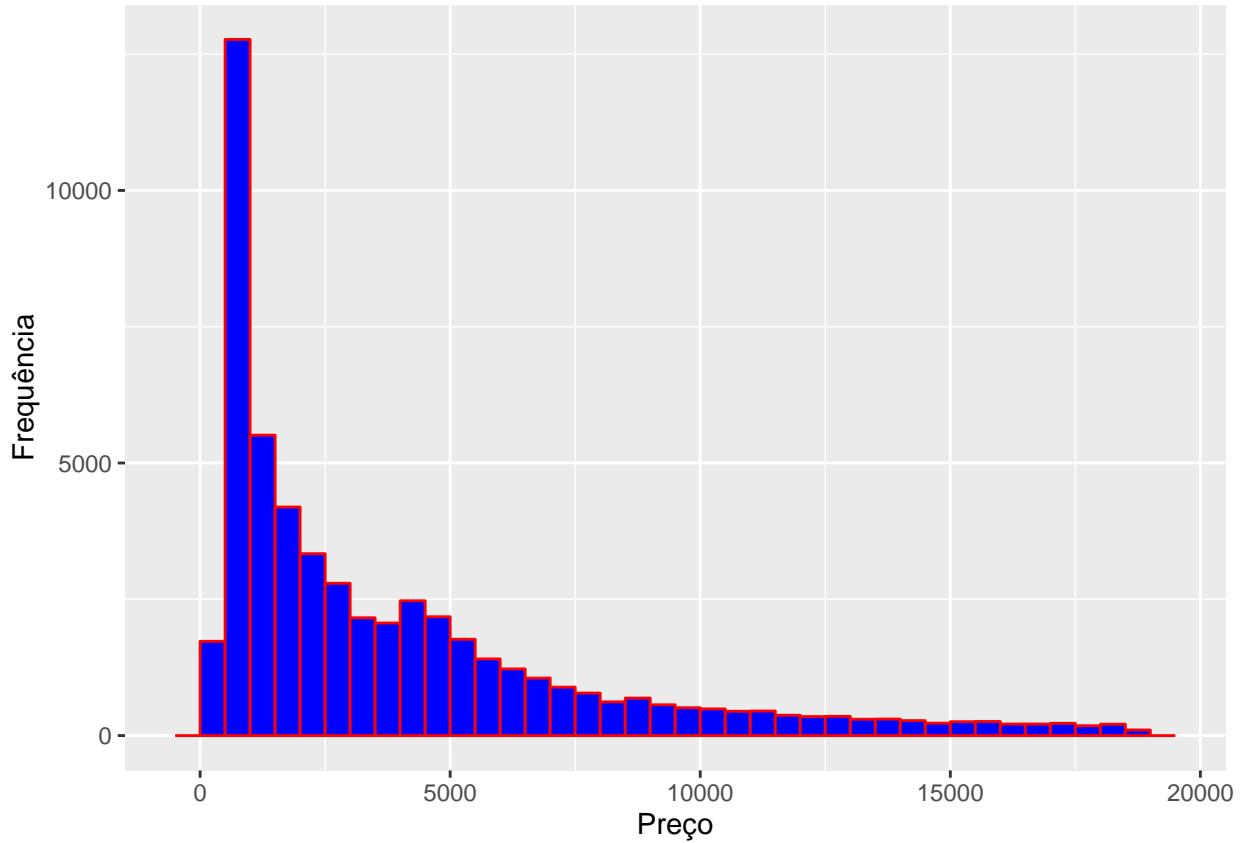
```
ggplot(data = diamonds, aes(x = price, y = carat)) +  
  geom_point() + xlab("preço") + ylab("quilates") + labs(title = "Preço X Quilates") +  
  facet_grid(. ~ clarity)
```



`geom_histogram`

4. Quais são os aspectos estéticos (aesthetics) exigidos (obrigatórios) da função `geom_histogram()`?
5. Faça um histograma da variável preço. Modifique a cor das barras de forma que elas não pareçam grudadas umas nas outras.

