Curso GGPLOT2 - Estatística UFMG

Virgilio

29/05/2019

Curso GGPLOT2 - Estatística UFMG - 29 de maio de 2019

O curso visa explanar e demonstrar as aplicações mais básicas do pacote GGPLO2. As aplicações do pacote permitem apresentações mais dinâmicas e elaboradas, com grande versatilidade e apresentação de análises de bancos de dados diversos. Todos os parâmetros executados nas funções do GGPLOT são estruturados em "layers" (camadas) seguindo uma lógica gramatical específica, cada qual responsável por uma funcionalidade específica, como mostra a disposição a seguir:

Data → banco de dados (representa o data framae onde estão as variáveis de interesse); Aesthetics → mapeamento dos dados no gráfico - eixos (x,y), cor, preenchimento, tamanho, opacidade, etc; Geometries → elemento visual que será usado (pontos, linhas, histograma, barra ,boxplot, etc); Facets → sub-divisões a serem plotadas (colunas e linhas); Estatístics→ visualização de resumos - curvas, resumos (média, mediana, etc.); Coordenadas → o espaço no gráfico em que as variáveis serão exibidas;

Formas gramaticais de representação dos layers na formação de um grafo usando o GGPLOT2: - Data e aesthetics: ggplot (data, aes(x,y, ...)) ; - Geometries: geom_nome_geometria ; - Facets: facet_grid(...) ; - Stats: stats_qq, etc. ; - Coordinates: coord_equal(), coord_cartesian(), etc. ;

OBS: No ggplot2, adicionamos as camadas com sinal de +

Executando a "library" do GGPLOT2

A princípio, iremos trabalhar com o banco Íris, e , para isso, inicialmente executamos a "library" do pacote GGPLOT.

```
# Executando a Library
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.3
```

```
# importando o banco de dados IRIS e nomeando ele de BD
bd = iris
```

Depois de se executar o pacote de interesse, é interessante estabelecer alguns processos analíticos, estes devem ser realizados quando se iniciam trabalhos com um banco de dados específicos de modo que proporcione a compreensão do que o banco de dados representa, quais perguntas ele responde, que tipos de problemas ele envolve e que tipos de variáveis o compõe. Para um melhor entendimento do mesmo é interessante tirar medidas ou métricas básicas para se ter uma noção dos limites explicativos e metodológicos do banco estudado, assim, passaremos brevemente para algumas funções dessa finalidade:

Função Summary

A função summary (resumo) é uma função genérica usada para produzir resumos de resultados dos resultados de várias funções de ajuste de modelo, sintetizando assim algumas informações essenciais básicas das variáveis do banco, como valores mínimos e máximos, média e mediana e os quantis.

```
summary(iris)
```

```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
##        Species
##  setosa    :50
##  versicolor:50
##  virginica :50
##
##
##
```

Nomes das variáveis do banco

A segunda função que apresentamos "puxa" e apresenta os nomes das variáveis que estão contidas no banco usado.

```
# Função que puxa todos os nomes das variáveis do banco
names(iris)
```

```
## [1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"  "Species"
```

Estrutura das variáveis do banco

Esta função exibe compactamente a estrutura interna de um objeto R, ou seja, é uma função de diagnóstico e uma alternativa ao summary (como visto acima). Idealmente, apenas uma linha para cada estrutura "básica" é exibida. Ele é especialmente adequado para exibir de forma compacta o conteúdo de listas.

```
# Função que puxa as estruturas de todas as variáveis presente no banco
str(iris)
```

```
## 'data.frame':    150 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Depois de uma breve demonstração de como compreender melhor a estrutura e as variáveis do banco de dados que se irá trabalhar passamos para a construção dos grafos:

Gráfico de Dispersão

No exemplo a seguir faremos um gráfico de Dispersão do banco de dados Íris (inato do próprio R que estamos utilizando). Nessa representação gráfica consideramos o eixo X como Sepal.Length (comprimento da Sépala) e o eixo Y como Sepal.Width (largura da Sépala), sendo cada espécie de planta diferenciada por uma cor distinta.

