

Universidade Federal do Rio Grande do Norte

Instituto Metr pole Digital

**IMD0601 - Bioestat stica**

# Visualiza  o dos dados em R

Prof. Dr. Tetsu Sakamoto

Instituto Metr pole Digital - UFRN

Sala A224, ramal 182

Email: [tetsu@imd.ufrn.br](mailto:tetsu@imd.ufrn.br)



# Baixe a aula (e os arquivos)

- Para aqueles que não clonaram o repositório:

```
> git clone https://github.com/tetsufmbio/IMD0601.git
```

- Para aqueles que já tem o repositório local:

```
> cd /path/to/IMD0601
```

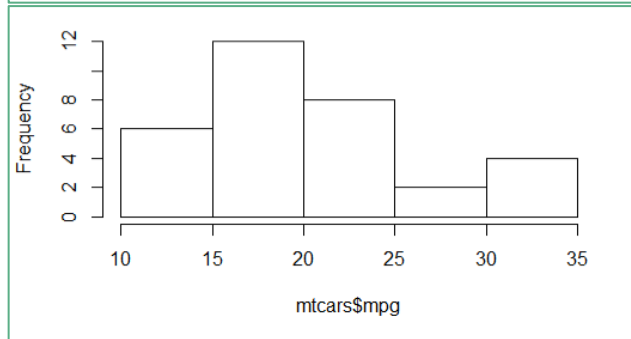
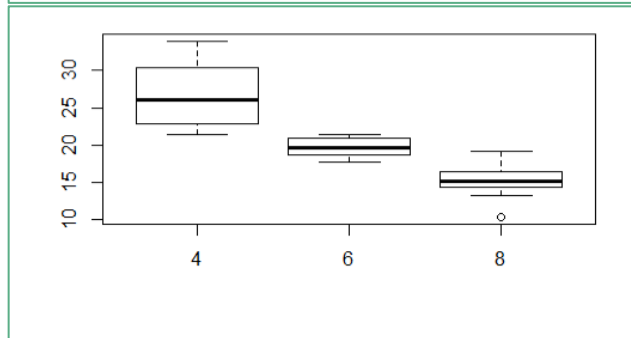
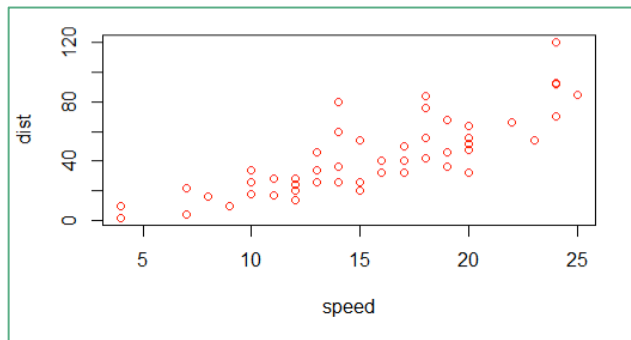
```
> git pull
```

# R Base Graphics

```
plot(x = cars$speed, y = cars$dist,  
     xlab = "Speed", ylab = "Stopping  
Distance", col = 2)
```

```
boxplot(formula = mpg~cyl, data =  
mtcars)
```

```
hist(mtcars$mpg)
```



# ggplot2

Hadley Wickham

“The Grammar of Graphics”

Adiciona camadas nos gráficos  
para melhor visualização dos  
dados;

```
library(ggplot2)
```

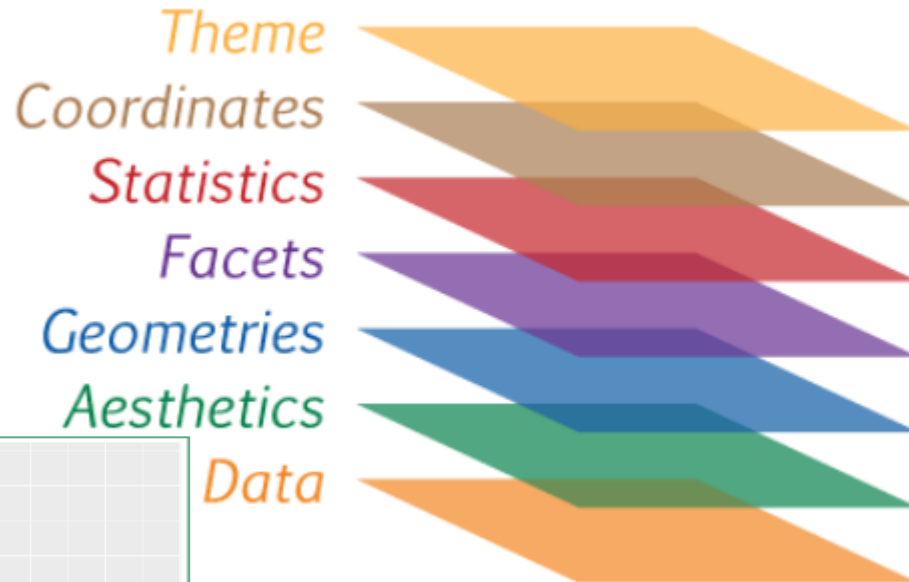
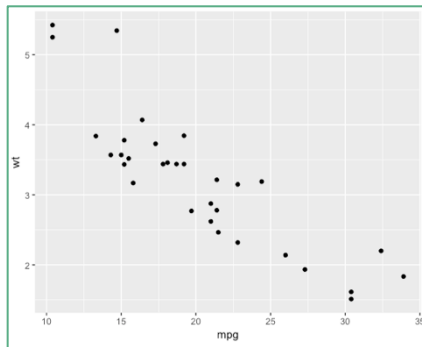


# ggplot2

## Geometries

Camada que indica a forma como os dados devem ser apresentados no gráfico.

```
ggplot(mtcars, aes(x=mpg, y=wt)) + geom_point()
```



# Iris dataset

```
data(iris)
```

```
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 ...
```

# ggplot2

## Exercício

Adicione uma coluna na tabela **iris** que corresponda a um identificador único de cada observação.

```
> iris$Flower <- 1:nrow(iris)
```

Crie uma tabela onde as variáveis Length e Width estejam cada uma em uma coluna, como abaixo.

	Species	Flower	part	Length	Width
1	setosa	1	Petal	1.4	0.2
2	setosa	1	Sepal	5.1	3.5

...