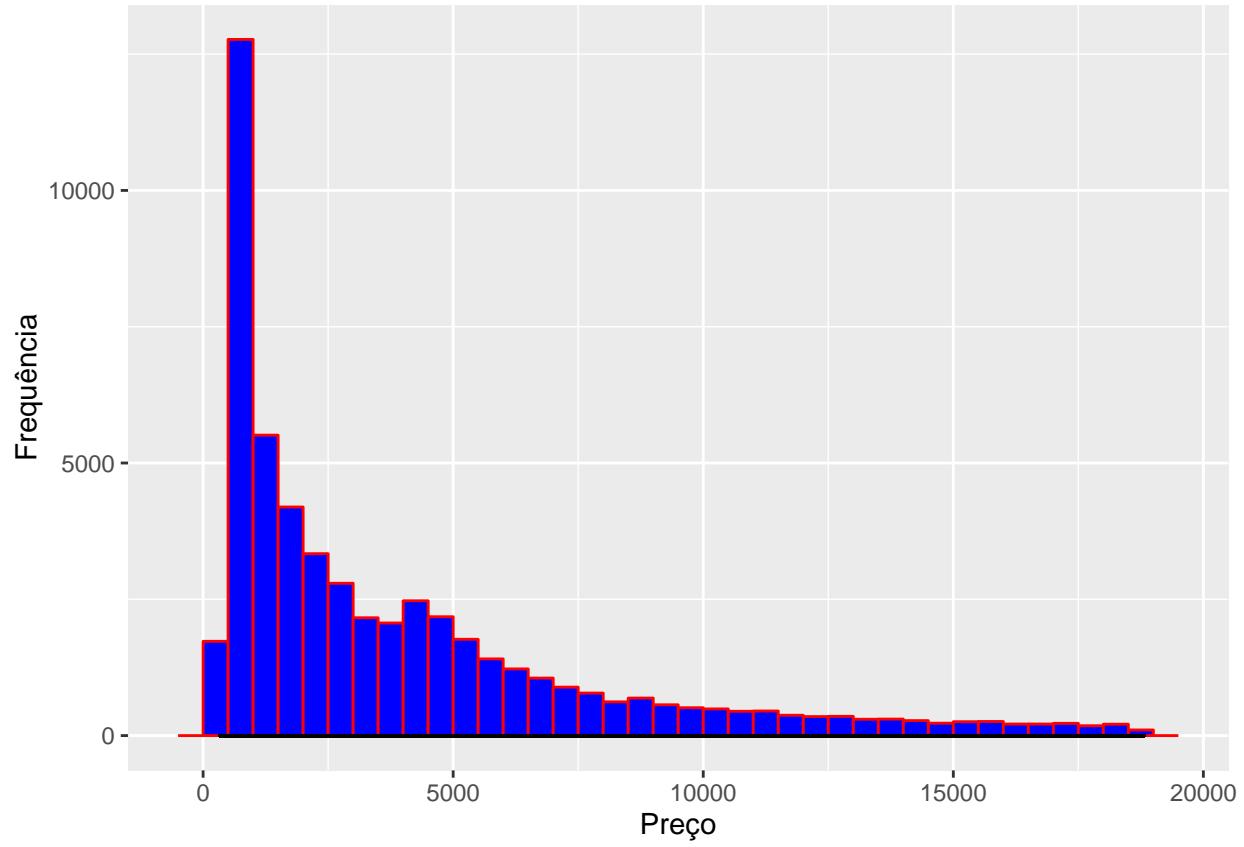
```
ggplot(data = diamonds, aes(x = price)) +
  geom_histogram(fill=I("blue"), binwidth = 500, col=I("red")) +
  xlab("Preço") +
  ylab("Frequência")
```



6. Utilize a função `geom_density()` para adicionar ao gráfico anterior uma estimativa

suavizada da densidade.

