

Experimentation and Statistics in Tropical Agriculture by using RStudio with ggplot2

Clarice GB Demétrio Cristian Villegas Renata A Sermarini

Luiz de Queiroz College of Agriculture

July 25th 2019

Introduction to R and RStudio software

- <https://www.r-project.org>
- <https://www.rstudio.com/products/rstudio/download/>

Some details about RStudio

- Console.
- Environment, History, Files.
- Plots, Packages, Help.
- Script (Maybe you do not have this window)

Designing experiments: CRD

-Consider five varieties of potato with three, five, three, three and four replicates respectively, to be randomized to 18 plots.

```
treat <- rep(c("A", "B", "C", "D", "E"),  
            times = c(3, 5, 3, 3, 4))  
sample(treat)
```

```
[1] "E" "D" "B" "B" "B" "E" "A" "D" "C" "E" "A" "D" "B" "B" "A" "E" "C"  
[18] "C"
```

```
set.seed(123)  
sample(treat)
```

```
[1] "B" "D" "B" "E" "E" "A" "E" "C" "E" "B" "B" "B" "C" "A" "D" "C" "D"  
[18] "A"
```

Designs experiments: RCBD

How would you design an experiment for the vineyard experiment?

- RCBD with three replicates of two treatments and one control.

How to create a data.frame() object

```
rm(list = ls())  
insects <- data.frame(  
  Gender = c("Male", "Female", "Male", "Female"),  
  Trap = c("Orange", "Orange", "Yellow", "Yellow"),  
  n = c(246, 17, 458, 32))  
insects
```

	Gender	Trap	n
1	Male	Orange	246
2	Female	Orange	17
3	Male	Yellow	458
4	Female	Yellow	32

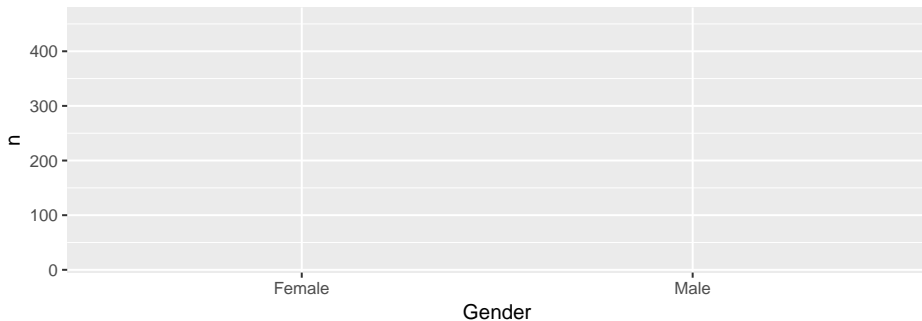
Install ggplot2 R Package

-Do you have ggplot2 package in your computer? Please check writing **library(ggplot2)** in your *Console*

```
#install.packages("ggplot2")  
#vignette(package = "ggplot2")  
#vignette("ggplot2-specs")  
#vignette("extending-ggplot2")
```

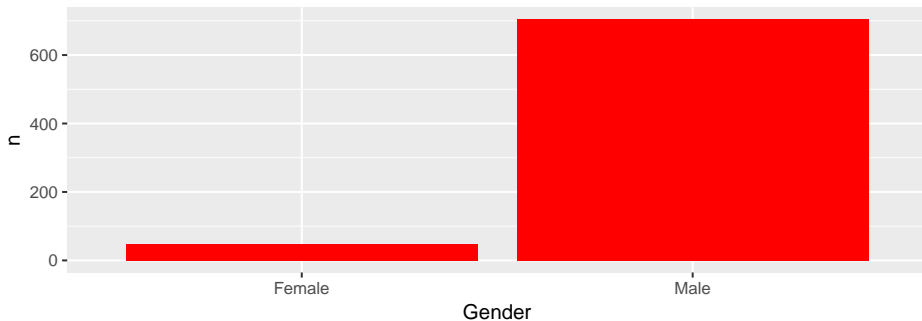
First step in ggplot()

```
library(ggplot2) #Loading ggplot2 R Package  
ggplot(insects, aes(x = Gender, y = n))
```



Second step ggplot()+geom_bar()