```
#Plotando um grafico de dispersão
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width)) +
  geom_point(size = 2, shape = 1)
```

Note-se que a estutura do comando inicia-se com "ggplot", em seguida vem "iris" representando o nome do banco de dados que contém as variáveis a serem trabalhadas, e posteriormente dois layers usados na construção do gráfico de dispersão o aesthetics (aes) e o geometries (geom_point). O aesthetics (aes) na função está determinando as variáveis dos eixos X e Y, enquanto o geometries (geom_point) determina o tipo de gráfico será plotado com as informações disponíveis, no caso, gráfico de dispersão.

Gráfico de Dispersão distinguido por Espécie de planta

O gráfico mostrado a seguir é a mesma representação do anterior, porém com uma leve distição, nele para além da distribuição do comprimento e largura da Sépala, tem-se a a separação por espécie de plantas. Esta especificação é representada no código - note-se que este atributo está representado dentro do aesthetics (aes), responsável pela distinção de cor, eixos (X e Y), tamanho, opacidade, etc - como "col". Desse modo, "col" está atribuindo para cada categoria da variavel Species uma cor específica, no caso, vermelho para as "setosas", verde para as "versicolor" e azul para as "virginica".

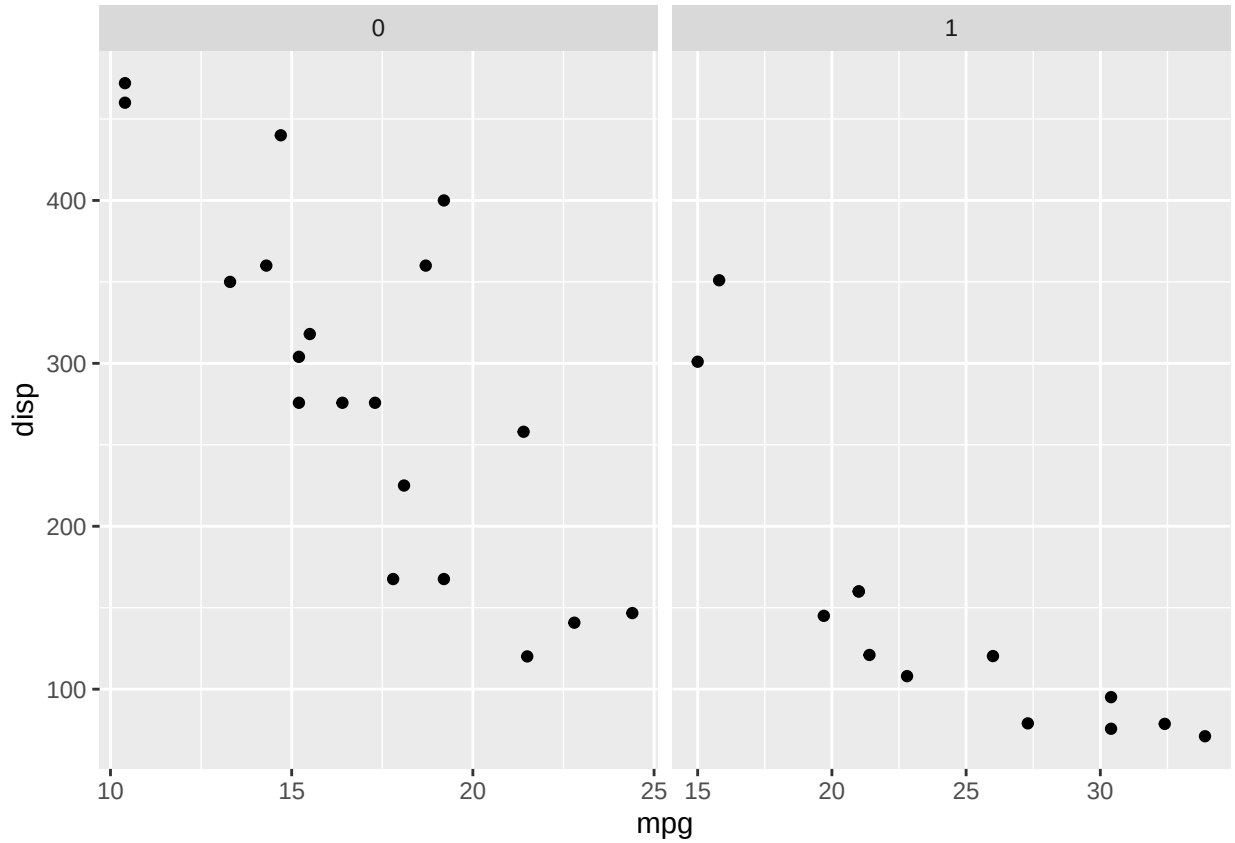
```r
#Plotando um grafico de dispersão
ggplot(iris, aes( x = Sepal.Length, y = Sepal.Width, col = Species)) +
  geom_point(size = 2, shape = 1)
```

Gráfico de Dispersão com Facets

O gráfico mostrado a seguir é a mesma representação da estrutura anterior, usando desta vez o banco *mtcars*. Nele utiliza-se a geometries *geom_point*, e para facilitar a visualização de diferentes subconjuntos dos dados em gráficos separados, permitindo a visualização de comportamentos diferentes dependendo do grupo (para mais informações acessar: http://material.curso-r.com/ggplot/#facets). Na estrutura apresentada temos dois tipos de *facets* representados, a separação na vertical das categorias e outra na horizontal, ambas seguindo a representação **facet_grid(variavel_facetada)**. Usa-se: # o "~." depois do AM separa duas categorias discretas na horizontal # o ".~" antes do AM separa duas categorias discretas na vertical

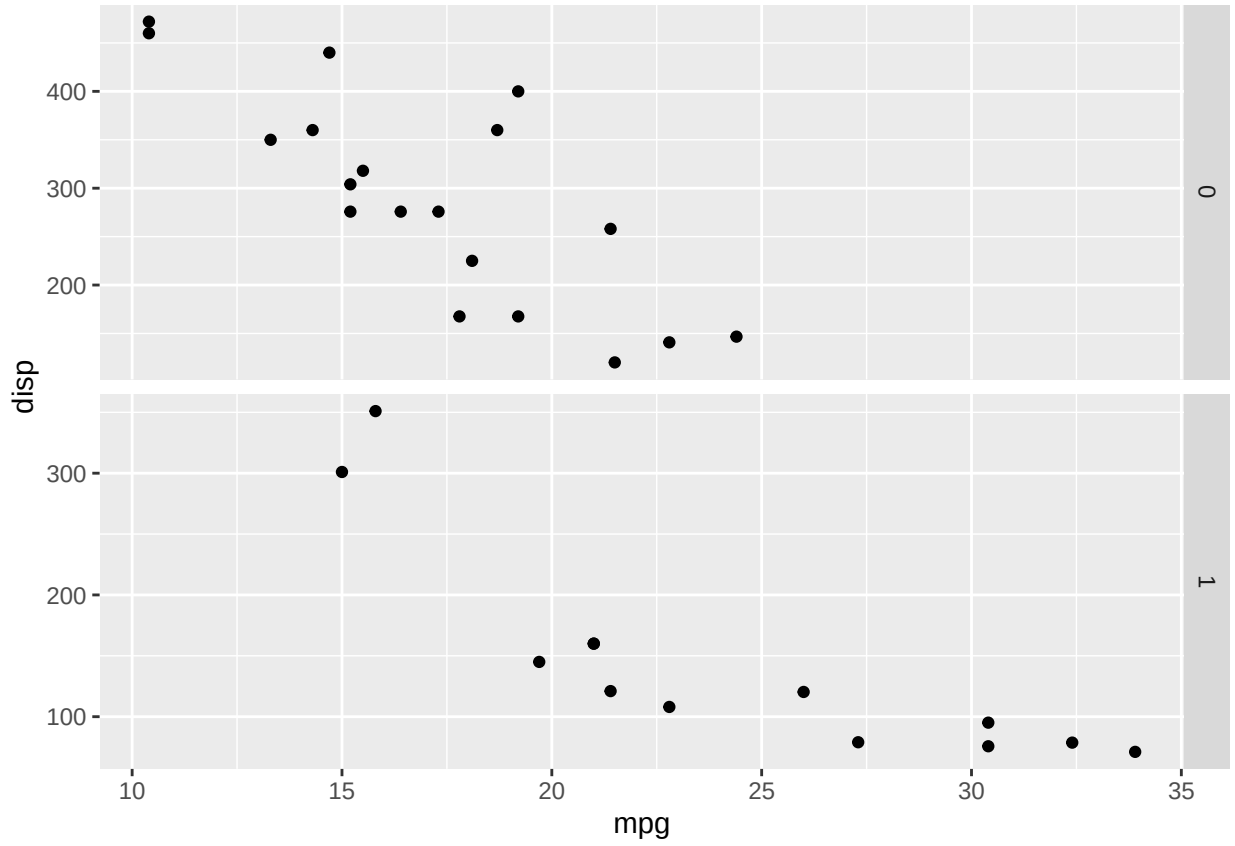
Representação na vertical:

```r
#Facets- tirar a escala do X e Y
ggplot(mtcars, aes(x = mpg, y = disp)) + geom_point() +
  facet_grid(.~am, scales = "free")
```



Representação na horizontal:

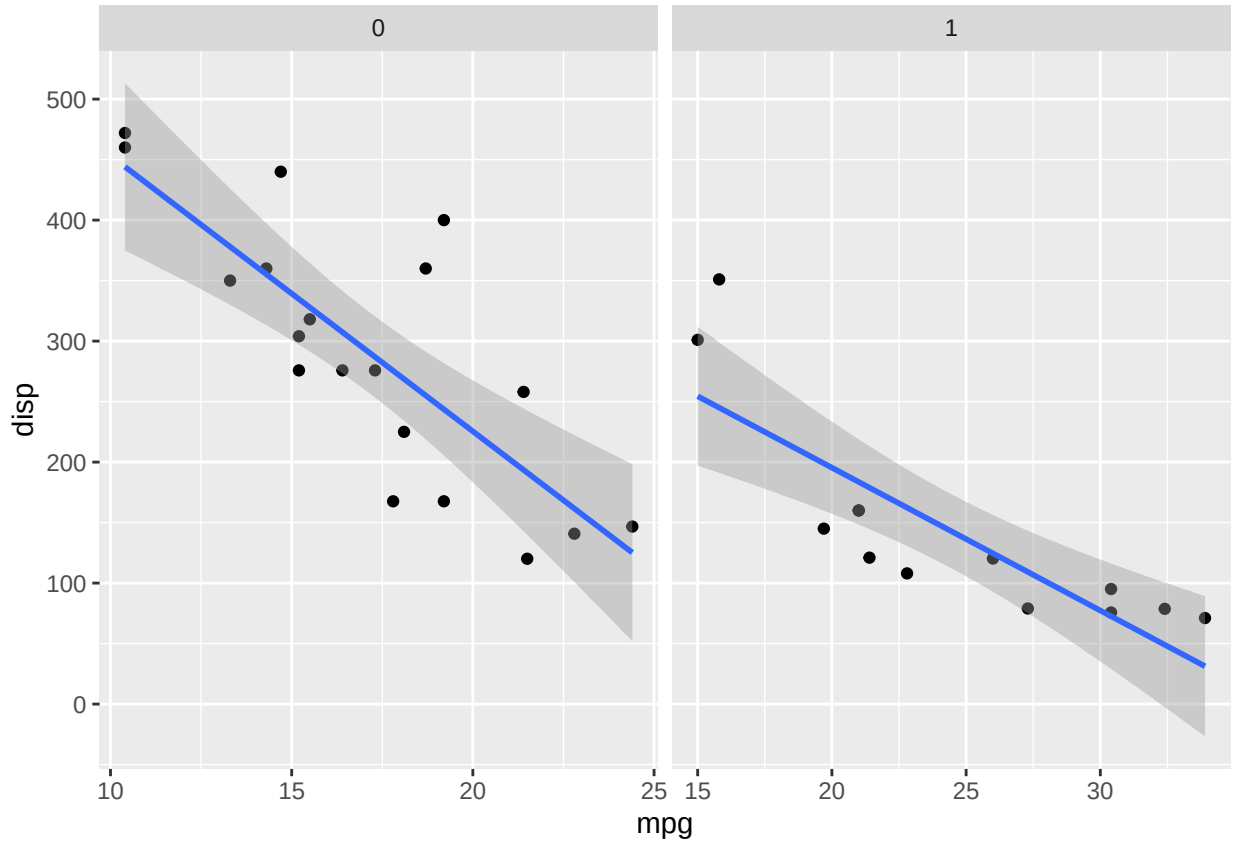
```
#Facets- tirar a escala do X e Y
ggplot(mtcars, aes(x = mpg, y = disp)) + geom_point() +
  facet_grid(am~., scales = "free")
```



Representação na vertical com curva de regressão:

```
#Facets- tirar a escala do X e Y
ggplot(mtcars, aes(x = mpg, y = disp)) + geom_point() +
  facet_grid(.~am, scales = "free") +
  geom_smooth(method = "lm")
```

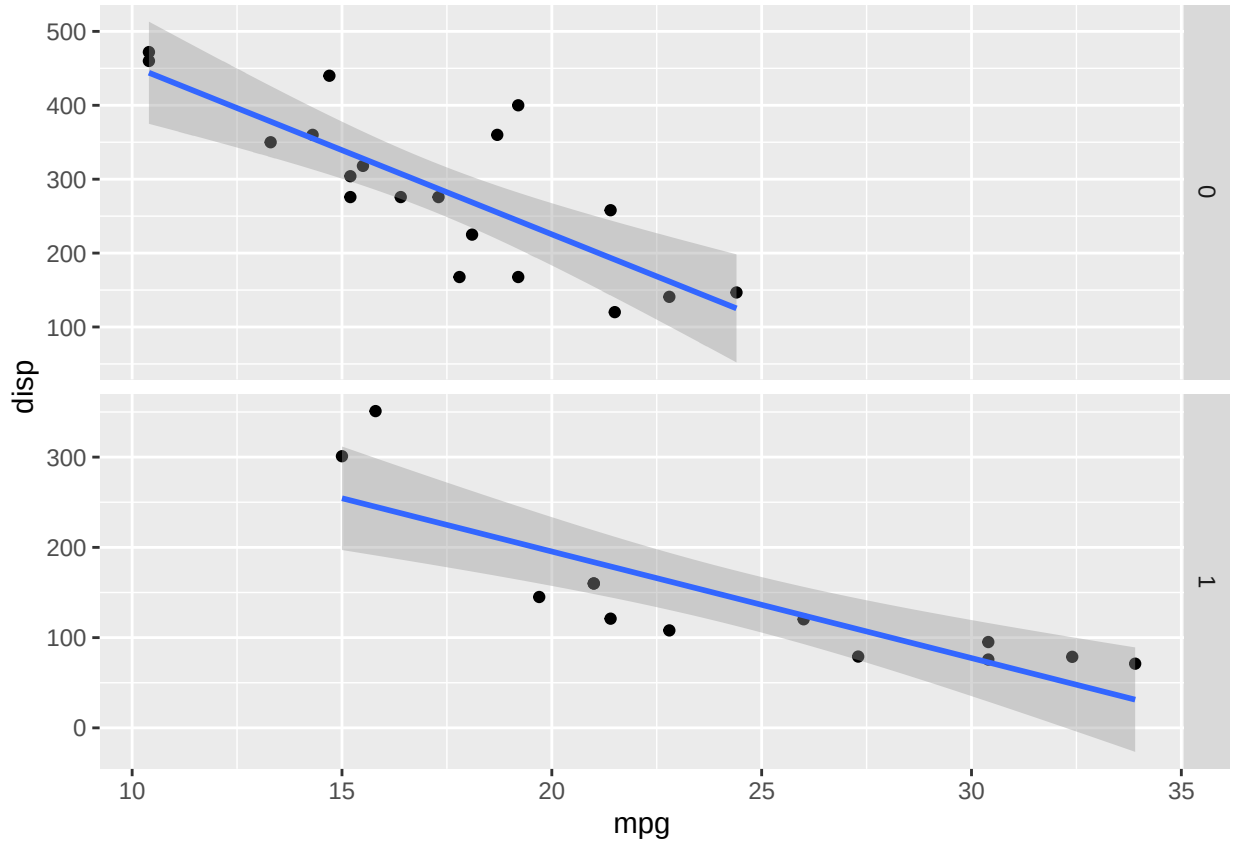
```
## `geom_smooth()` using formula 'y ~ x'
```



Representação na horizontal com curva de regressão:

```
#Facets- tirar a escala do X e Y
ggplot(mtcars, aes(x = mpg, y = disp)) + geom_point() +
  facet_grid(am~., scales = "free") +
  geom_smooth(method = "lm")
```

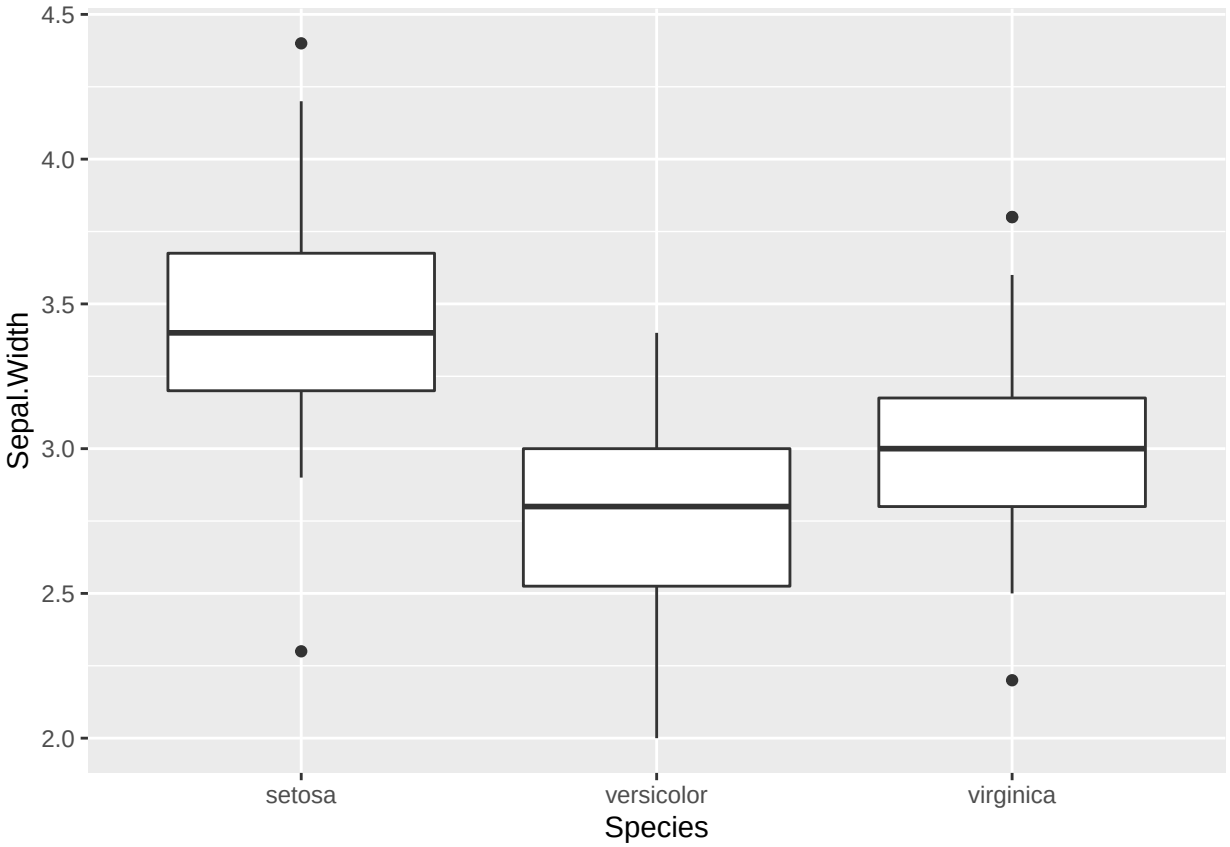
```
## `geom_smooth()` using formula 'y ~ x'
```



Boxplot

O gráfico demonstrado a seguir representa um Boxplot. A sua estrutura de códico utiliza o nome do banco de dados ("iris"), o aesthetics (aes) e o geometries (geom_boxplot). Sua estrutura é essencialmente a mesma do gráfico de dispersão, só se diferenciará pela geometria utilizada para a plotagem, na anterior era a "geom_point" e agora é a "geom_boxplot".