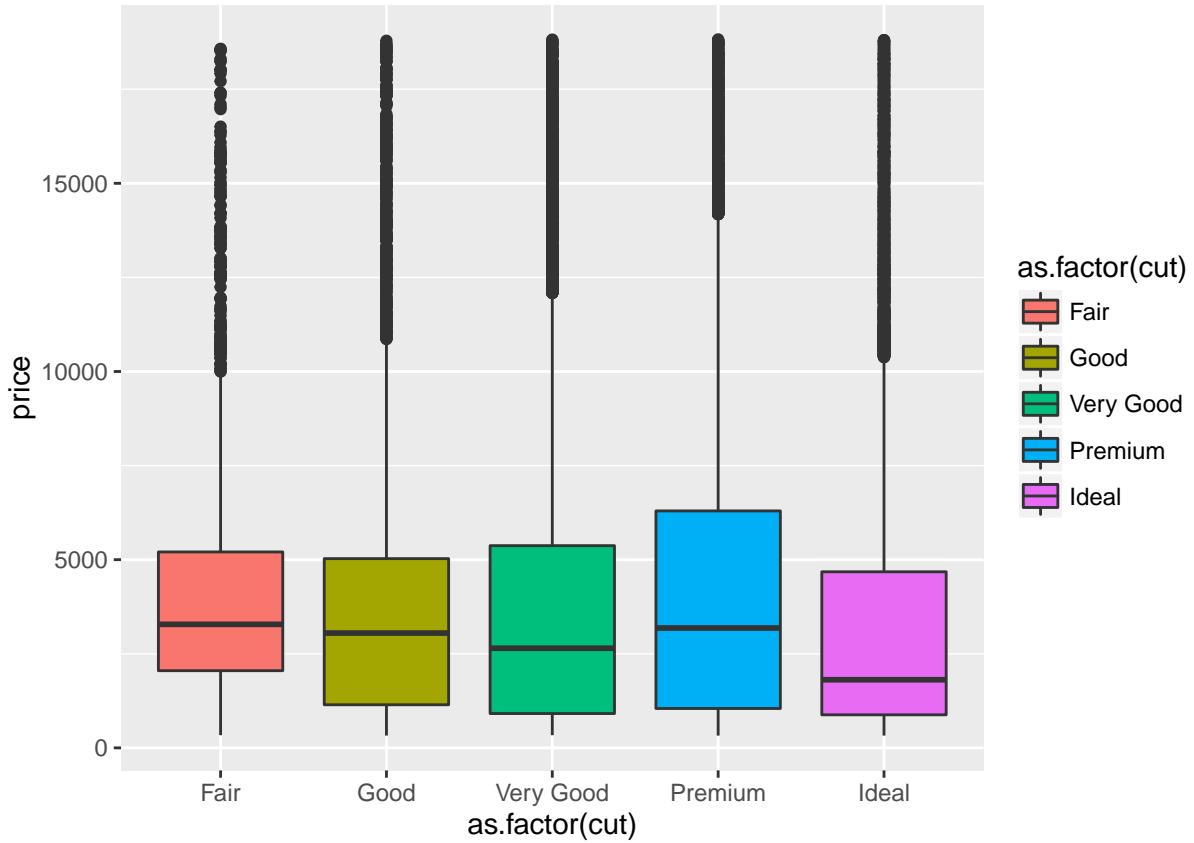
```
ggplot(data = diamonds, aes(x = price)) +  
  geom_histogram(fill=I("blue"), binwidth = 500, col=I("red")) +  
  xlab("Preço") +  
  ylab("Frequência") +  
  geom_density(data=diamonds, aes(x=price, group=carat), alpha=0.5, adjust=2) +  
  xlab("Preço") +  
  ylab("Frequência")
```



geom_boxplot

7. Quais são os aspectos estéticos (aesthetics) exigidos (obrigatórios) da função `geom_boxplot()`?
8. Faça boxplots da variável preço coloridos de acordo com a variável corte (cut).