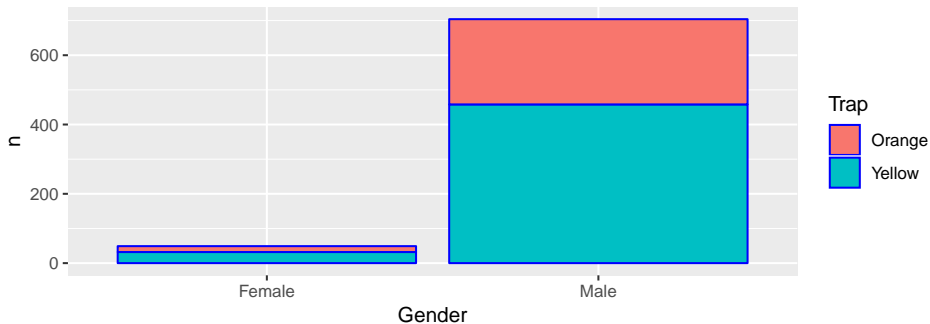
```
ggplot(insects, aes(x = Gender, y = n)) +  
  geom_bar(stat="identity", fill="red")
```



```
# Number of cars in each class:  
#ggplot(mpg, aes(class)) + geom_bar()
```

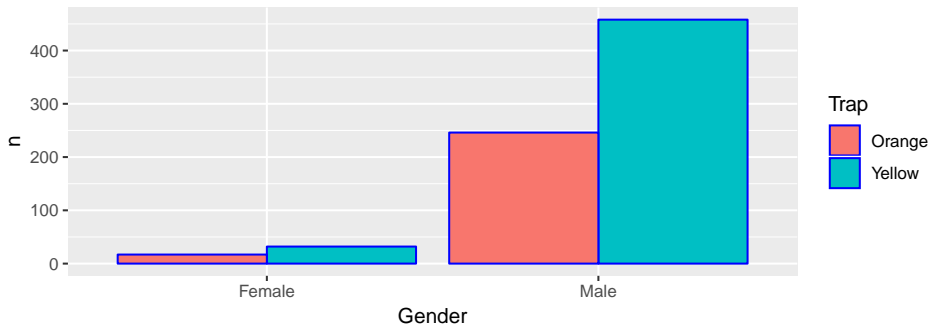
geom_bar()

```
ggplot(insects, aes(x = Gender, y = n, fill=Trap)) +  
  geom_bar(stat="identity", col="blue")
```



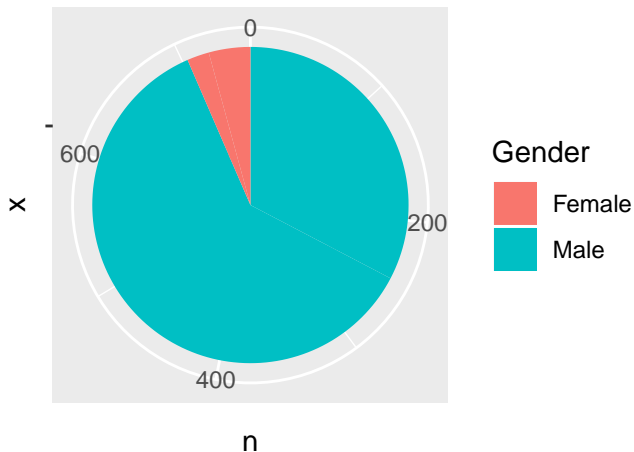
geom_bar()

```
ggplot(insects, aes(x = Gender, y = n, fill = Trap)) +  
  geom_bar(stat="identity", col="blue",  
           position="dodge")
```



Pie Chart

```
ggplot(insects, aes(x = "", y = n, fill = Gender)) +  
  geom_bar(stat = "identity") + coord_polar("y")
```



A factorial example

Table shows the total weight of 24 six-week-old chicks. The treatments, twelve different methods of feeding, consisted of all combinations of three factors: level of protein at three levels, type of protein at two levels, and level of fish solubles, at two levels. The resulting $3 \times 2 \times 2$ factorial experiment was independently replicated in two different houses, which we treat as blocks in a randomized block experiment.

A factorial example

Table 1: Total weight (g) of six-week-old chicks.