# ggplot2

## Exercício

```
> library(tidyr)

> iris$Flower <- 1:nrow(iris)

> iris.wide <- gather(iris, part_measure, val, -Species, -Flower )

> iris.wide <- separate(iris.wide, part_measure,
c("part","measure"))

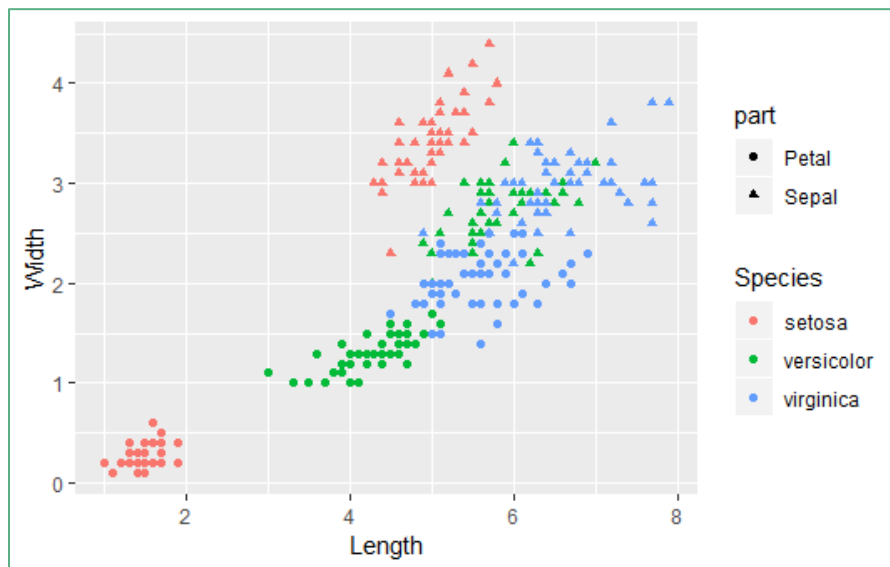
> iris.wide <- spread(iris.wide, measure, val)
```



# ggplot2

## Exercício

Plote um gráfico abaixo usando o ggplot2:

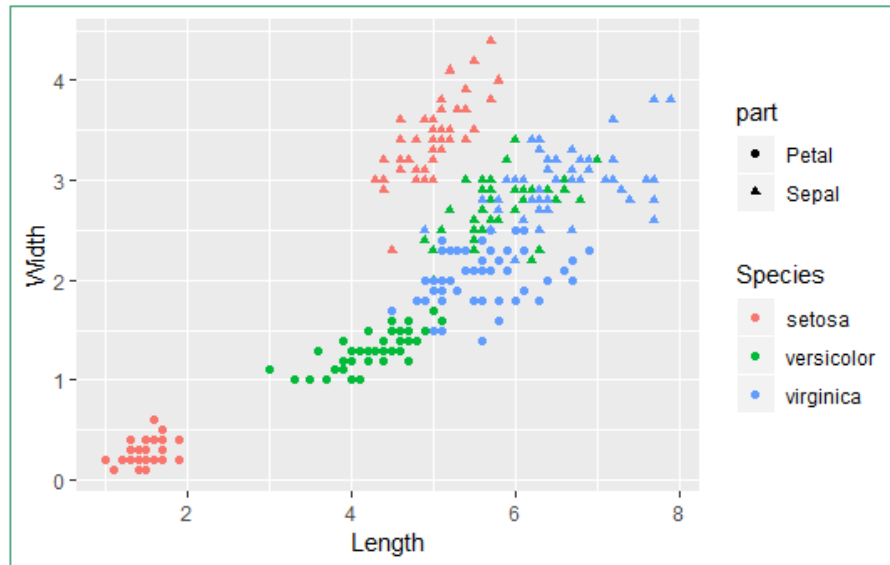


# ggplot2

## Exercício

Plote um gráfico abaixo usando o ggplot2:

```
> ggplot(iris.wide, aes(Length,  
Width, col = Species, shape =  
part)) + geom_point()
```



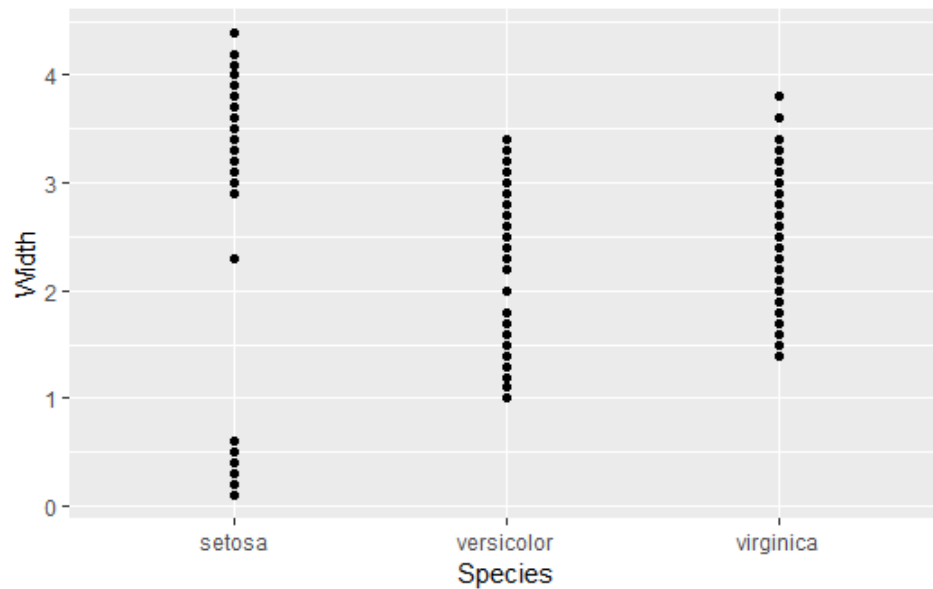
# Parâmetros típicos da estética

- **x** → posição no eixo x;
- **y** → posição no eixo y;
- **col** → cor dos pontos, ou de outras formas;
- **fill** → cor a ser preenchido;
- **size** → diâmetro do ponto, largura da linha;
- **alpha** → transparência;
- **linetype** → padrão de tracejamento da linha;
- **labels** → texto no gráfico;
- **shape** → formas;

# ggplot2

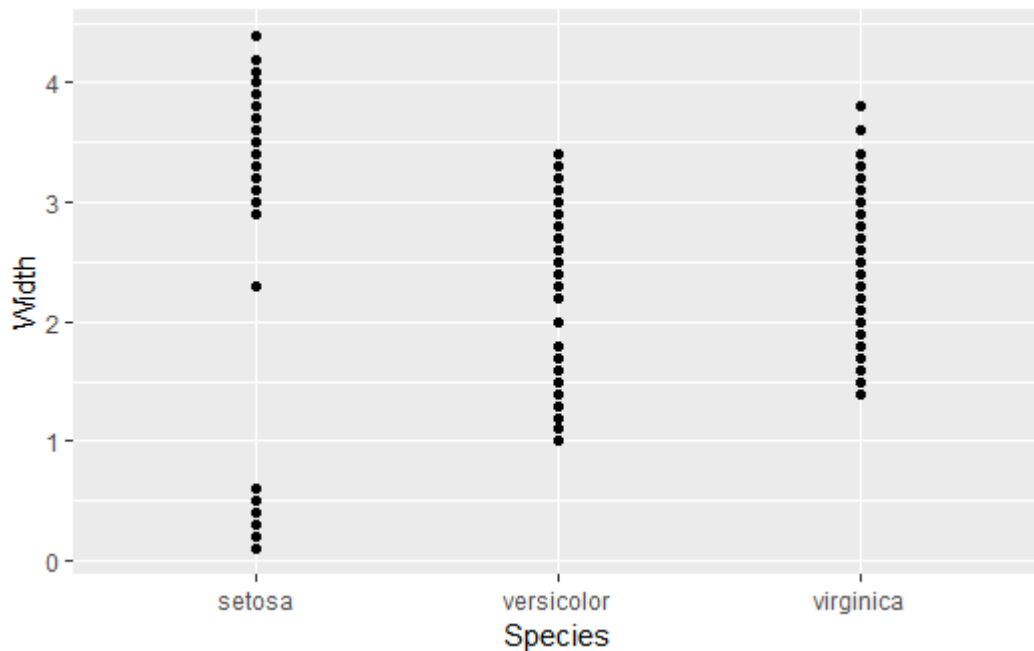
## Exercício

Plote um gráfico abaixo usando o ggplot2:



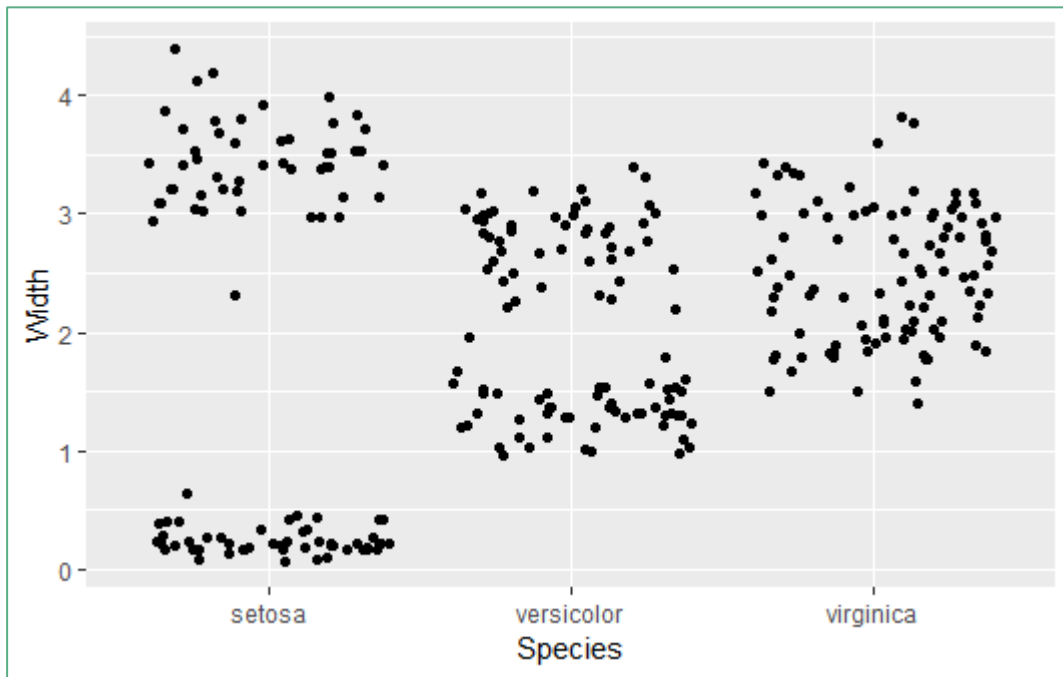
# ggplot

```
> ggplot(iris.wide, aes(Species, Width)) + geom_point()
```



# ggplot

```
ggplot(iris.wide, aes(Species, Width)) + geom_point(position = "jitter")
```

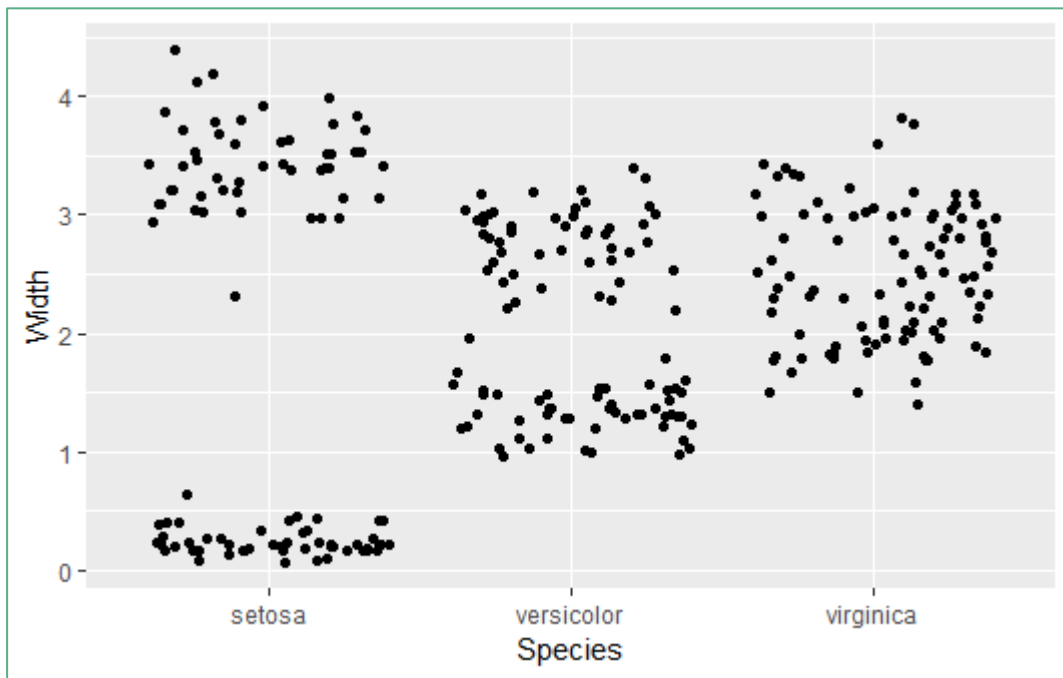


position:

- identity
- jitter
- dodge
- stack
- fill

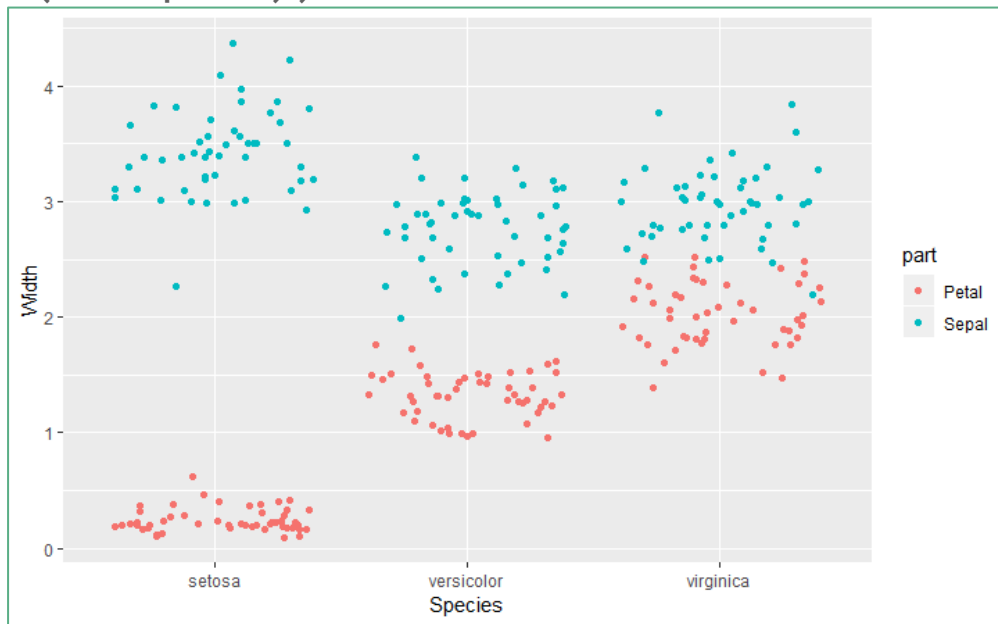
# ggplot

```
> ggplot(iris.wide, aes(Species, Width)) + geom_jitter()
```