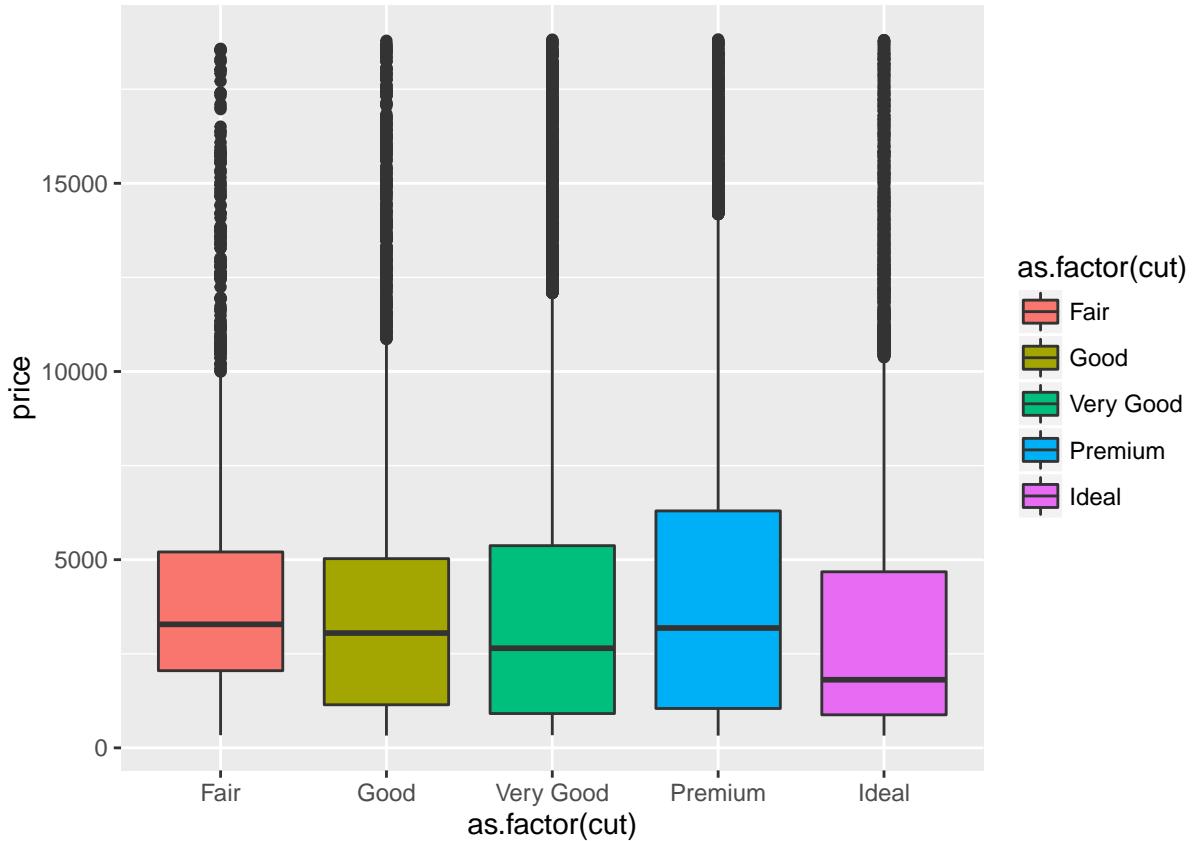
```
ggplot(data = diamonds, aes(x = as.factor(cut), y = price, fill = as.factor(cut))) + geom_boxplot()
```



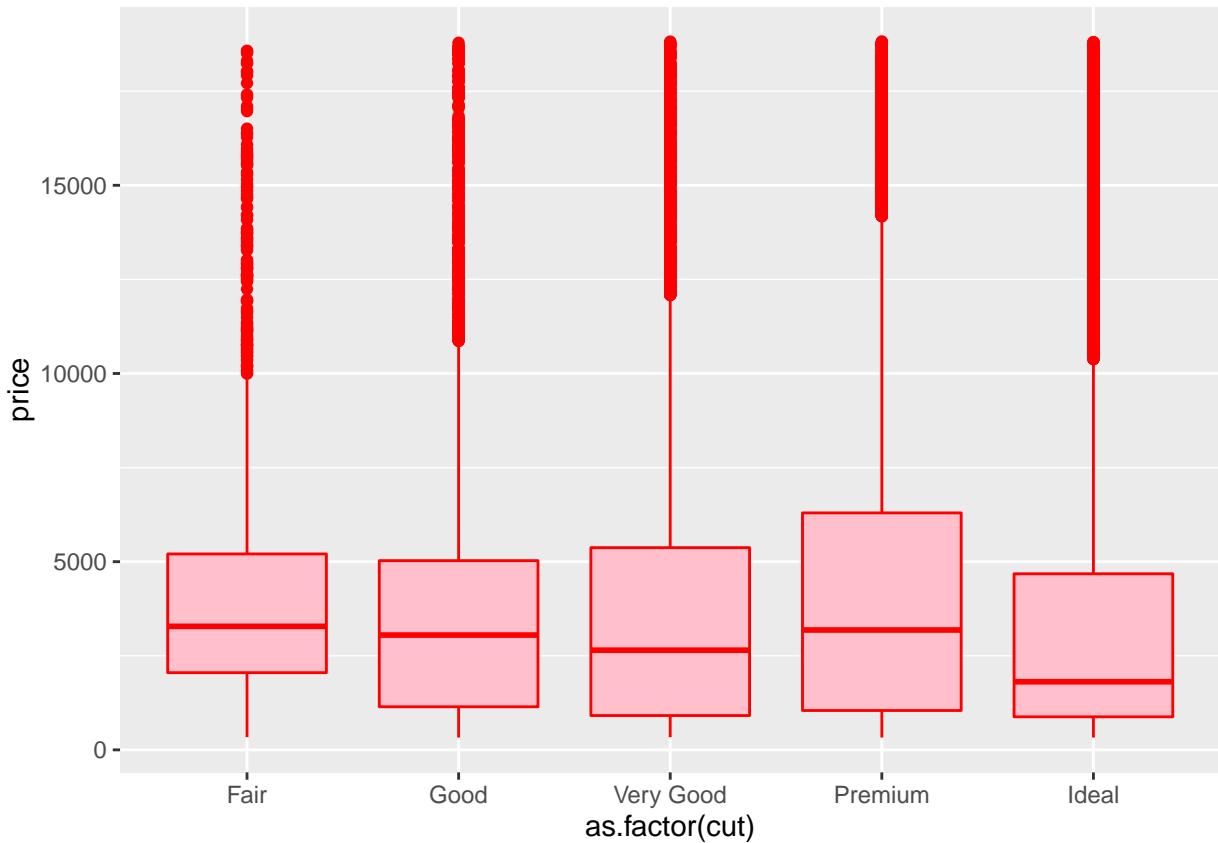
9. Substitua as cores padrões e remova a legenda do gráfico anterior.

