Data cleaning and visualization with R

Ecological and Epidemiological Modeling Madagascar (E2M2)
Institut Pasteur de Madagascar, Antananarivo, Madagascar
ValBio, Ranomafana, Fianarantsoa, Madagascar
09-17 Décembre 2022

*Hafaliana **Christian** Ranaivoson, PhD*
Ecology and Evolution
University of Chicago, Illinois, USA
Laboratoire de Biologie des Populations Parasitaires
Mention of Zoology and Animal Biodiversity
Faculty of Sciences, University of Antananarivo

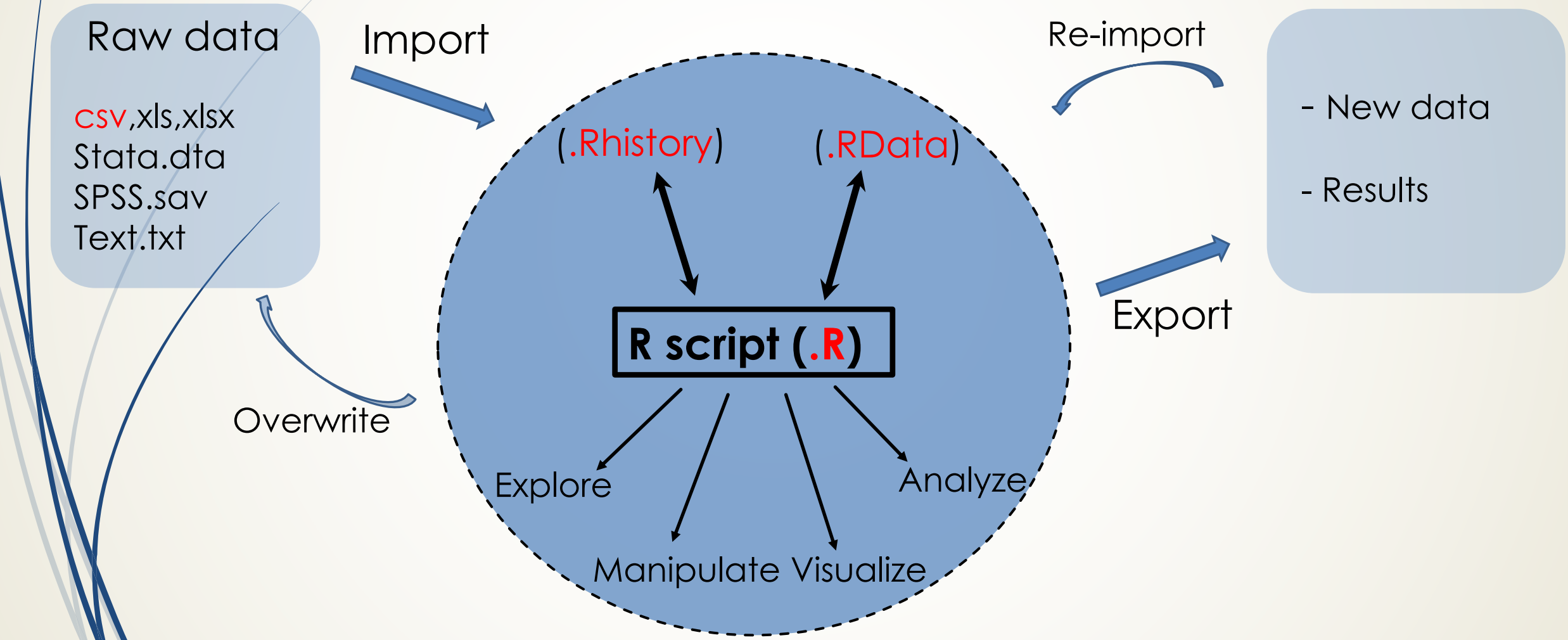
Cleaning and visualizing data in R

- R and RStudio softwares
- Importing data
- Exploring and cleaning data
- Visualizing data
- Tutorial