```
# bloxpot
ggplot(iris, aes(x = Species, y = Sepal.Width)) +
  geom_boxplot()
```



Histograma

O gráfico do tipo Histograma que veremos a seguir apresenta a mesma composição de código das anteriores, diferenciado-se pelo layer geometries ("geom_histogram"). Nele é representado no eixo X o comprimento da Sépala e no eixo Y a frequência simples observada, além disso, tem-se a variável comprimento da Sépala ("Sepal.Length") como um fator ("factor") representado na legenda. Assim, temos um grafo que identifica a frequência pelo comprimento da Sépala, onde cada cor refere-se a um comprimento distinto. Os gráficos do tipo Histograma são utilizados de forma a apresentar a densidade intervalar da distribuição da variável analisada (geralmente usado em variáveis numéricas).

```
# Plotando um histograma
ggplot(iris, aes(x = Sepal.Length,fill = factor(Sepal.Length))) +
  geom_histogram(bins = 10, col = "white")
```

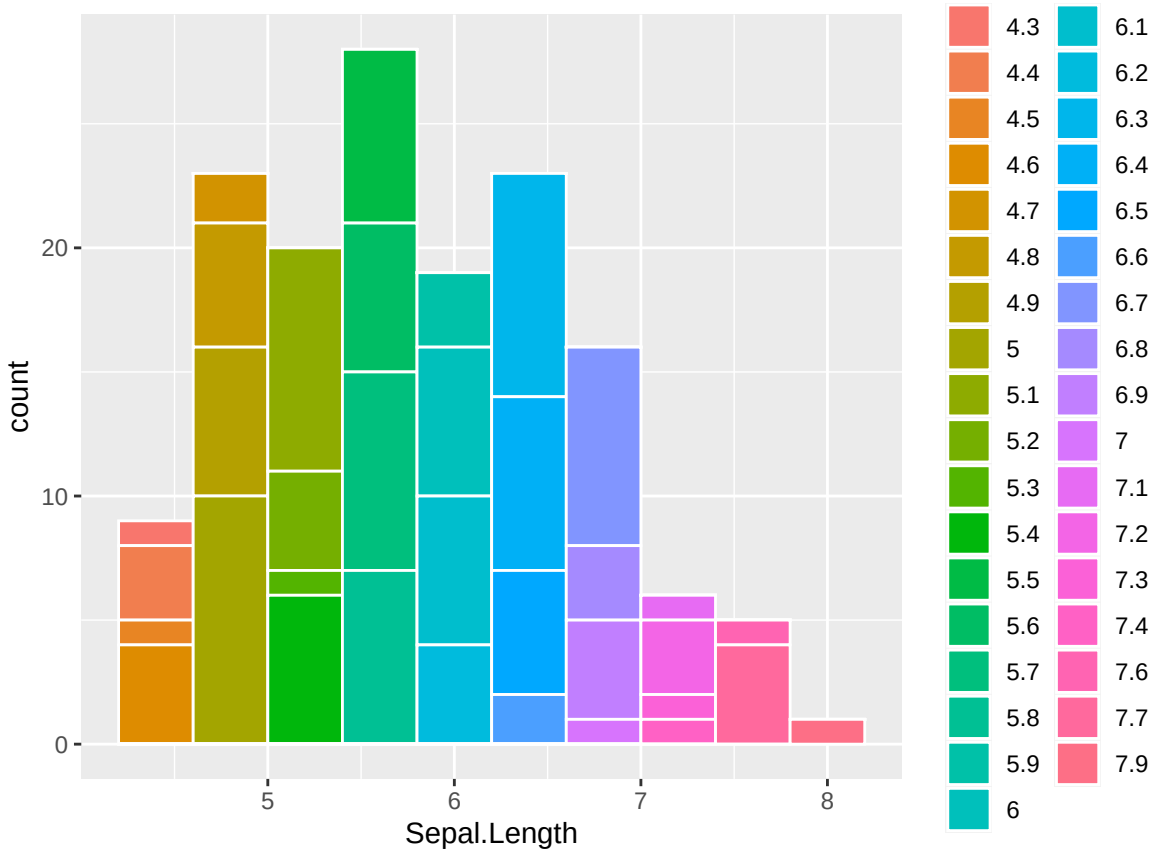


Gráfico de Barras

O gráfico de barra indicado a seguir apresenta a mesma composição de código das anteriores (layers), diferenciado-se pelo layer geometries ("geom_bar"). No layer geometries, no caso *geom_bar*, é delimitado os parâmetros tamanho (*size*), cor (*colour*) e opacidade (*alpha*) das bordas das barras plotadas. Os gráficos de Barra, ao contrário dos Histogramas, permitem uma apresentação visual qualificada para dados categóricos. O gráfico a seguir utiliza o banco de dados "mtcars" inato do R base, igual ao "íris".

```
ggplot(mtcars, aes( x = gear, fill = factor(carb))) +
  geom_bar(size = 1.5, colour = 'white', alpha = 0.6)
```

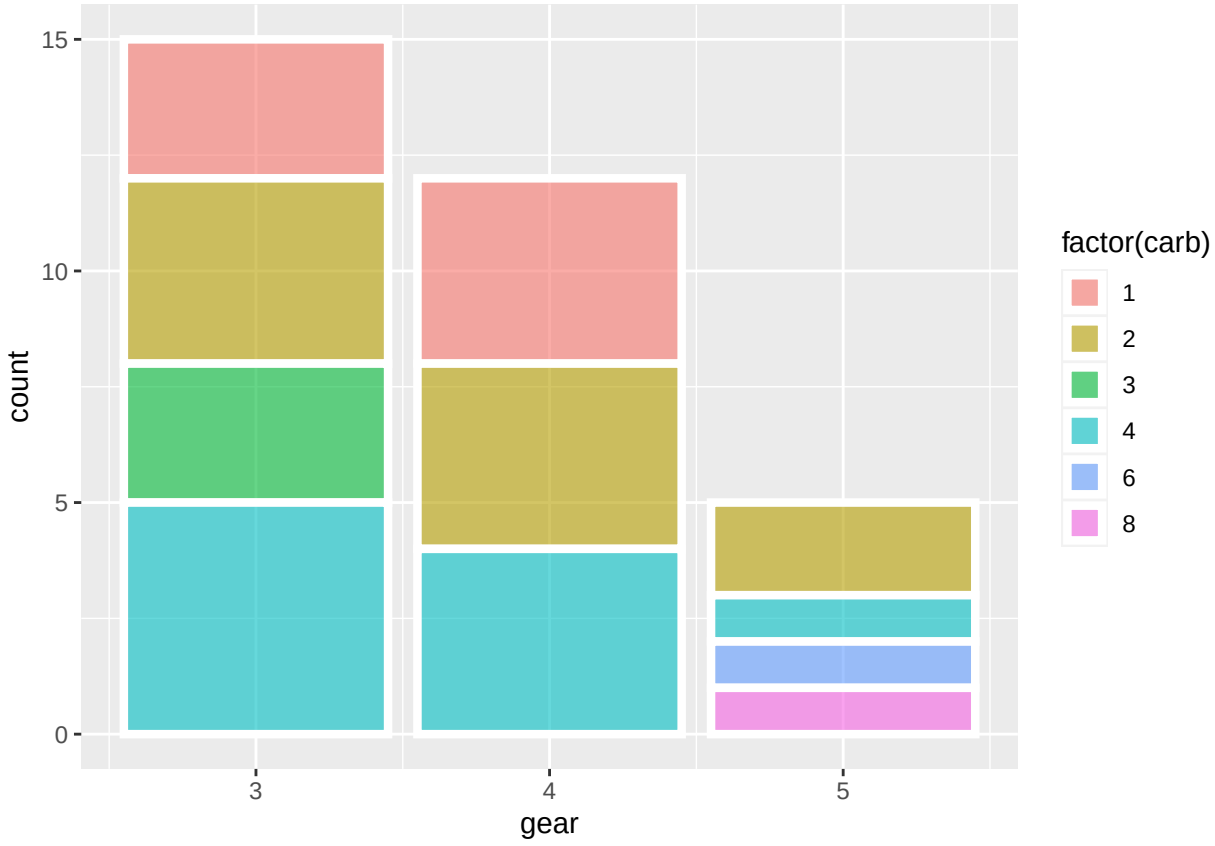


Gráfico de Densidade

O gráfico de Densidade plota a densidade e a distribuição dos casos observados no banco por uma variável específica, no caso, uma variável contínua. Nele temos a distribuição da densidade de ocorrências pelo comprimento da Sépalas, distinguidos entre três espécies de plantas: as "setosas", as "versicolor" e as "virginicas".

A estrutura do código é composta pelos layers: dados ("iris"), o aesthetics (aes) e o geometries ("geom_density"). Seguindo a mesma linha de raciocínio dos grafos anteriores, vemos a construção do gráfico a partir do código:

```
#Grafico de Densidade
ggplot(iris, aes(x = Sepal.Length, fill = Species)) +
  geom_density(alpha = 0.5)
```

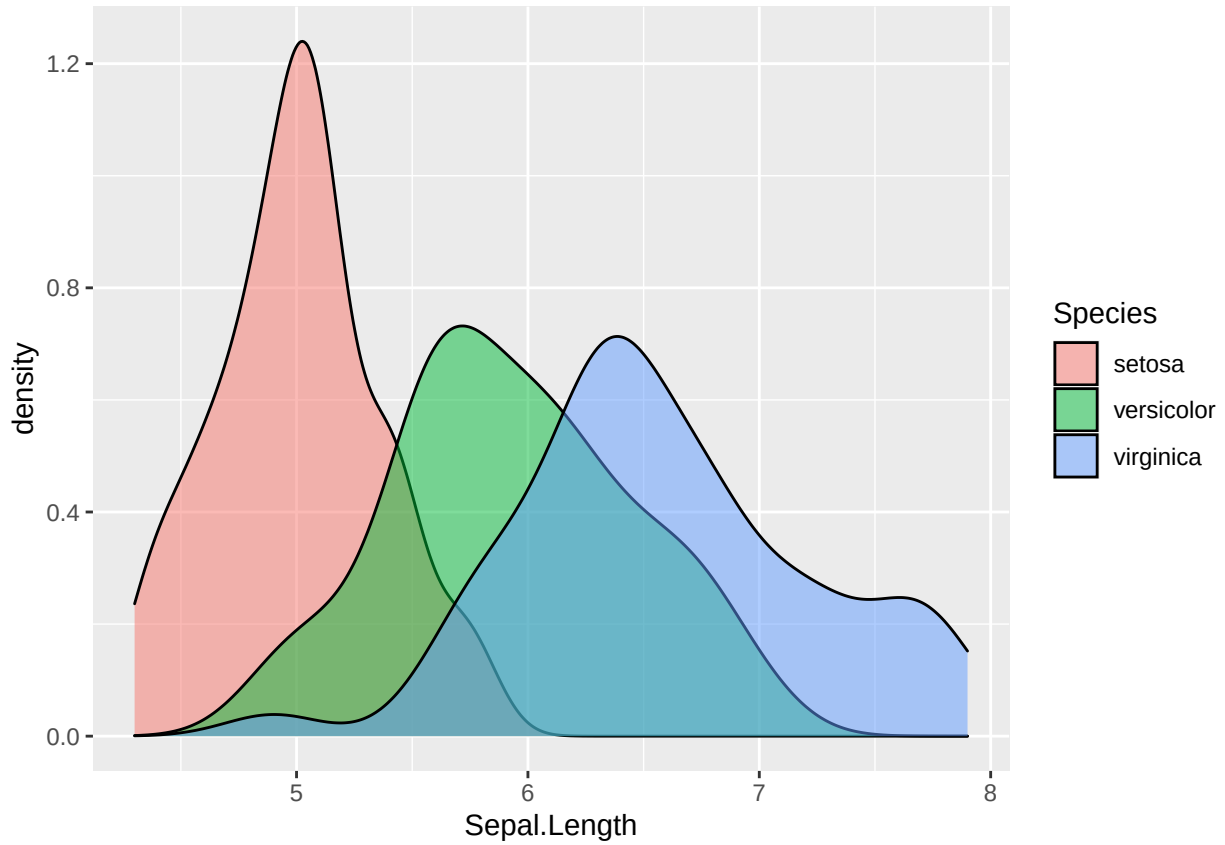
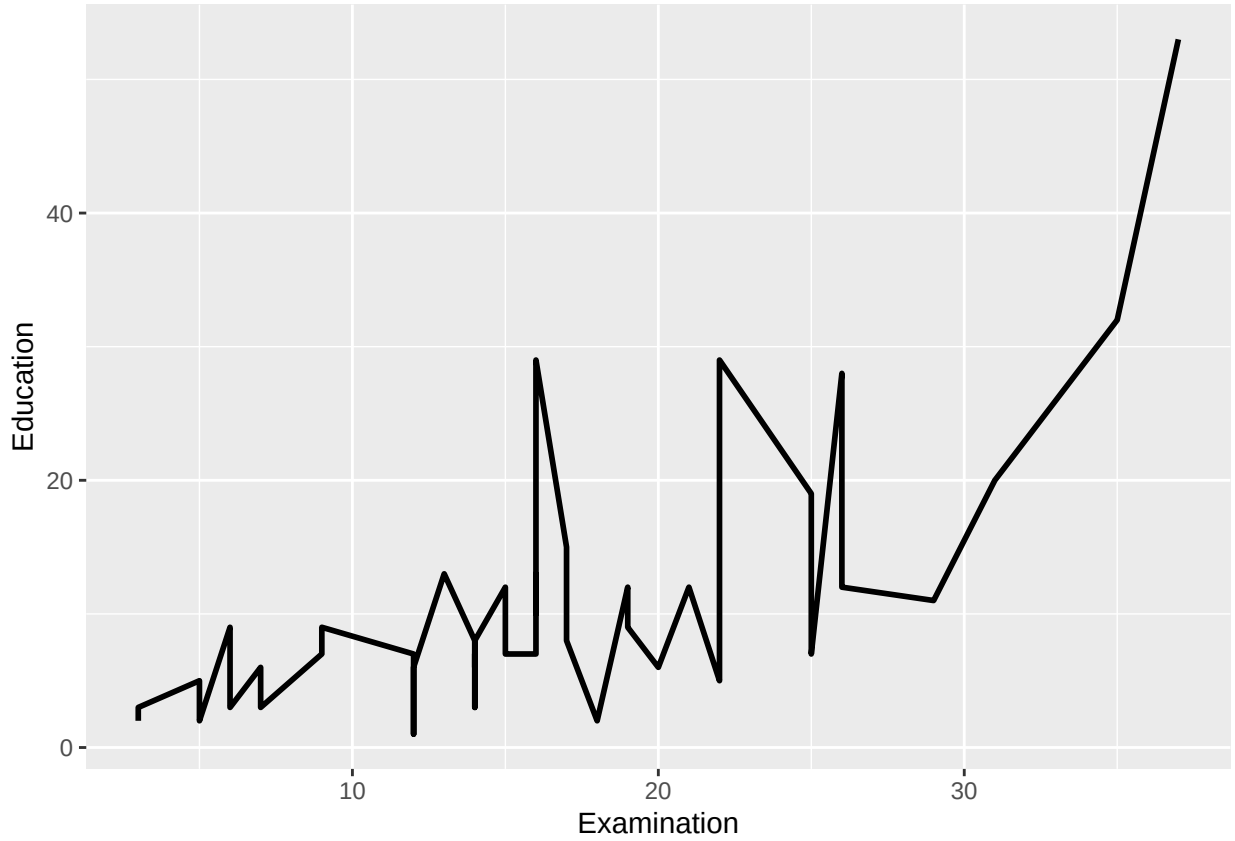


Gráfico de Linha

Para este gráfico, utilizamos outro banco já inato do R chamado “siwss”. Nele com estrutura semelhante às anteriores vemos: dados (“swiss”), o aesthetics (aes) e o geometries (“geom_line”).