(Dica: use a função `scale_color_manual`)

```
ggplot(data = diamonds, aes(x = as.factor(cut), y = price, fill = as.factor(cut))) + geom_boxplot()
```



```
ggplot(data = diamonds, aes(x = as.factor(cut), y = price)) +  
  geom_boxplot(color = "red", fill = "pink")
```



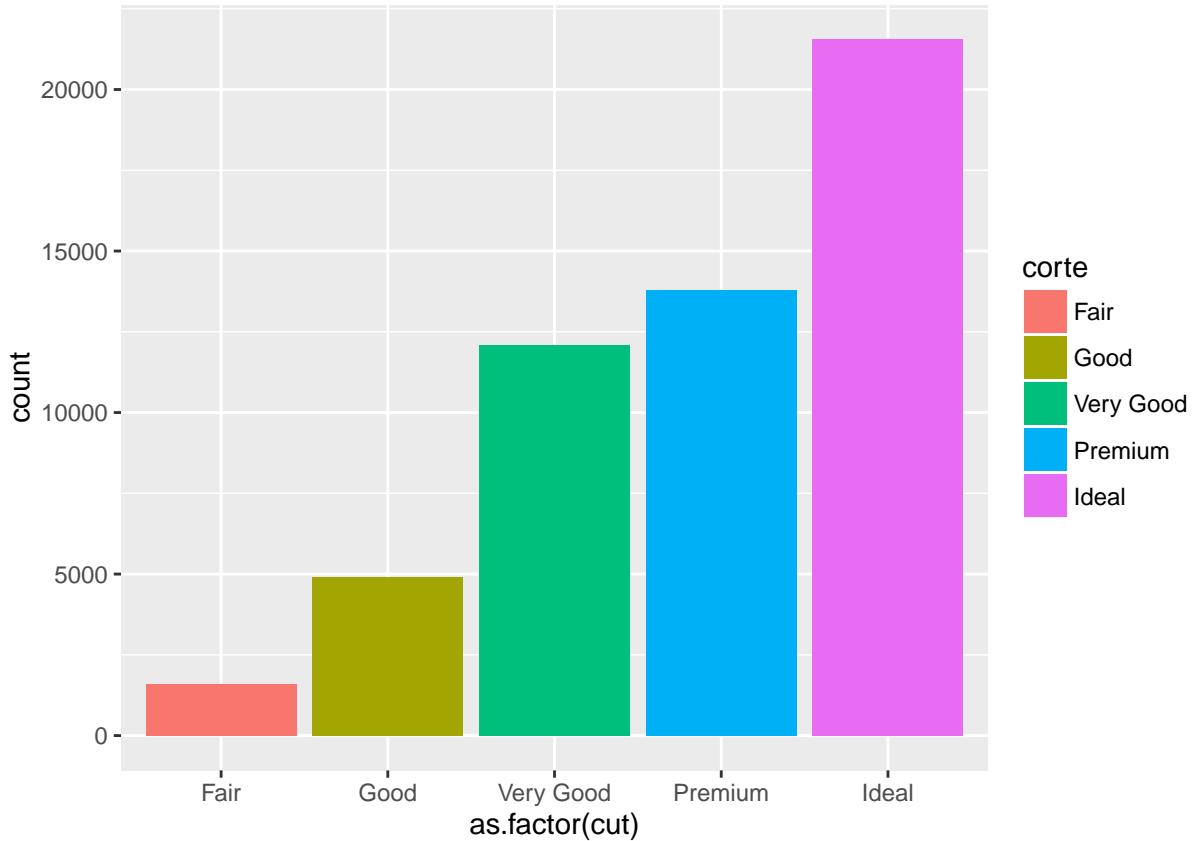
```
?geom_bar
```

`geom_bar`

10. Quais são os aspectos estéticos (aesthetics) exigidos (obrigatórios) da função `geom_bar()`?