# ggplot

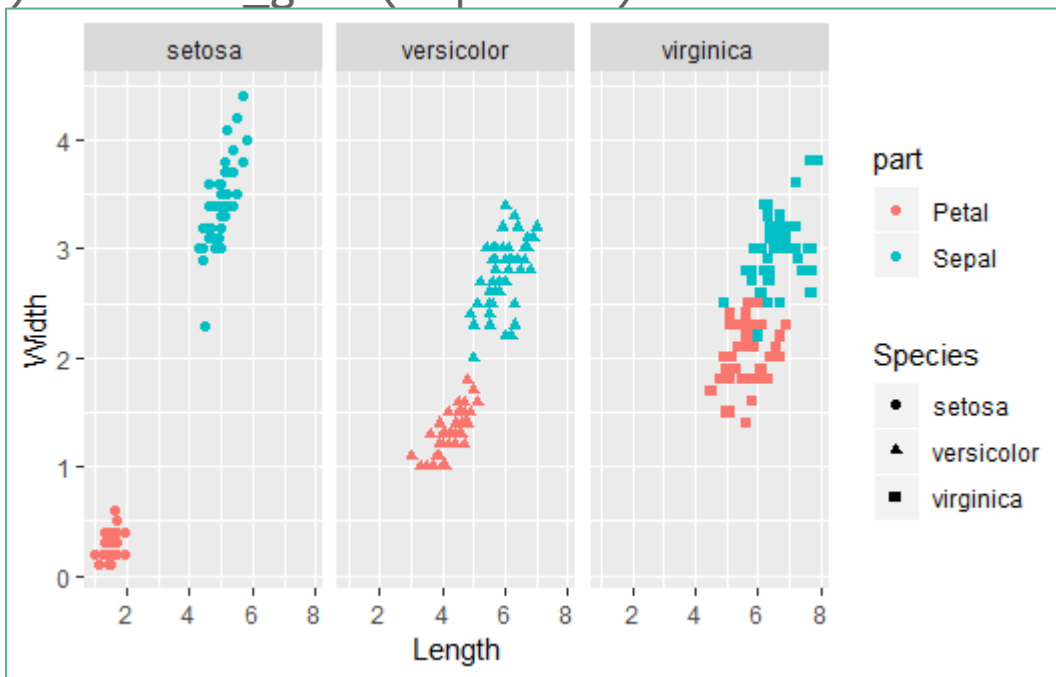
```
> ggplot(iris.wide, aes(Species, Width)) +  
  geom_jitter(aes(col=part))
```





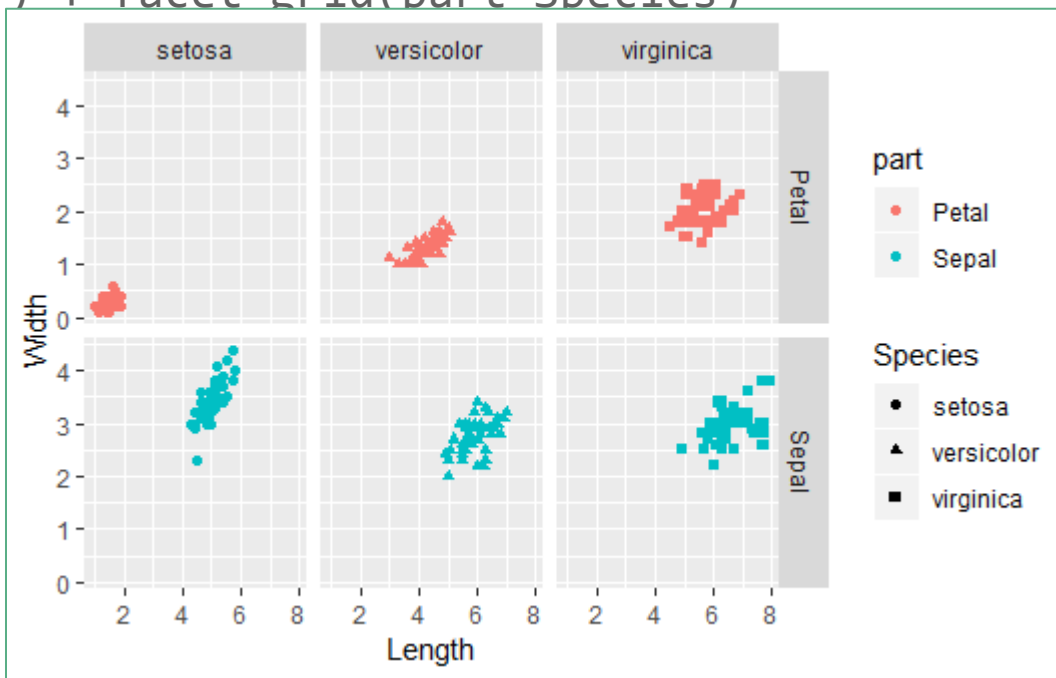
# ggplot2 - facet

```
ggplot(iris.wide, aes(Length, Width, col = part, shape = Species))  
+ geom_point() + facet_grid(~Species)
```



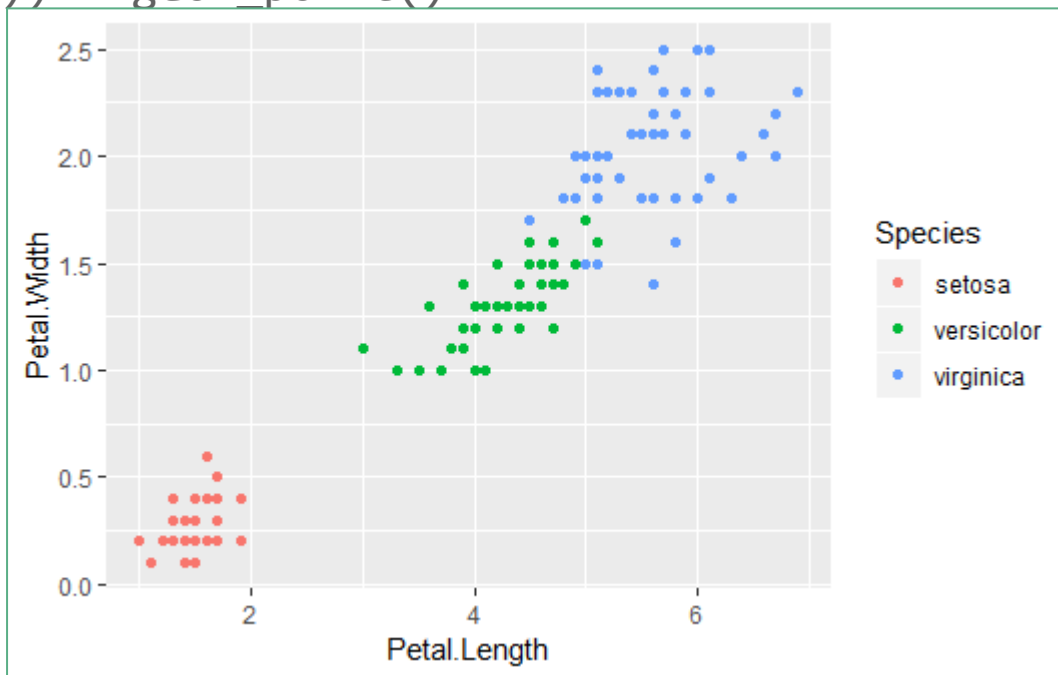
# ggplot2

```
ggplot(iris.wide, aes(Length, Width, col = part, shape = Species))  
+ geom_point() + facet_grid(part~Species)
```



# ggplot2 - adicionando camadas

```
> ggplot(iris, aes(x = Petal.Length, y = Petal.Width, col =  
Species)) + geom_point()
```



Como adicionar  
a média do  
comprimento e  
da largura de  
cada espécie?