```
#Grafico de Linha
ggplot(swiss, aes(x = Examination, y = Education)) +
  geom_line(size = 1)
```



Histograma com labels (Títulos)

Para este gráfico, novamente um histograma, adicionamos um layer onde se distingue os títulos do gráfico, seu subtítulo e o título de seus eixos X e Y. No código há tambem, dentro do aesthetics (aes), o comprimento entre as variáveis plotadas (escala) sendo representado por "bins" e possuindo o intervalo de 10 unidades no eixo da Frequência e a cor das bordas colunas (com a cor preta - "black") representada por "col". No layer onde se codifica os títulos, "labs", tem-se a inclusão dos títulos entre aspas ("") por representarem informações textuais.

```
#Histograma com Nomes (titulos)
ggplot(iris, aes(x = Sepal.Length)) + geom_histogram(bins = 10, col = "black") +
  labs(title = "Histograma de Comprimento da Sépala",
       subtitle = "Gráfico", x = "Comprimento da Sépala",
       y = "Frequência")
```

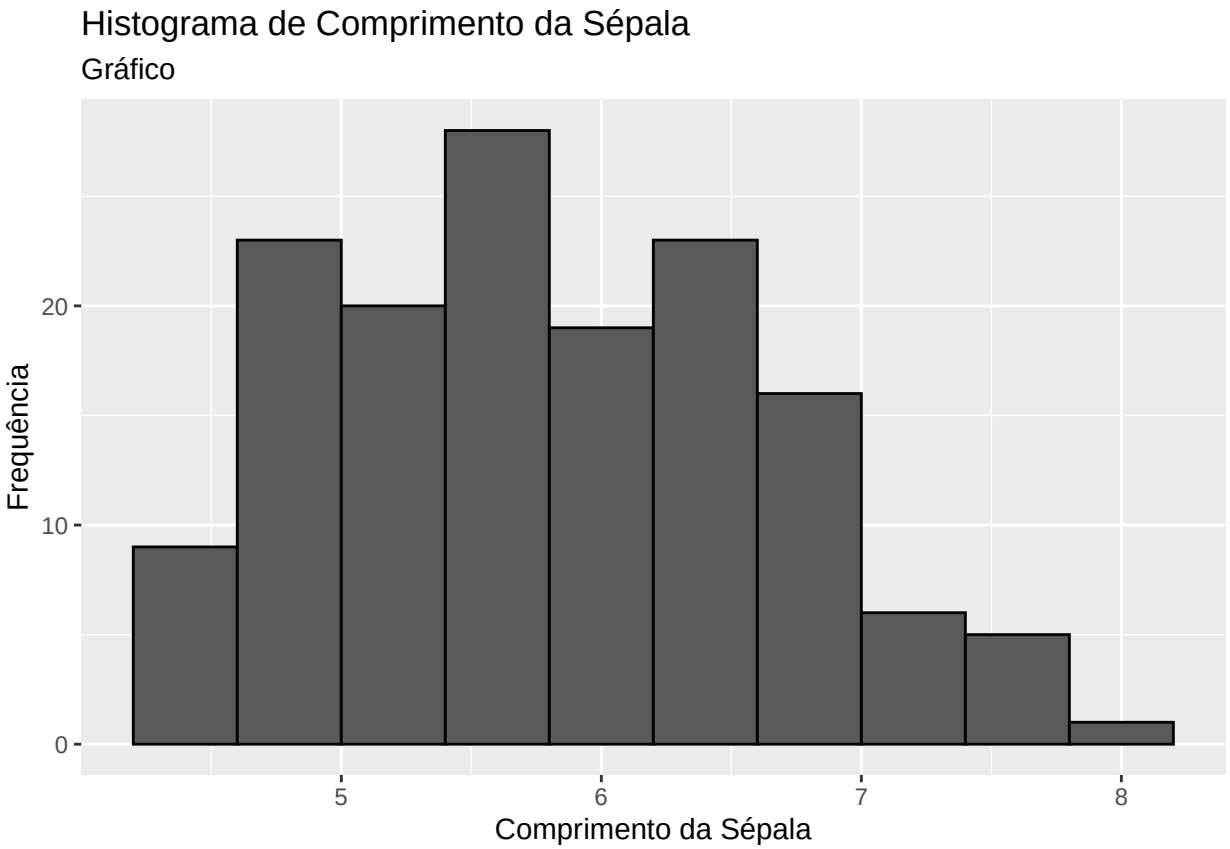


Gráfico Quantil-Quantil

Para um gráfico Quantil-Quantil tem-se a linha de estatísticas ("stat_qq") que computa a inclinação e intercepta a linha que conecta os pontos em quartis especificados das distribuições teóricas e amostrais.

```
# Plotando com "Stats"
ggplot(iris, aes(sample = Sepal.Width)) +
  stat_qq()
```

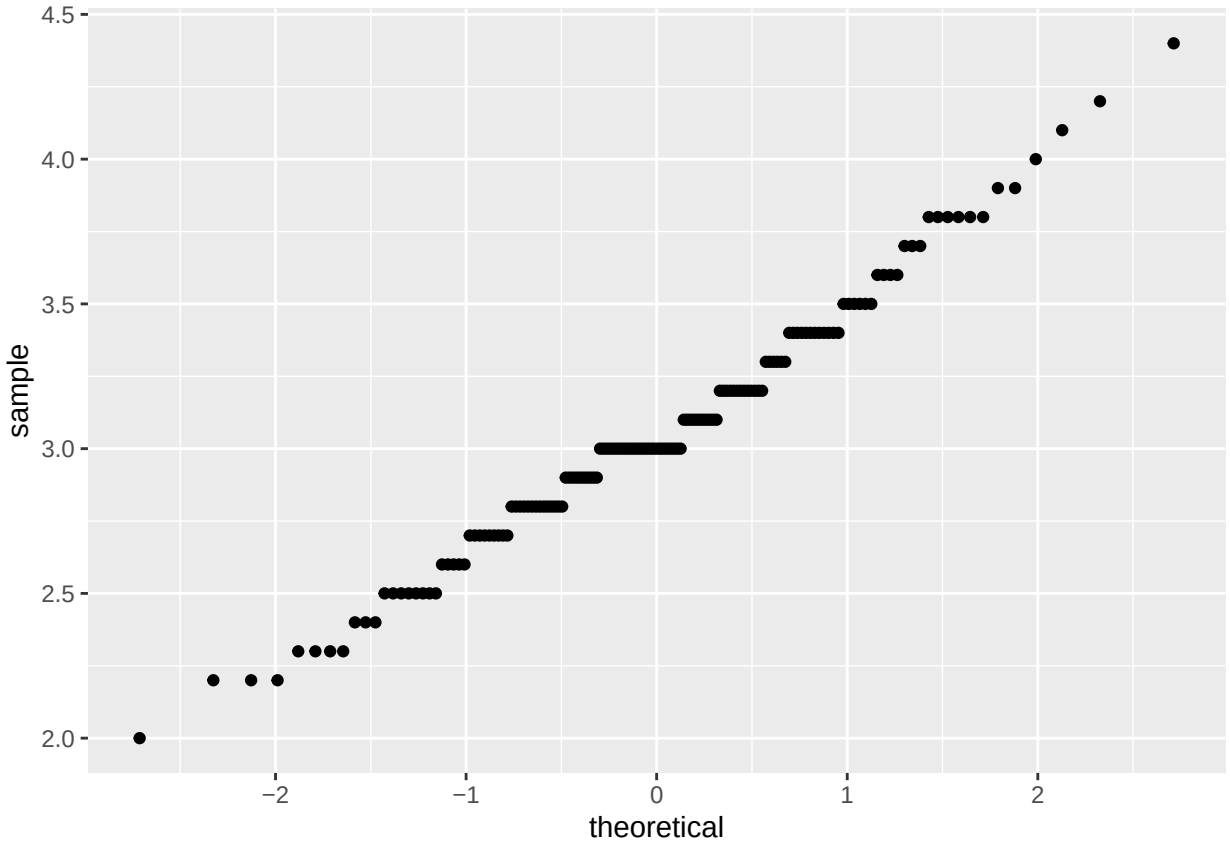


Gráfico Quantil-Quantil (com escala)

Sendo o mesmo tipo de grafo do mostrado anteriormente, este se destaca por apresentar uma escala determinada pelo código. Desse modo, tem-se a linha de estatísticas ("stat_qq") que computa a inclinação e intercepta a linha que conecta os pontos em quartis especificados das distribuições teóricas e amostrais, sendo a mesma da anterior. Sua representação e construção se difere no entanto pelos layers de títulos ("labs") e pela delimitação da escala (contínua) dos eixos X e Y: scale_x_continuous() e scale_y_continuous () .