11. Faça um gráfico de barras do número de diamantes em cada categoria da variável cor (color).

```
ggplot(data = diamonds, aes(x = as.factor(cut), fill = as.factor(cut))) +
  geom_bar() +
  labs(fill = "corte")
```

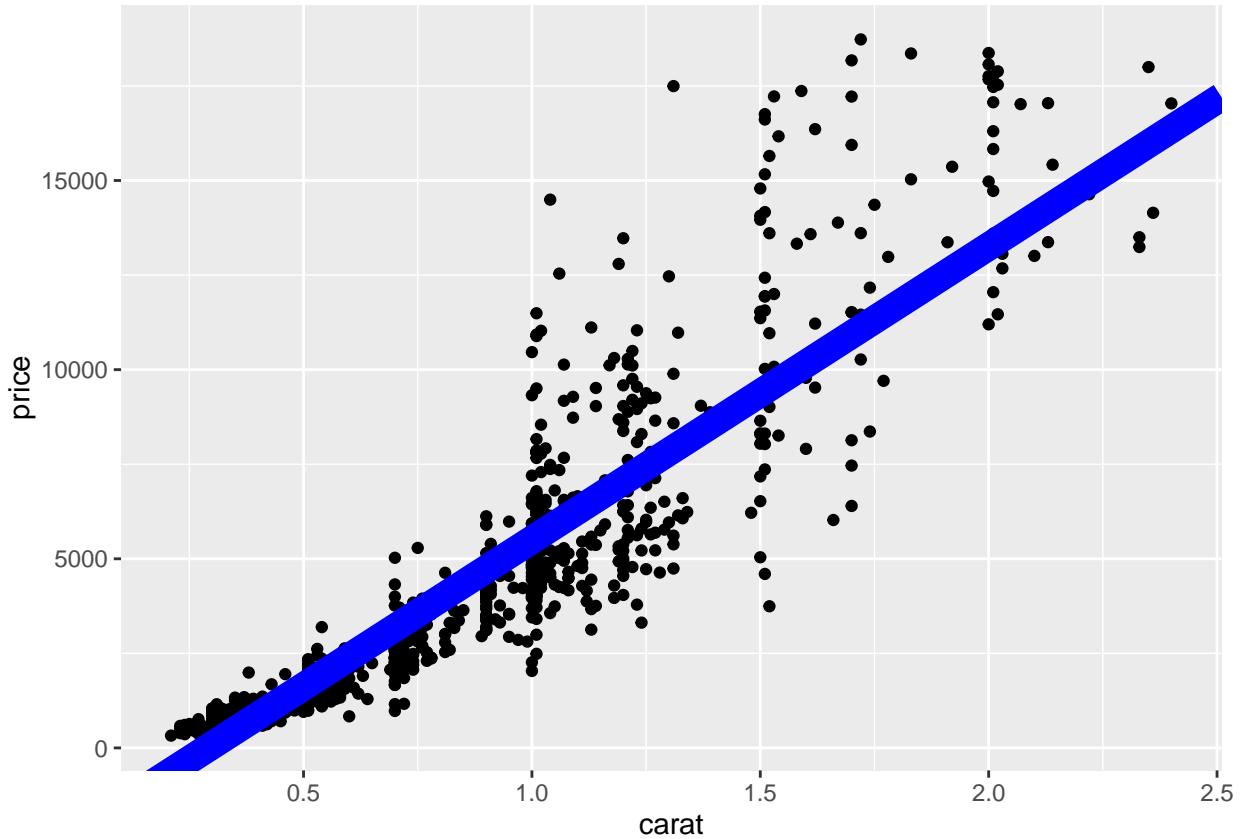


```
## um modelo de regressão
```

```
coef(lm(price ~ carat , data = diamonds))
```