Database: « e2m2_FB.csv »

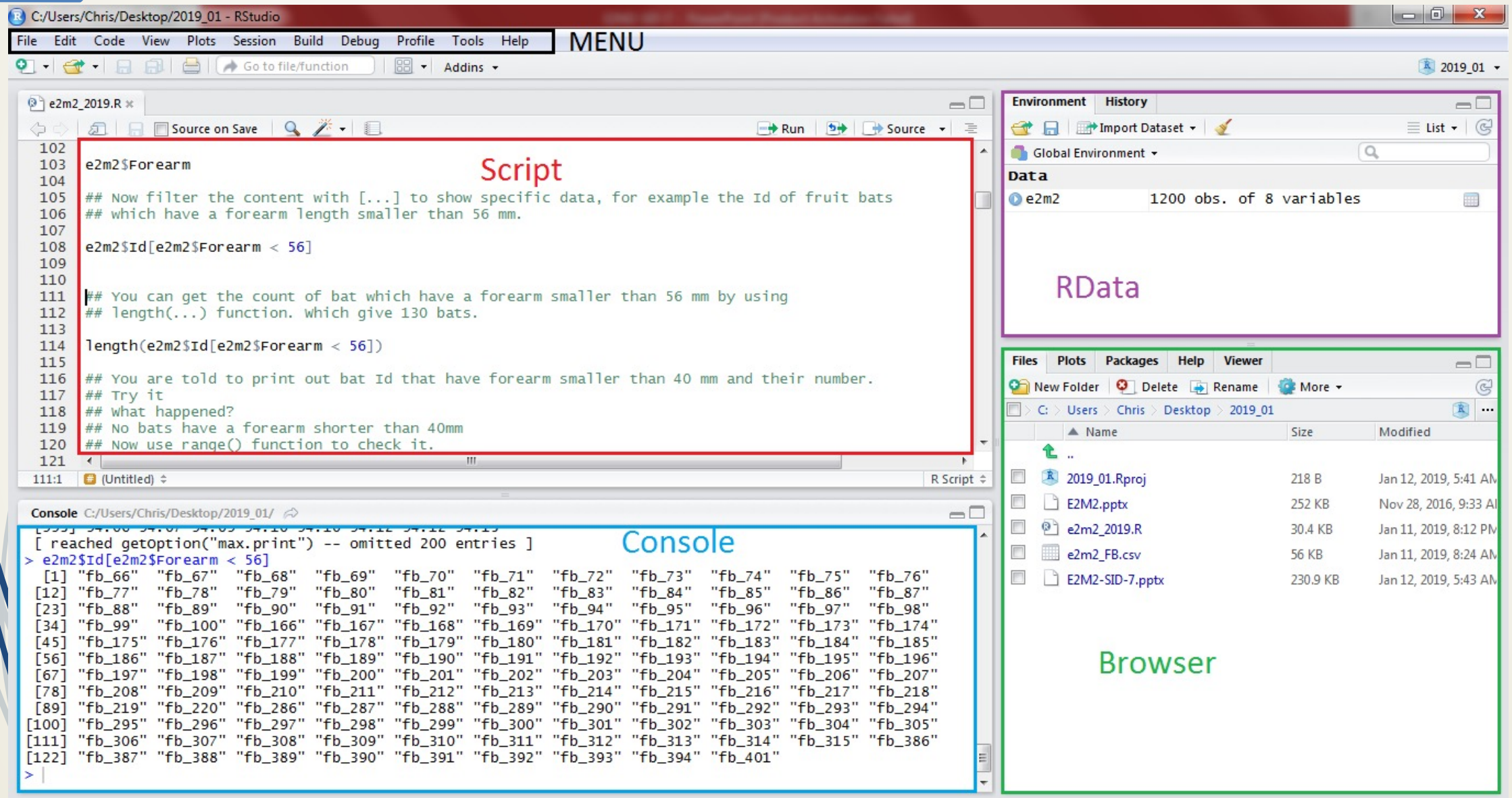
R software (a statistical tool)

- It is free!
- Powerful analysis capability
- Versatile, flexible, open source



RSTUDIO

An interface for R software



Importing Data (loading data into R environment)

- Set working directory (Where to put all files?)