## ggplot2 - adicionando camadas

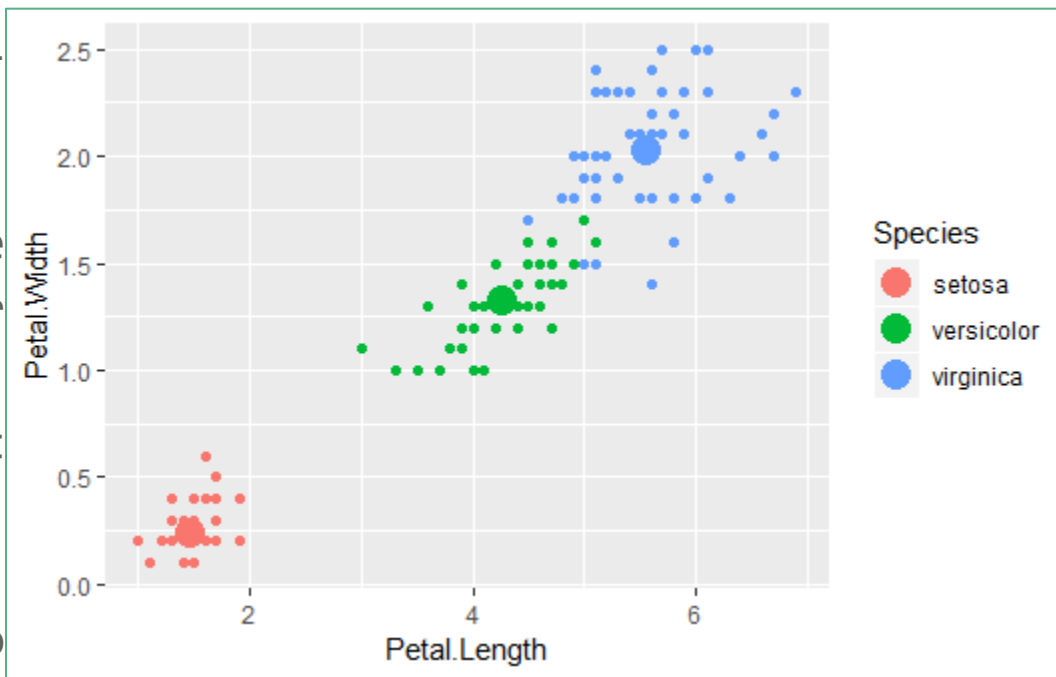
```
> library(dplyr)
> iris.group <- group_by(iris, Species)
> iris.group <- summarise(iris.group, PLM = mean(Petal.Length),
PWM = mean(Petal.Width))
> str(iris.group)

> g <- ggplot(iris, aes(x = Petal.Length, y = Petal.Width, col =
Species)) +
  geom_point()
> g + geom_point(data = iris.group, aes(x = PLM, y = PWM), size =
5)
```

# ggplot2 - adicionando camadas

```
> library(dplyr)
> iris.group <- iris %>% group_by(Species)
> iris.group
Petal.Width
> str(iris.group)

> g <- ggplot(iris, aes(Petal.Length,
  Species)) +
  geom_point(
    size = 5)
> g + geom_point(aes(Petal.Length,
  Species),
  size = 5)
```



Petal.Length),

width, col =

PWM), size =

# ggplot2 - Camada Geométrica

37 geometrias

abline	contour	errorbarh	line	polygon	segment	vline
area	crossbar	freqpoly	linerrange	quantile	smooth	
bar	density	hex	map	raster	step	
bin2d	density2d	histogram	path	rect	text	
blank	dotplot	hline	point	ribbon	tile	
boxplot	errorbar	jitter	pointrange	rug	violin	

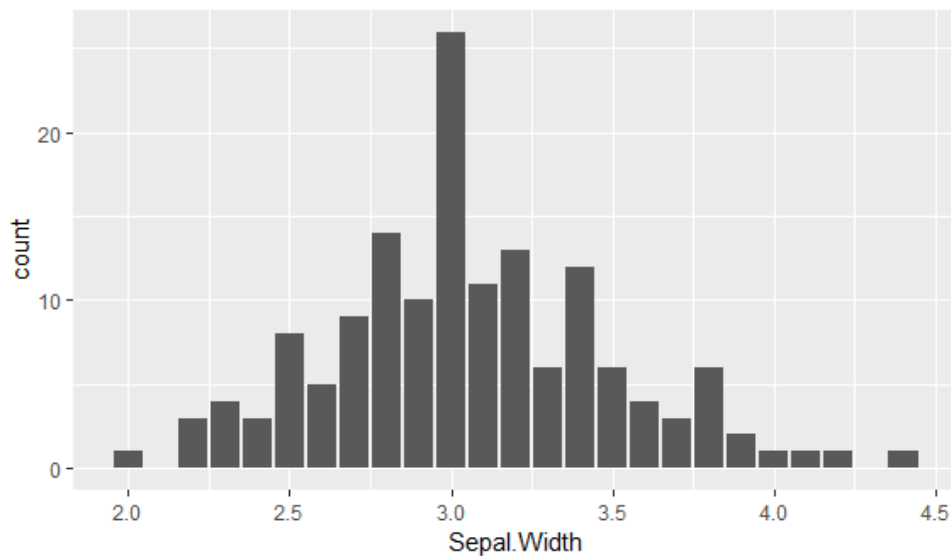
# ggplot2 - gráfico de barra

```
data(iris)
str(iris)
ggplot(iris, aes(Species)) + geom_bar()
```