```
#Grafico de dispersão - Stats com escala
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width)) +
  geom_point() +
  scale_x_continuous(limits = c(4, 9), breaks = c(4, 5, 6)) +
  scale_y_continuous(limits = c(0, 9), breaks = c(4, 5, 6)) +
  labs(title = "Grafico de Dispersão de Comprimento da Sépala por Largura da Sépala",
       subtitle = "Gráfico", x = "Comprimento da Sépala",
       y = "lagura da Sépala")
```

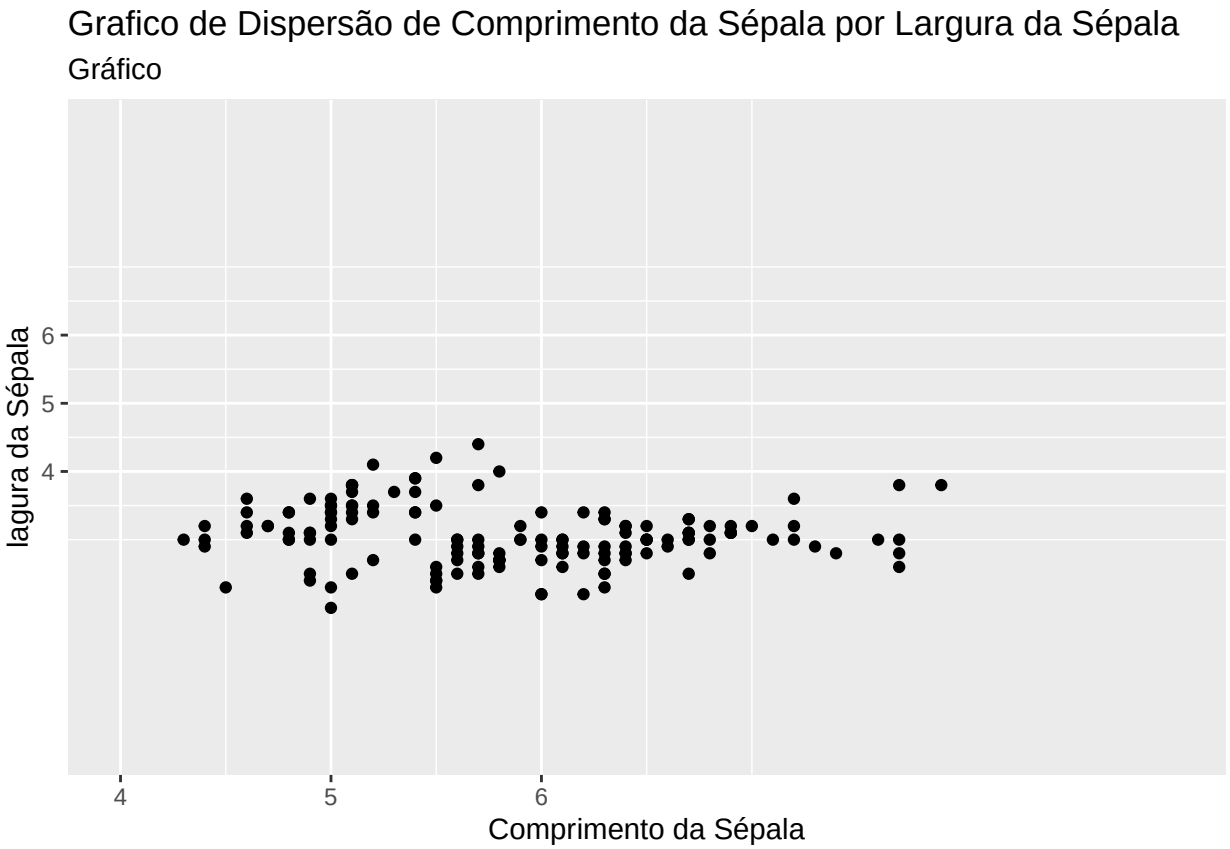


Gráfico de Tendência

O gráfico de tendência auxilia na observação de padrões na plotagem dos casos quando ocorre uma super-plotação dos mesmos. Sua distinção se dá no layer geometrics, em que se acrescenta para além da plotagem de dispersão ("geom_point") a tendência ("geom_smooth").

```
#Geom_ Smooth - grafico de tendencia
ggplot(iris, aes(x = Sepal.Length, y = Petal.Length)) +
  geom_point() +
  geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

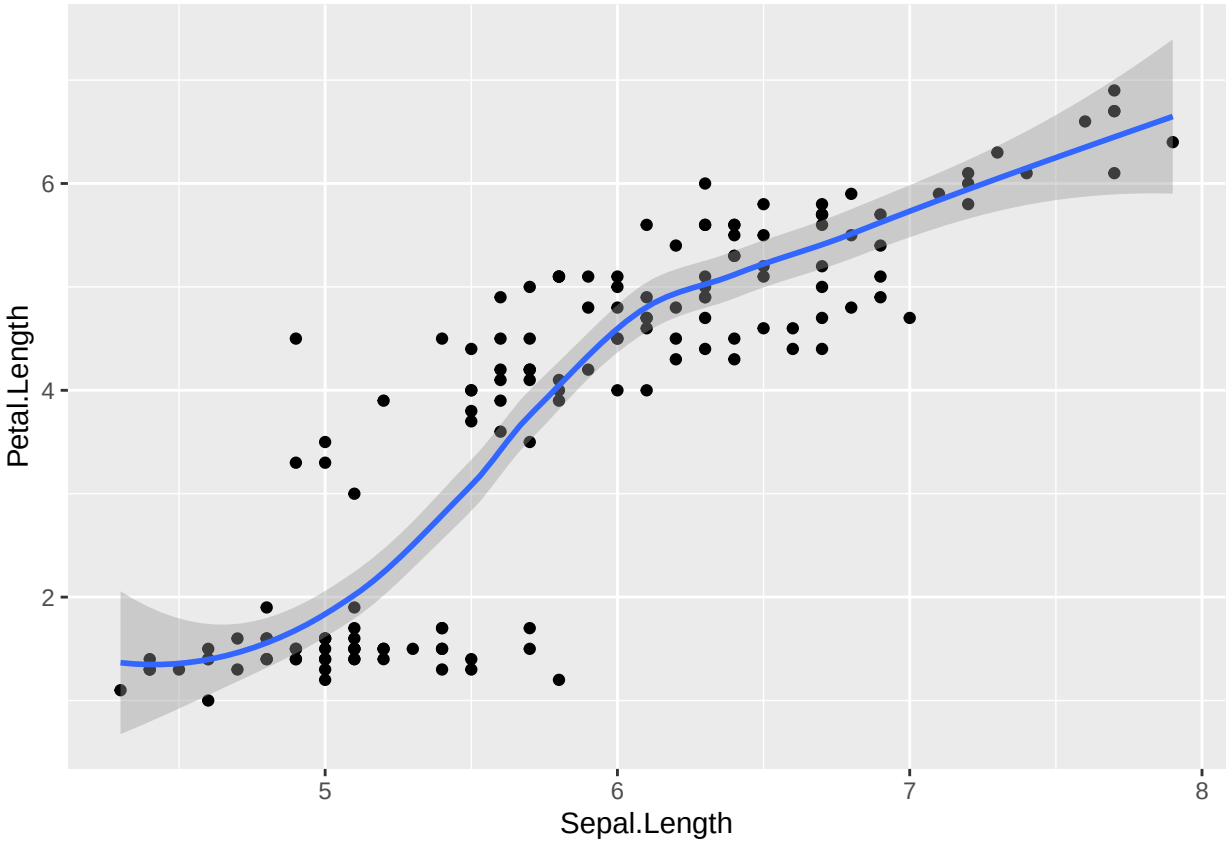
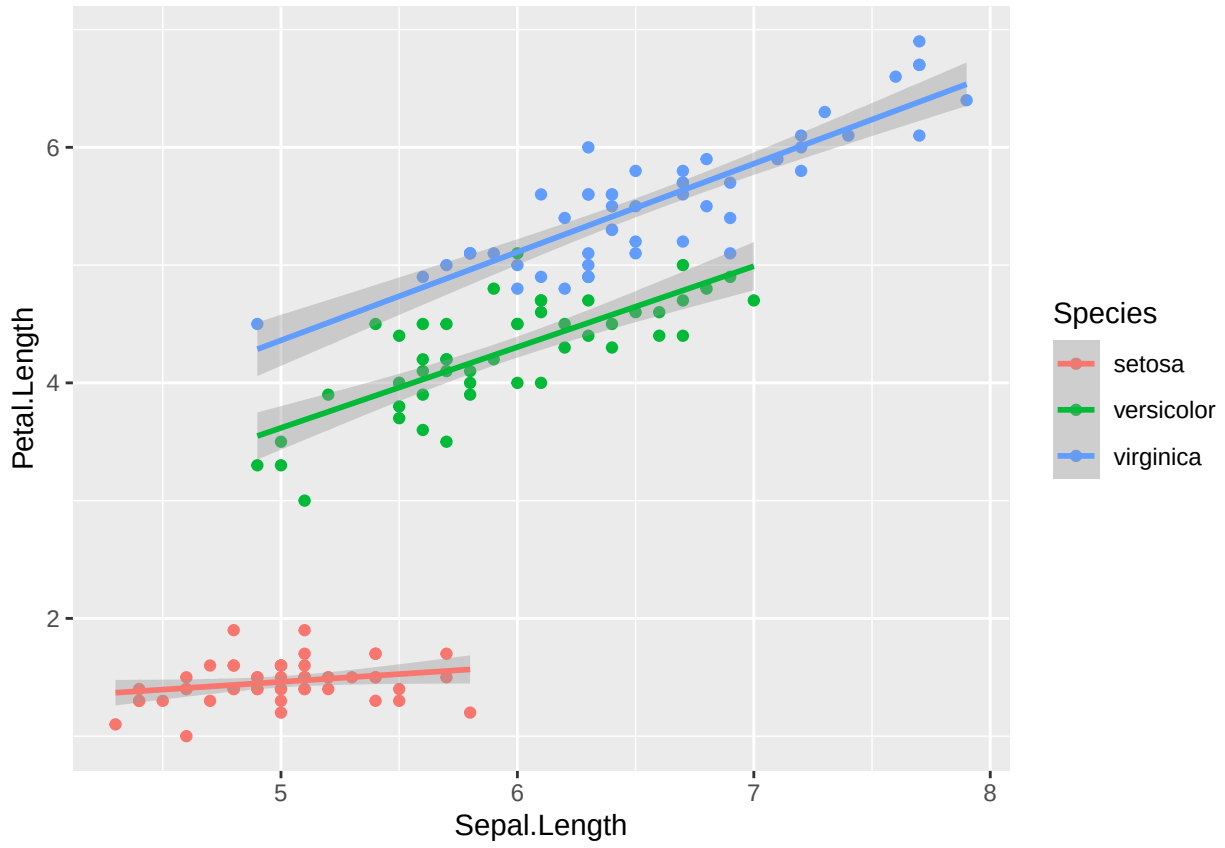