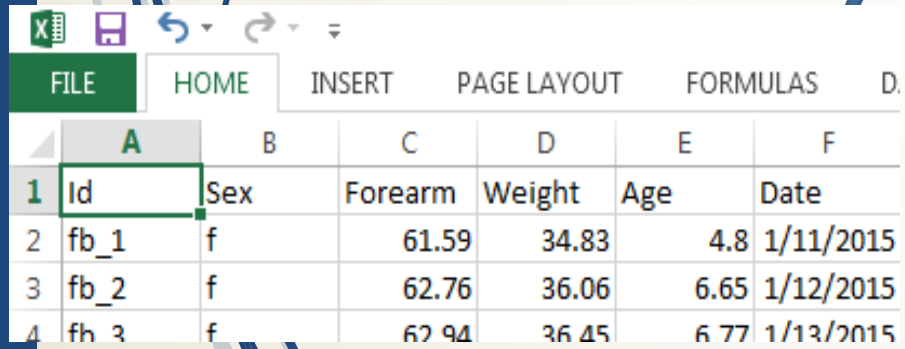
```
getwd()
setwd("Folder path")
```

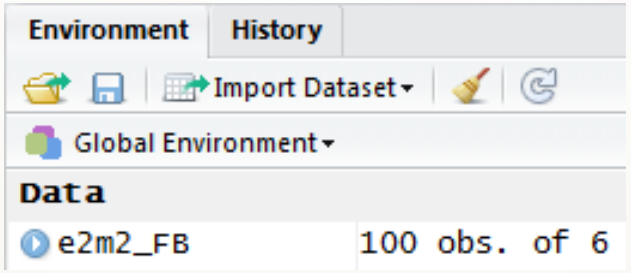
```
?getwd
```

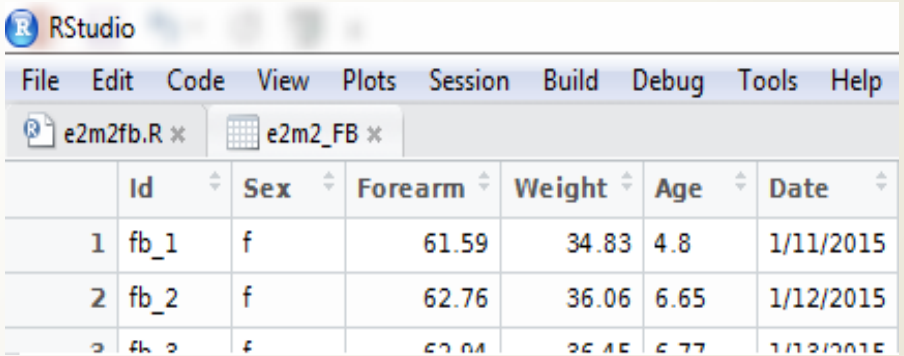
- Import data (read the data source and load it into RData)

```
e2m2_FB <- read.csv("e2m2_FB.csv", header=T, stringsAsFactors=F)

View(e2m2_FB)
```