# ggplot2 - gráfico de barra

```
data(iris)
str(iris)
ggplot(iris, aes(Sepal.Width)) + geom_bar()
```



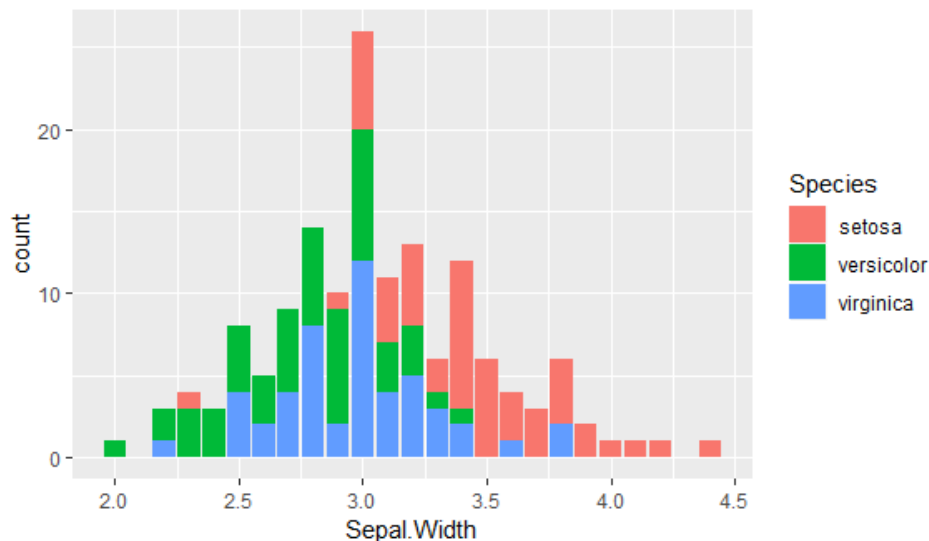


# ggplot2 - gráfico de barra

```
data(iris)
```

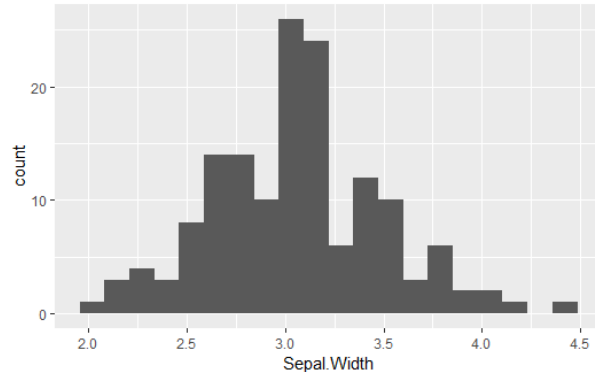
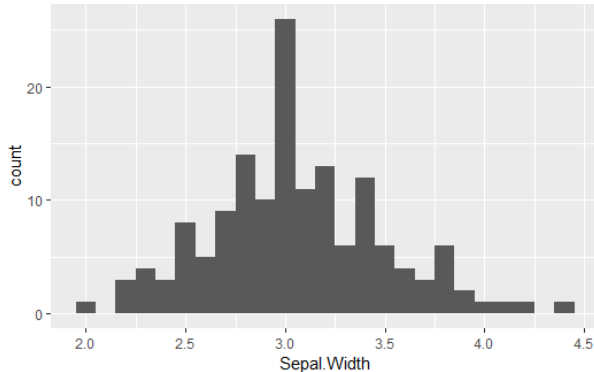
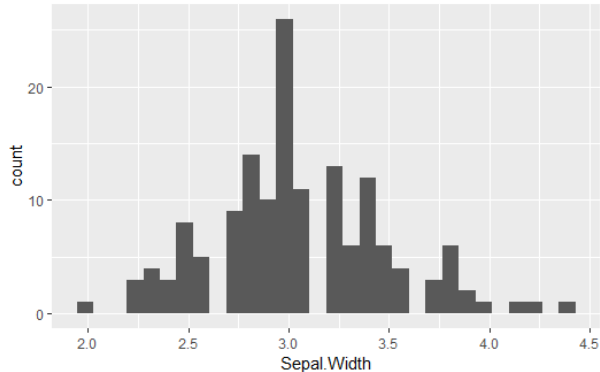
```
str(iris)
```

```
ggplot(iris, aes(Sepal.Width, fill = Species)) + geom_bar()
```



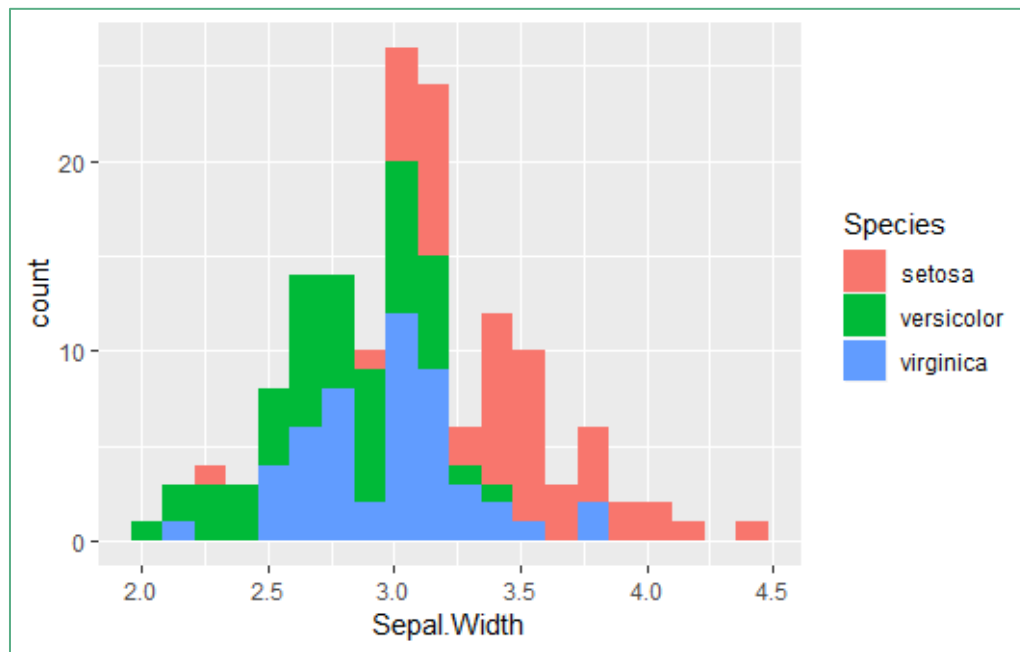
# ggplot2 - Histogramas

```
ggplot(iris, aes(x = Sepal.Width)) + geom_histogram()  
# default: bins = 30, position = stacks;  
ggplot(iris, aes(x = Sepal.Width)) + geom_histogram(binwidth = 0.1)  
ggplot(iris, aes(x = Sepal.Width)) + geom_histogram(bins = 20)
```



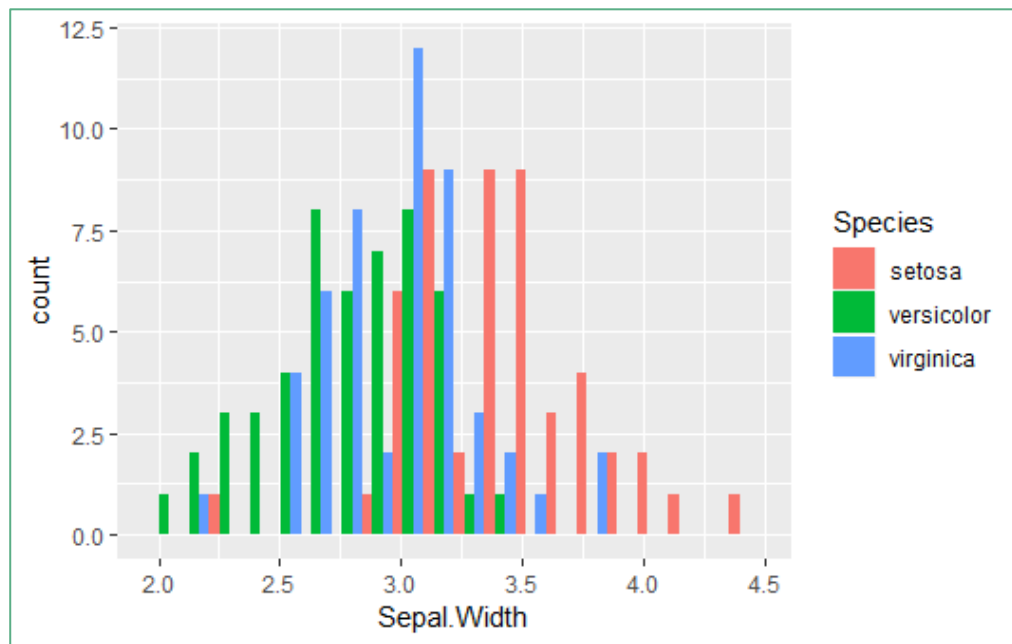
# ggplot2 - Histogramas

```
ggplot(iris, aes(x = Sepal.Width, fill = Species)) +  
geom_histogram(bins = 20)
```