Gráfico de Tendência qualificado por uma variável categórica

É uma repetição da plotagem anterior, porém com uma detalhamento: foi distinguido a tendência na curva de plotagem pela espécie da planta (variável "Species"), deste modo foi produzido três curvas de tendência, cada um representando uma espécie. Esta distinção foi qualificada no layer "geom_smooth" pelo "lm" (um dos vários tipos de critérios para se fazer uma distinção).

```
# grafico de tendencia por especie
ggplot(iris, aes(x = Sepal.Length, y = Petal.Length, col = Species)) +
  geom_point() +
  geom_smooth(method = "lm")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

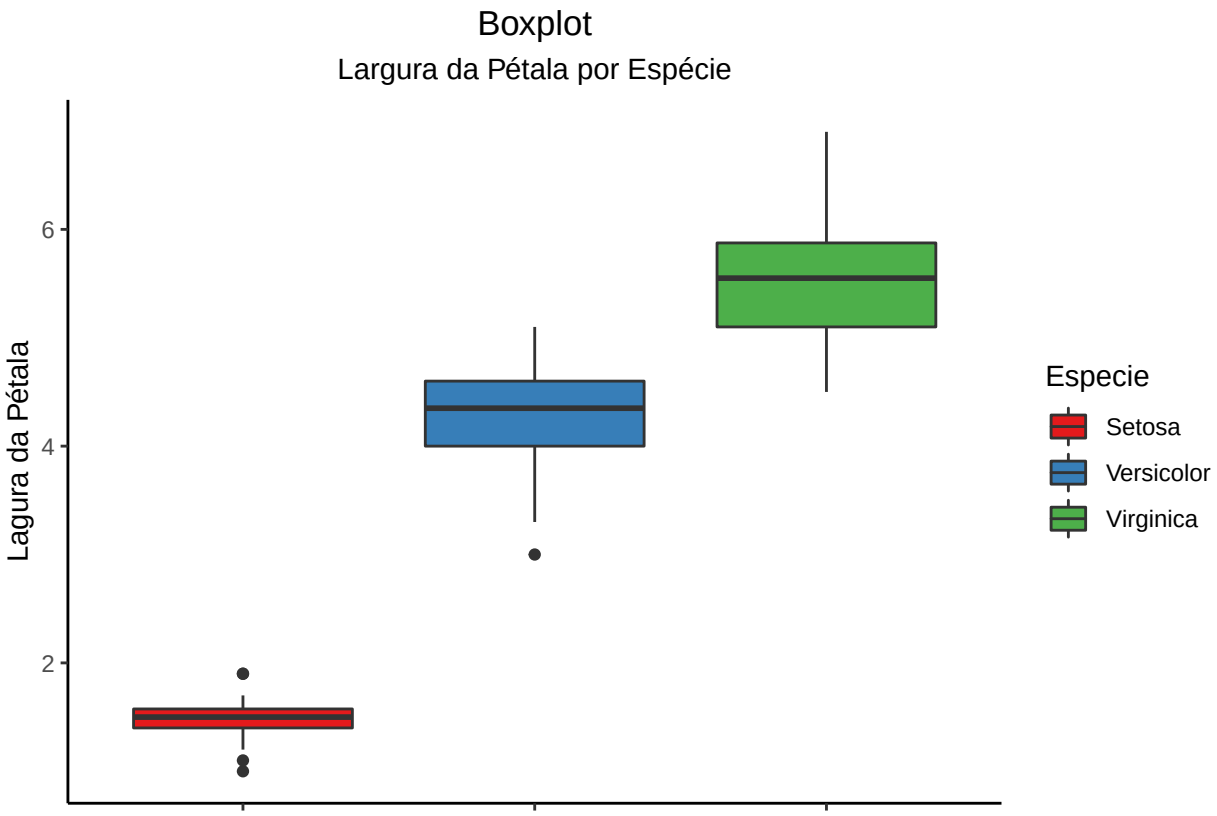


Personalização de Gráficos

A partir de todas as informações apresentadas ao longo deste documento e das dicas na construção de visualizações diversas, apresentaremos a seguir um exemplo de como *customizar* uma determinada visualização a partir dos elementos visto até aqui:

```
# Teste de BoxPlot

# bloxpot
ggplot(bd, aes(x = Species, y = Petal.Length, fill = Species)) +
  geom_boxplot() +
  theme_classic() +
  scale_fill_brewer(palette = "Set1",
                    labels=c("Setosa", "Versicolor", "Virginica")) +
  scale_x_discrete(lab=NULL) +
  labs(title = "Boxplot", subtitle = "Largura da Pétala por Espécie",
       x = "", y = "Lagura da Pétala", fill = "Especie") +
  theme(plot.title = element_text(hjust = 0.5), plot.subtitle = element_text(hjust = 0.5))
```


Boxplot
Largura da Pétala por Espécie
Lagura da Pétala
6
4
2
Especie
Setosa
Versicolor
Virginica