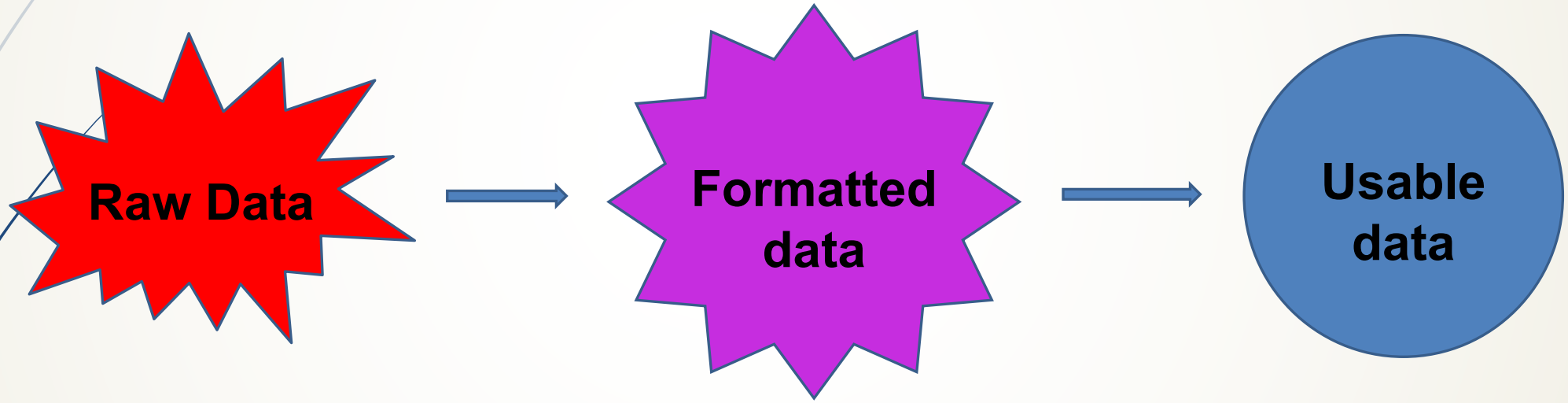






Exploring and cleaning Data (look at the dataset)

Data cleaning steps



Initial data

Technically correct

Consistent

Exploring and cleaning Data (look at the dataset)

Consistent data

Give a sense to your data

- Important step for good data interpretation
- Need a good comprehension of each variables
- Should be reproductible
- Extreme values will affect the outcome
- Make a consistent link between each variable
- No cheating
- Cleaning is not inventing

Exploring and cleaning Data (look at the dataset)

Consistent data

What is wrong with this data?

Id	Site	Date	Sex	Weight	Rainfall	gmam	stes
bat_12	Site_1	41586	F	650	140mm	*NA*	3.90
bat_13	Site_1	41586	f	70	140mm	*NA*	3.89
bat_14	Site_2	41593	M	710	136mm	*NA*	10.05
bat_15	Site_2	41593	M	690	136mm	*NA*	10.02
bat_16	Site_2	41593	F	590	136mm	3.88	*NA*
bat_17	Site_2	41593	M	125	136mm	*NA*	9.95
bat_18	Site_2	41593	M	150	136mm	*NA*	9.64
bat_19	Site_2	41593	F	530	136mm	4.03	*NA*
bat_20	SITE_2	41593	M	130	136mm	*NA*	10.61
bat_21	Site_2	41594	F	640	136mm	4.26	*NA*
bat_22	Site_2	41594	F	590	136mm	4.18	*NA*
bat_23	Site_2	41594	M	145	136mm	*NA*	10.02
bat_23	Site_2	41594	F	520	136mm	3.97	*NA*
bat_25	Site_2	41594	M	150	136mm	*NA*	10.00
bat_26	Site_2	41596	f	650	136mm	4.10	*NA*
bat_27	Site_2	41596	F	165	136mm	4.14	*NA*
bat_28	Site_2	41596	M	130	136mm	*NA*	10.34

Exploring and cleaning Data (look at the dataset)

Data overview

Data frame= e2m2_FB

Variables= Column (names)

Cases = Row (length)

Id	Sex	Forearm	Weight	Age	Date
fb_1	f	61.59	34.83	4.8	1/11/2015
fb_2	f	62.76	36.06	6.65	1/12/2015
fb_3	f	62.94	36.45	6.77	1/13/2015

Value= Contents

```
> dim(e2m2_FB)
[1] 100   6
```

How big is the data frame?

```
> names(e2m2_FB)
[1] "Id"      "Sex"     "Forearm" "Weight"  "Age"     "Date"
```

What are the variables?