Protein	Level of de protein	Level of fish solubles	House		Mean
			I	II	
Groundnut	0	0	6559	6292	6425.5
		1	7075	6779	6927.0
	1	0	6564	6622	6593.0
		1	7528	6856	7192.0
	2	0	6738	6444	6591.0
		1	7333	6361	6847.0
Soybean	0	0	7094	7053	7073.5
		1	8005	7657	7831.0
	1	0	6943	6249	6596.0
		1	7359	7292	7325.5
	2	0	6748	6422	6585.0
		1	6764	6560	6662.0

Importing dataset

```
rm(list=ls())
Data <- read.csv2("chicks.csv")

Data <- transform(Data,
                  House = factor(House),
                  Level = factor(Level),
                  Fish = factor(Fish))

#install.packages("dae")
library(dae)
Data$Treatments <- with(Data,
                        fac.combine(
                          list(Type, Level, Fish),
                          combine.levels = TRUE))
```

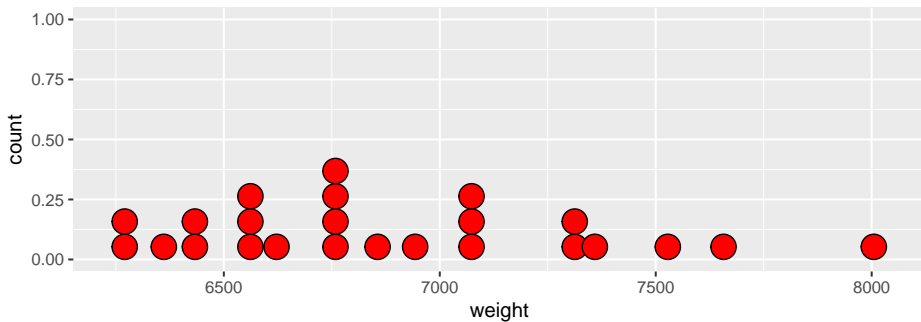
Exploratory Data analysis of dataset

```
summary(Data)
```

House	Type	Level	Fish	weight	Treatments
1:12	Groundnut:12	0:8	0:12	Min. :6249	Groundnut,0,0: 2
2:12	Soybean :12	1:8	1:12	1st Qu.:6560	Groundnut,0,1: 2
		2:8		Median :6772	Groundnut,1,0: 2
				Mean :6887	Groundnut,1,1: 2
				3rd Qu.:7144	Groundnut,2,0: 2
				Max. :8005	Groundnut,2,1: 2
					(Other) :12

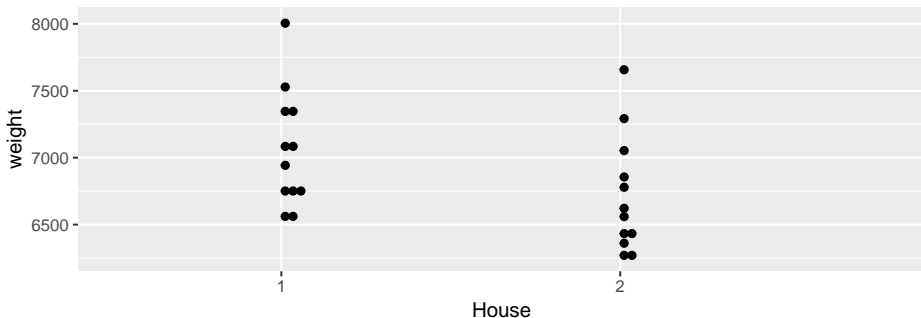
geom_dotplot()

```
ggplot(Data, aes(x = weight)) +  
  geom_dotplot(fill = "red")
```



geom_dotplot()

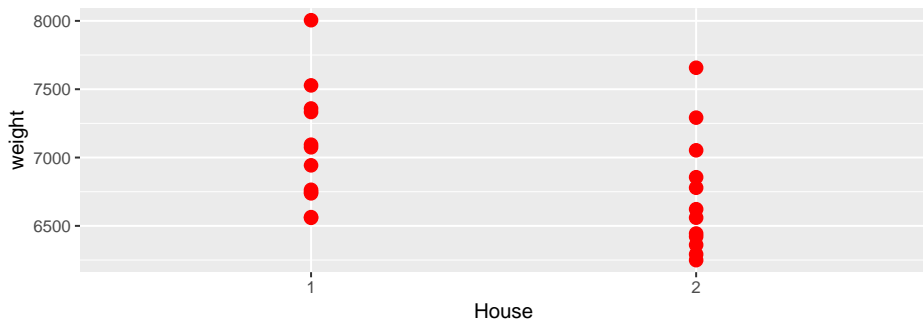
```
ggplot(Data, aes(x=House,y=weight))+  
  geom_dotplot(binaxis = "y")
```