# ggplot2 - Histogramas

```
ggplot(iris, aes(x = Sepal.Width, fill = Species)) +  
geom_histogram(bins = 20, position = "dodge")
```



# ggplot2 - Histogramas

```
ggplot(iris, aes(x = Sepal.Width, fill = Species)) +  
geom_histogram(bins = 20, position = "fill")
```

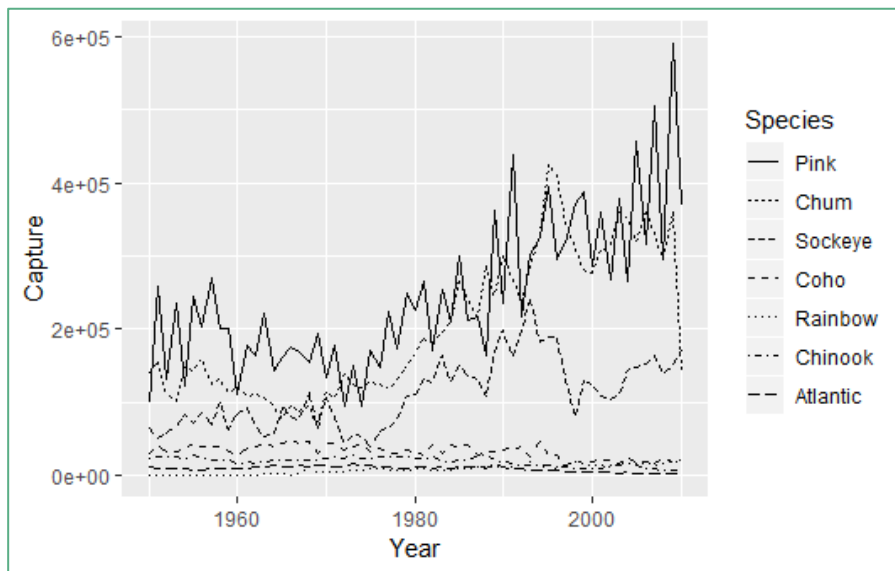


# ggplot2 - gráfico de linha

```
load("fish.RData")
```

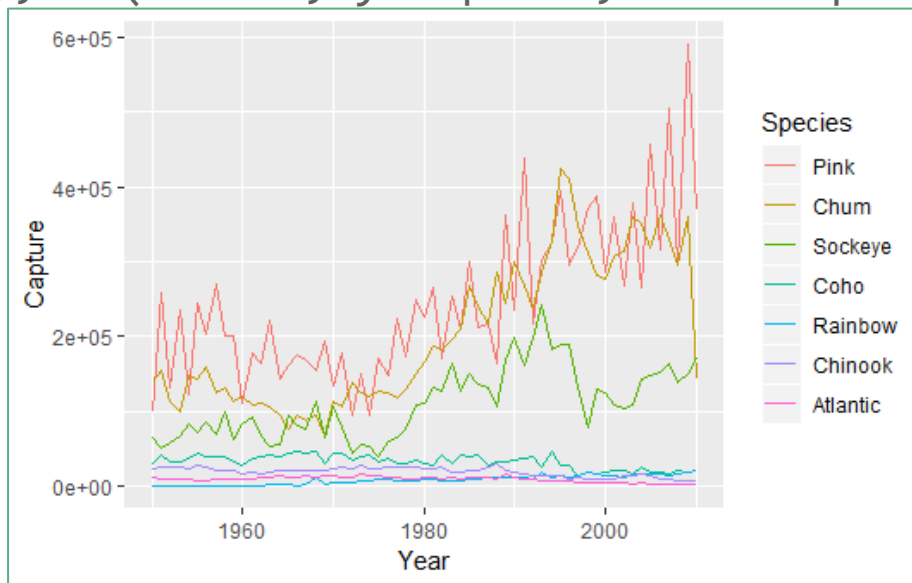
```
str(fish.tidy)
```

```
ggplot(fish.tidy, aes(x=Year, y=Capture, linetype = Species)) +  
geom_line()
```



# ggplot2 - gráfico de linha

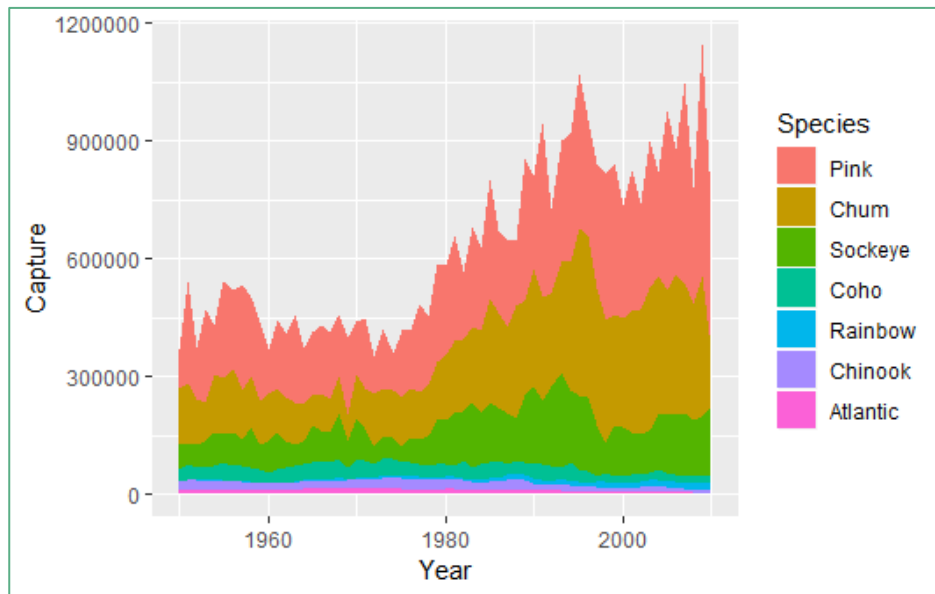
```
load("fish.RData")  
str(fish.tidy)  
ggplot(fish.tidy, aes(x=Year, y=Capture, col = Species)) +  
geom_line()
```