Exploring and cleaning Data (look at the dataset)

Accessing dataset contents (From Outside to Inside!)

```
> e2m2_FB$Id
 [1] "fb_1"   "fb_2"   "fb_3"   "fb_4"
 [12] "fb_12"  "fb_13"  "fb_14"...
 [100] "fb_100"
```

Data frame > Variables > Contents

Dataset name $ Variable name

```
> e2m2_FB$Id[e2m2_FB$Forearm < 56]

[1] "fb_64" "fb_65"
```

Filter Contents […]

Data frame name $ Variable name [Filter]

Get the Bat Id with Weight > 75

```
> length(e2m2_FB$Id[e2m2_FB$Forearm < 56])

[1]  2
```

Get the count with length(…)

Exploring and cleaning Data (look at the dataset)

Variable types and error

```
> str(e2m2_FB)

 $ Id      : chr  "fb_1" "fb_2" "fb_3" "fb_4"…
 $ Sex     : chr  "f" "f" "f" "F" "f " ...
 $ Weight : num  34.8 36.1 36.5 36.6 38.9 ...
 $ Age     : chr "one" "6.65" "6.77" "seven" ...
 $ Date   : chr  "1/11/2015" "1/12/2015 "…
```

```
> as.factor(e2m2_FB$Sex)
Levels: f F f  m
> as.numeric(e2m2_FB$Age)
Warning message:
NAs introduced by coercion
> as.Date(e2m2_FB$Date,"%m/%d/%Y")
```

str(…)

Categorical: Factor (n levels)
Continuous: Numeric (Range)
Time: Date (Range)
Binary: logic (T,F)
Missing Value: NA

as.factor(…)
as.Date(…) **"%Y-%m-%d"**
as.numeric(…)

Needed format -> Re-format
Value error -> Correct value
Missing error -> Handle NA values

Exploring and cleaning Data (look at the dataset)

Correcting Values

(Wrong value ← Right Value)

```
> e2m2_FB$Age[e2m2$Age== “one"] <- “1" >
e2m2_FB$Sex[e2m2$Sex== "F"] <- "f"
> e2m2_FB$Sex[e2m2$Sex== "f "] <- "f"
```

```
> as.factor(e2m2_FB$Sex)
Levels: f m
> as.numeric(e2m2_FB$Age)
[1] 4.80  6.65  6.77  7.00
```

Save the format to the variable

```
> e2m2_FB$Sex <- as.factor(e2m2_FB$Sex)
> e2m2_FB$Age <- as.numeric(e2m2_FB$Age)
> e2m2_FB$Date <- as.Date(e2m2_FB$Date,"%m/%d/%Y")
```

```
> str(e2m2_FB)
$ Id     : chr  "fb_1" "fb_2" "fb_3" "fb_4" ...
$ Sex    : Factor w/ 2 levels "f","m": 1 1 1 1  2
$ Age    : num  4.8 6.65 6.77 7 8.89 ...
$ Date   : Date, format: "2015-01-11"
```

```
> e2m2_FB$NewVar <- as.factor(e2m2_FB$Sex)
> e2m2_FB$NewVar <- e2m2_FB$Forearm/2
```

Or create new variable

Visualizing Data (Play with data)

Install and Load Library

```
Install.packages("…")
Installed.packages()
```

```
> library(dplyr)
> require(ggplot2)
```

Data summarizing ("**dplyr**")

```
> fb_male <-filter(e2m2_FB,Sex=="m")
> range(fb_male$Forearm)
> mean(fb_male$Forearm)
> sd(fb_male$Forearm)
```