Homework: Create a dotplot with the following categorical variables *Type*, *Level* and *Fish*.

geom_point()

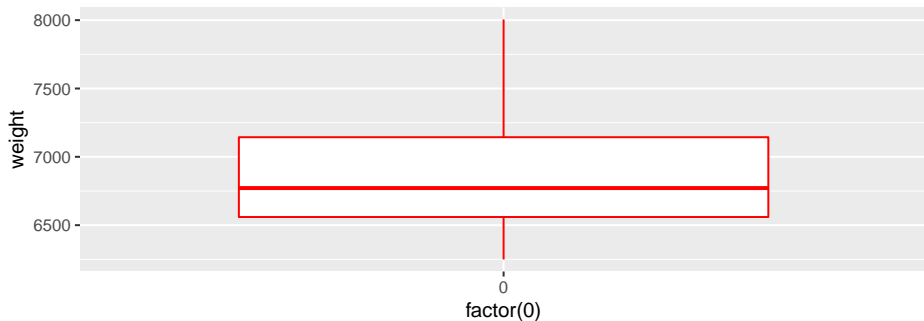
```
ggplot(Data, aes(House, weight)) + geom_point(col="red", lwd=3)
```



Homework: Create a dotplot with the following categorical variables *Type*, *Level* and *Fish*.

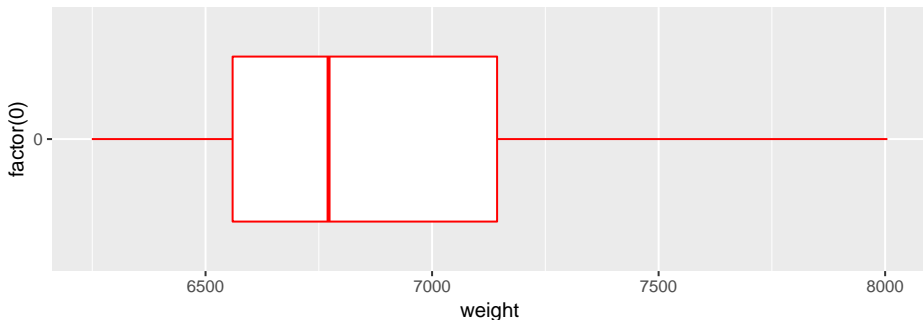
geom_boxplot() with one numerical variable

```
ggplot(Data, aes(x=factor(0),y=weight))+  
  geom_boxplot(col="red",fill="white")
```



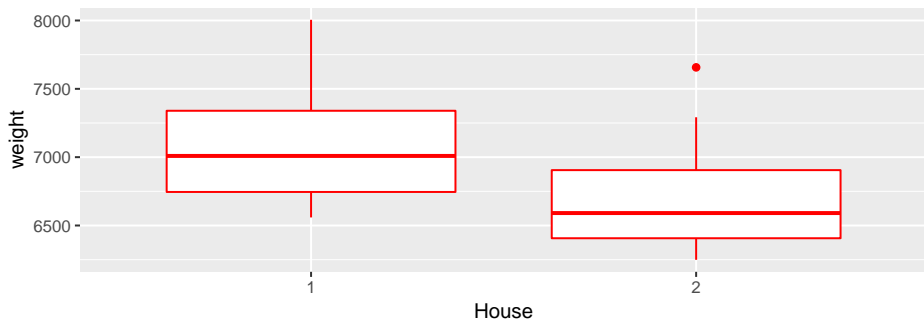
geom_boxplot() with one numerical variable (different option)

```
ggplot(Data, aes(x=factor(0),y=weight))+geom_boxplot(col="red",fill="white")
```



geom_boxplot(): weight separated by House

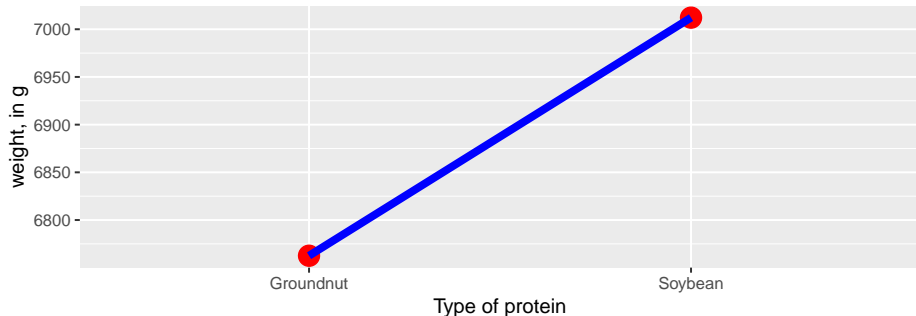
```
ggplot(Data, aes(House, weight))+geom_boxplot(col="red")
```



Homework: Create a boxplot with the following categorical variables *Type*, *Level* and *Fish*.