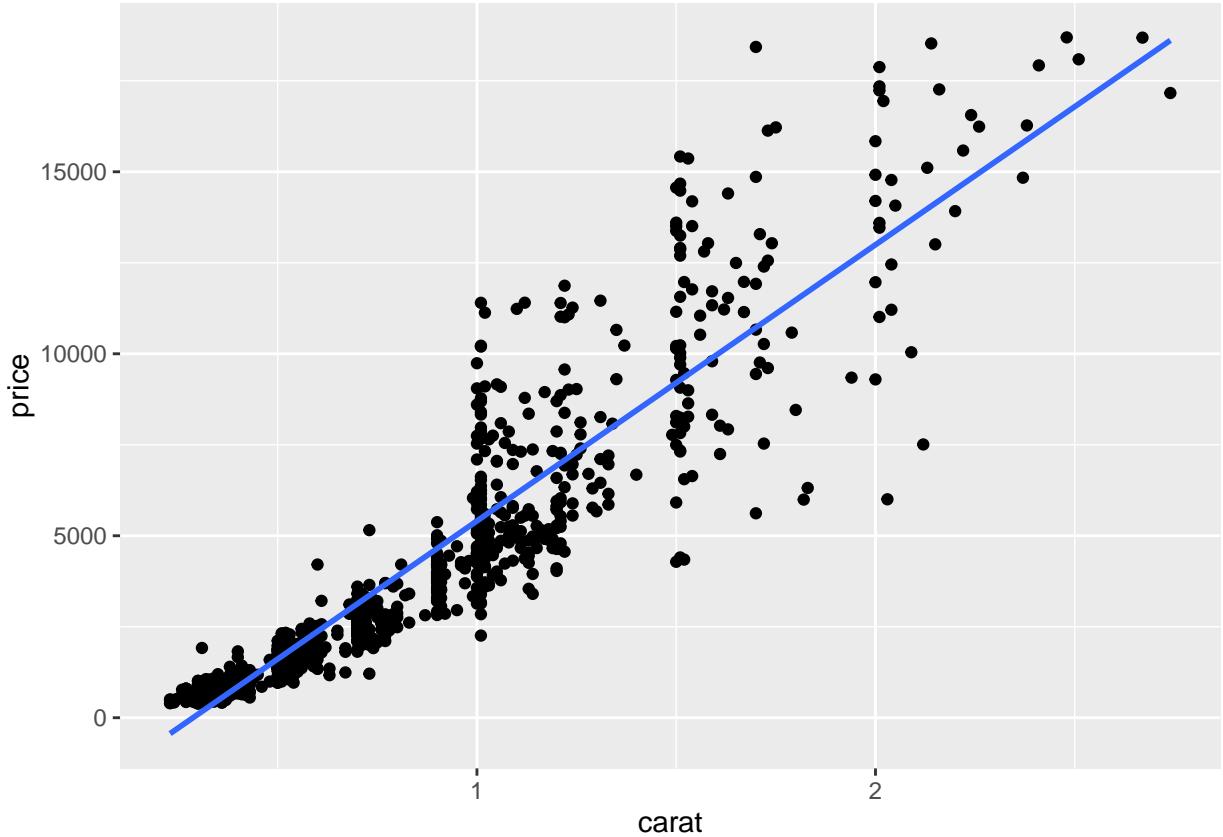
```
## (Intercept)      carat
## -2256.361     7756.426
```

```
ggplot(data = diamonds %>% sample_n(1000) , aes(x = carat, y = price)) +
  geom_point() +
  geom_abline(intercept = -2256.361, slope = 7756.426, color = "blue", size = 5)
```



suavizer as variaveis por um modelo de regressão

```
ggplot(data = diamonds %>% sample_n(1000), aes(x = carat, y = price)) +  
  geom_point() +  
  geom_smooth(method = "lm" , se= F)
```