# ggplot2 - gráfico de área

## Exercício

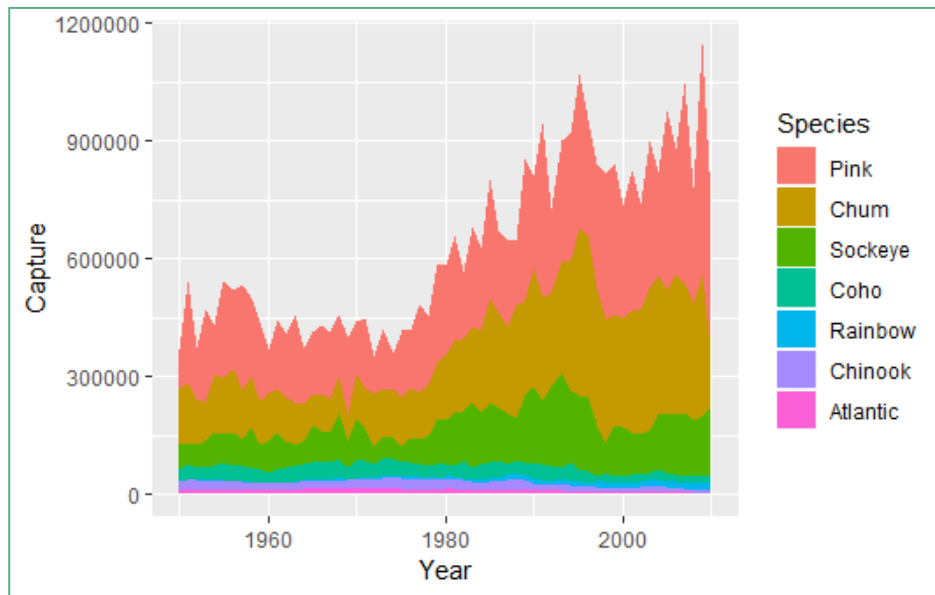
Plote o gráfico abaixo usando o **ggplot2** e os dados **fish.tidy**:





# ggplot2 - gráfico de área

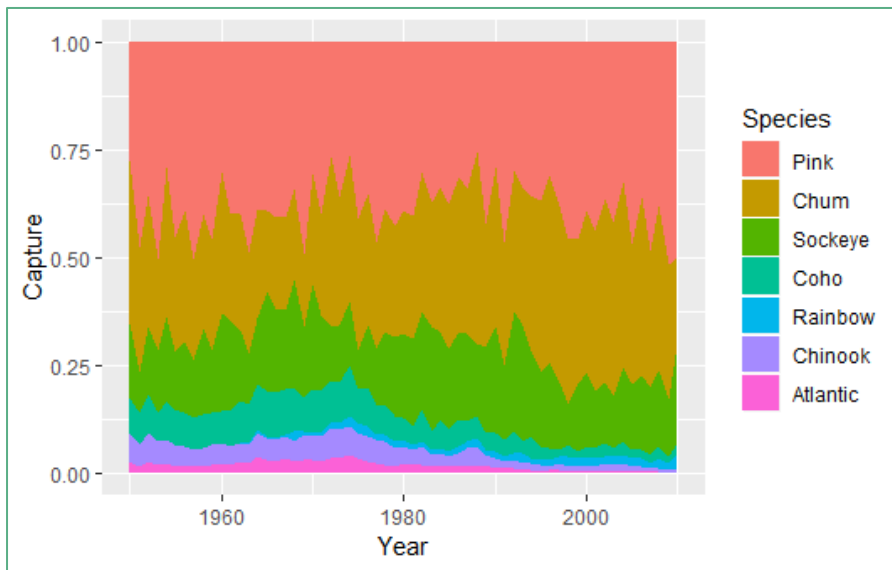
```
load(fish.RData)
str(fish.tidy)
ggplot(fish.tidy, aes(x=Year, y=Capture, fill = Species)) +
  geom_area()
```



# ggplot2 - gráfico de área

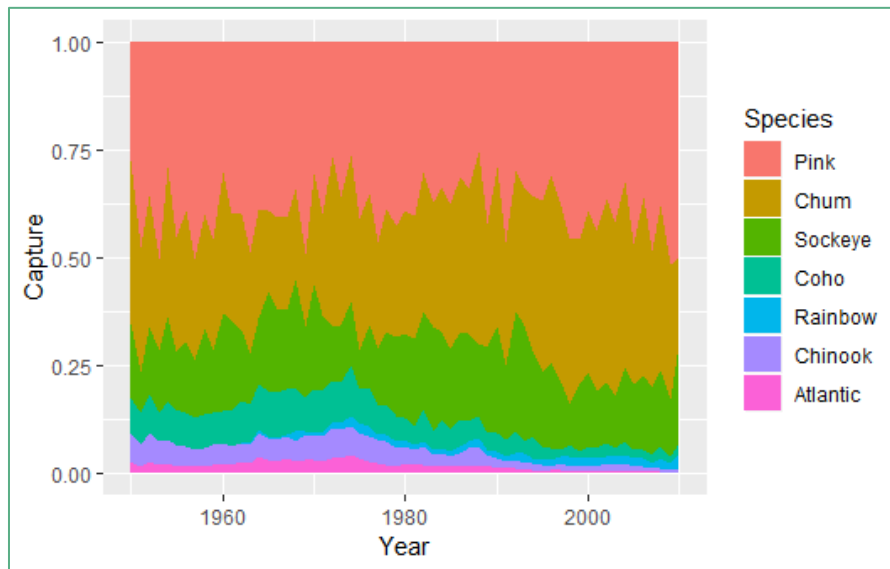
## Exercício

Plote o gráfico abaixo usando o **ggplot2** e os dados **fish.tidy**:



# ggplot2 - gráfico de área

```
load(fish.RData)
str(fish.tidy)
ggplot(fish.tidy, aes(x=Year, y=Capture, fill = Species)) +
  geom_area(position = "fill")
```



# Boas Práticas

## Cor

- Incompatível com daltônicos (principalmente vermelho e verde)
- Paleta incorreta para o tipo de dados (sequencial, qualitativa e divergente)
- Grupos indistinguíveis (cores muito semelhantes)
- Feio (cores primárias de alta saturação)

## Texto

- Illegível (resolução muito pequena e baixa)
- Não descritivo (i.e., "comprimento" - de quê? Quais unidades?)
- Ausência de texto

## Conteúdo informativo

- Muita informação
- Pouca informação
- Nenhuma mensagem ou finalidade clara

## Eixos

- Relação de aspecto ruim
- Supressão da origem
- Eixos x ou y quebrados
- Comum, mas não alinhado

## Estatísticas

- Visualização não corresponde às estatísticas reais

# Boas Práticas

## Geometrias

- Tipo de plotagem errado
- Orientação incorreta

## ***Non-data ink*** (tudo o que não é do próprio dado)

- Uso inadequado

## Gráficos 3D

- Problemas perceptivos
- Terceiro eixo inútil

***Use seu bom senso:***

***Existe algo no meu gráfico que obscurece uma leitura clara do dados ou a mensagem?***



# Referências

Esta aula foi baseada no curso “**Data Visualization with ggplot2 (Part 1)**” de Rick Scavetta: <https://www.datacamp.com/courses/data-visualization-with-ggplot2-1>

Ciência de Dados com R – IBPAD: <https://www.ibpad.com.br/o-que-fazemos/publicacoes/introducao-ciencia-de-dados-com-r/>

Tidyverse: <https://ggplot2.tidyverse.org/index.html> (link para cheat sheet!)

R for data Science: <https://r4ds.had.co.nz/>

<https://skillgaze.com/2017/10/31/understanding-different-visualization-layers-of-ggplot/>