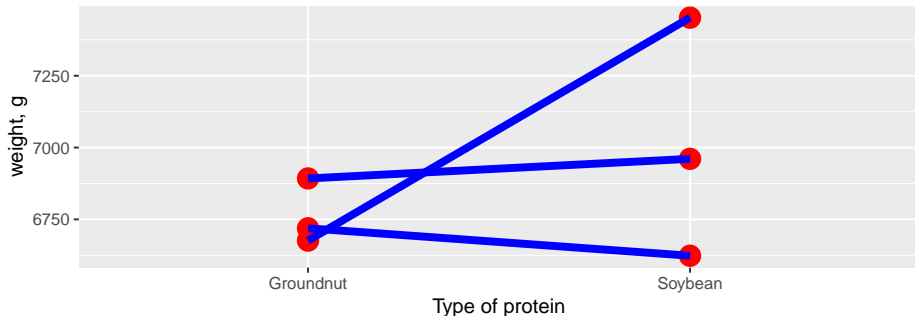
Interaction plot

```
ggplot(Data, aes(x = Type, y = weight, group = 1)) +  
  geom_point(stat = 'summary', fun.y = mean, col="red", lwd=5)+  
  geom_line(stat = 'summary', fun.y = mean, col="blue", lwd=2)+  
  xlab("Type of protein")+ylab("weight, in g")
```



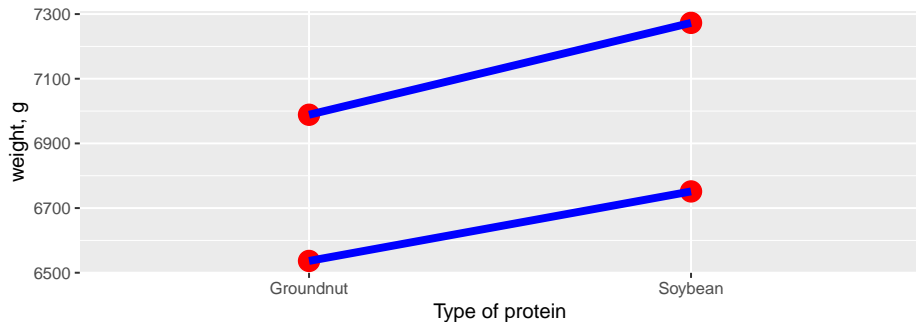
Interaction plot

```
ggplot(Data, aes(x = Type, y = weight, group = Level, color = Level)) +  
  geom_point(stat = 'summary', fun.y = mean, col="red", lwd=5)+  
  geom_line(stat = 'summary', fun.y = mean, col="blue", lwd=2)+  
  xlab("Type of protein")+ylab("weight, g")
```



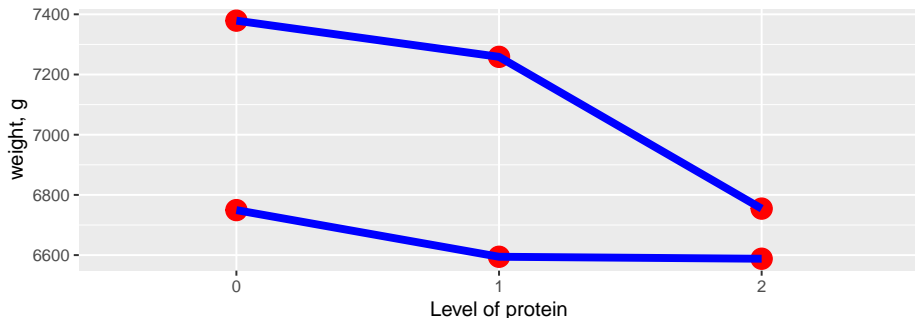
Interaction plot

```
ggplot(Data, aes(x = Type, y = weight, group = Fish, color = Fish)) +  
  geom_point(stat = 'summary', fun.y = mean, col="red", lwd=5)+  
  geom_line(stat = 'summary', fun.y = mean, col="blue", lwd=2)+  
  xlab("Type of protein")+ylab("weight, g")
```