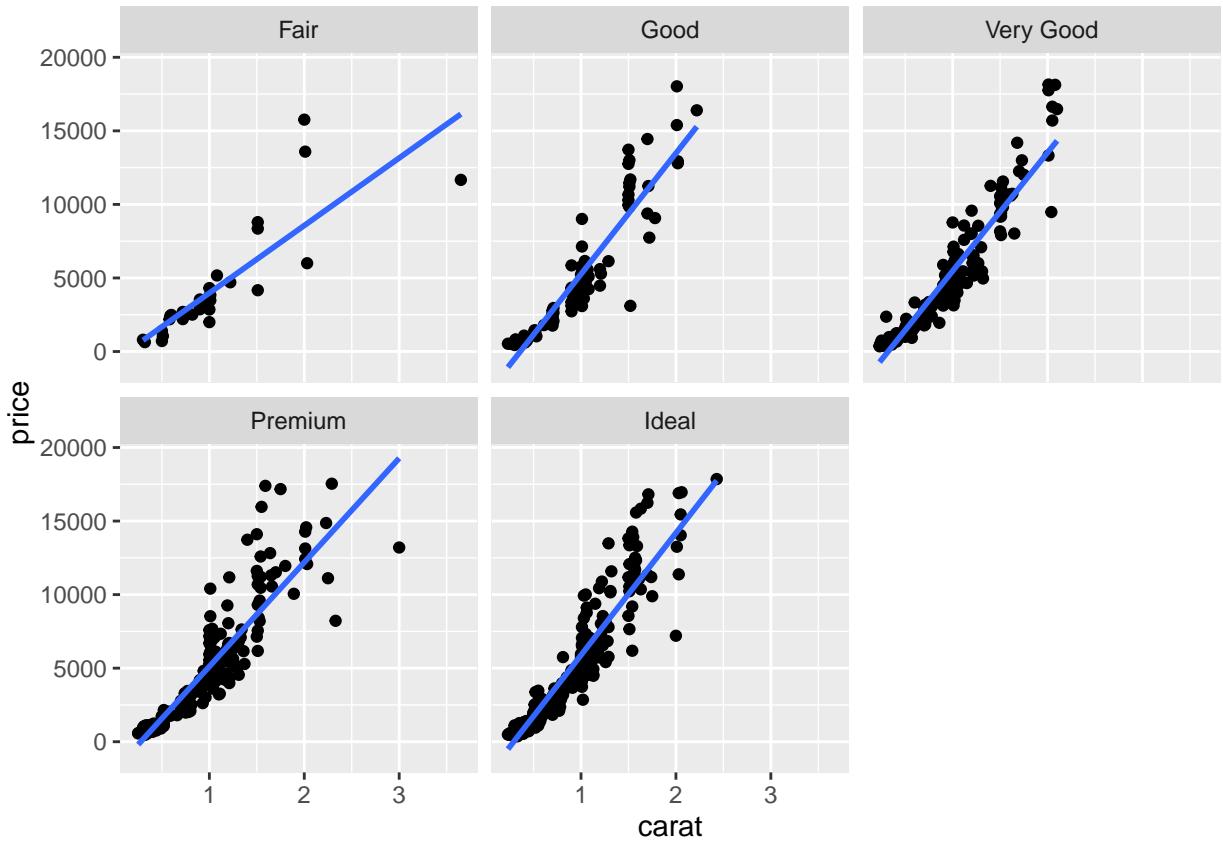


```
geom_abline(intercept = -2256.361, slope = 7756.426, color = "blue", size = 5)
```

```
## mapping: intercept = intercept, slope = slope
## geom_abline: na.rm = FALSE
## stat_identity: na.rm = FALSE
## position_identity
```

suavizer as variaveis por um modelo de regressão

```
ggplot(data = diamonds %>% sample_n(1000), aes(x = carat, y = price)) +
  geom_point() +
  geom_smooth(se= F, method = "lm") + facet_wrap(~cut)
```



```
geom_abline(intercept = -2256.361, slope = 7756.426, color = "red", size = 5)
```

```
## mapping: intercept = intercept, slope = slope
## geom_abline: na.rm = FALSE
## stat_identity: na.rm = FALSE
## position_identity
```