Interaction plot

```
ggplot(Data, aes(x = Level, y = weight, group = Fish, color = Fish)) +  
  geom_point(stat = 'summary', fun.y = mean, col="red", lwd=5)+  
  geom_line(stat = 'summary', fun.y = mean, col="blue", lwd=2)+  
  xlab("Level of protein")+ylab("weight, g")
```



geom_bar() + geom_errorbar()

-The following data refer to heights (meters, averages of 25 plants/plot) of 7-year-old *Eucalyptus grandis* (in 1980) of three randomized complete block designs with 6 treatments (progenies). The three RCBD were installed at three different farms.

```
Data2 <- read.csv2("eucalyptus.csv")  
head(Data2)
```

	Farm	Block	Treat	height
1	L1	B1	T1	20.3
2	L1	B1	T2	21.7
3	L1	B1	T3	22.0
4	L1	B1	T4	20.8
5	L1	B1	T5	21.5
6	L1	B1	T6	19.6

Calculating mean for Farm L1

```
(Data2.L1.mean <- with(subset(Data2, Farm == "L1"),  
                        tapply(height, Treat, mean)))
```

T1	T2	T3	T4	T5	T6
20.625	19.050	23.025	22.500	21.950	19.500

Calculating sd for Farm L1

```
(Data2.L1.sd <- with(subset(Data2, Farm == "L1"),  
                      tapply(height, Treat, sd)))
```

T1	T2	T3	T4	T5	T6
1.978846	2.074448	1.950000	1.856520	0.341565	1.838478

Calculating se for Farm L1

```
(Data2.L1.se <- with(subset(Data2, Farm == "L1"),  
  tapply(height, Treat,  
    function(y){sd(y)/sqrt(length(y))}))
```

T1	T2	T3	T4	T5	T6
0.9894232	1.0372239	0.9750000	0.9282600	0.1707825	0.9192388

Summary

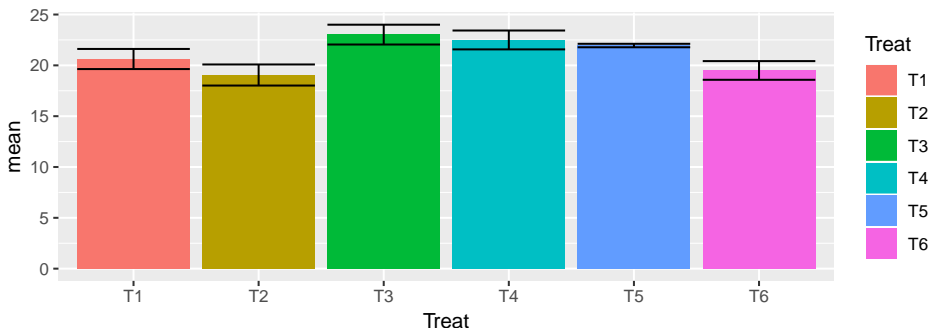
```
Data2.L1.Summary <- data.frame(mean = c(Data2.L1.mean),  
                                sd = c(Data2.L1.sd),  
                                se = c(Data2.L1.se),  
                                Treat = c("T1", "T2", "T3",  
                                           "T4", "T5", "T6"))
```

Data2.L1.Summary

	mean	sd	se	Treat
T1	20.625	1.978846	0.9894232	T1
T2	19.050	2.074448	1.0372239	T2
T3	23.025	1.950000	0.9750000	T3
T4	22.500	1.856520	0.9282600	T4
T5	21.950	0.341565	0.1707825	T5
T6	19.500	1.838478	0.9192388	T6

geom_bar() + geom_errorbar()

```
ggplot(Data2.L1.Summary,  
  aes(x = Treat, y = mean, fill = Treat)) +  
  geom_bar(stat="identity",  
    position=position_dodge(.9)) +  
  geom_errorbar(aes(ymin = mean - se,  
    ymax = mean + se),  
    position=position_dodge(.9))
```



Some references

Book

- Hadley Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016.

Some useful links

- <https://ggplot2.tidyverse.org>
- <http://chris.brien.name/rpackages/>
- <https://cran.r-project.org/web/packages/ggplot2/index.html>