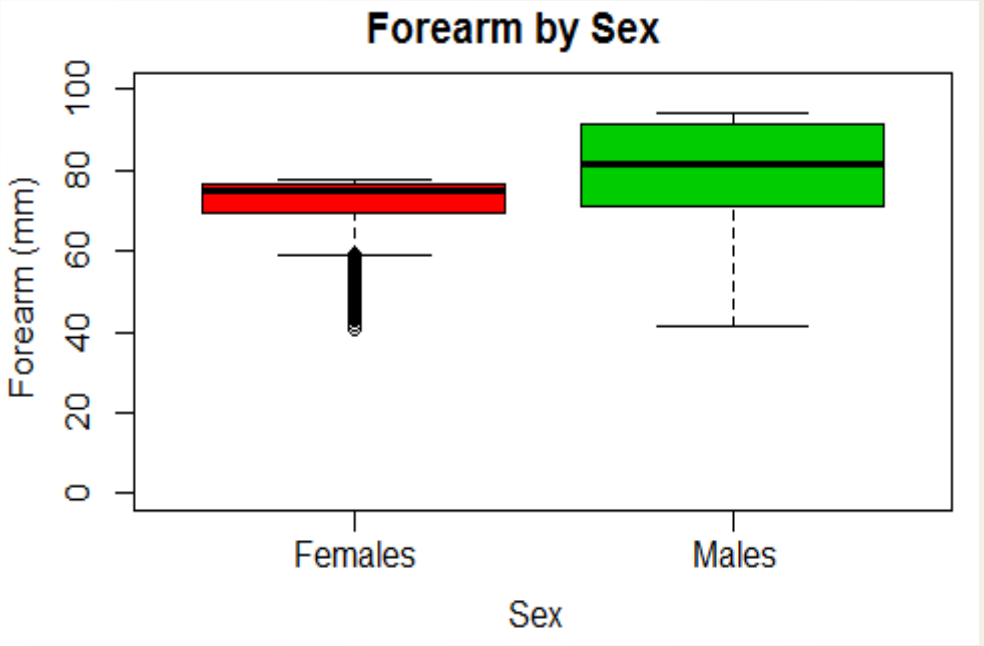
	Sex (fctr)	mean_forearm (dbl)	sd_forearm (dbl)	nbr (int)
1	f	59.6040	11.90278	60
2	m	60.9985	14.12073	40

```
gp_fb <- group_by(e2m2_FB,Sex)
gp_fb_stat <- summarise(gp_fb,
                        mean_forearm=mean(Forearm,na.rm=T),
                        sd_forearm=sd(Forearm,na.rm=T),
                                nbr=n())
```

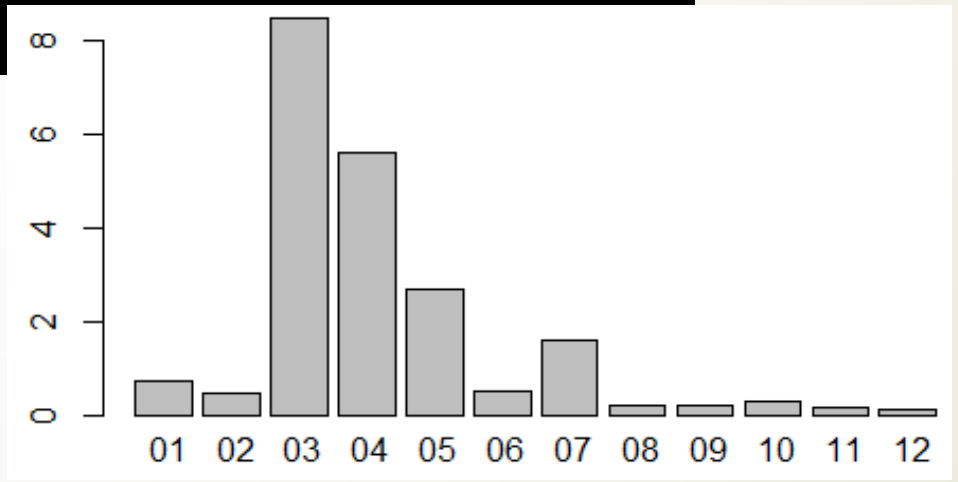
Visualizing Data (Play with data)

R base graphical function

```
boxplot(Forearm~Sex,
        names=c("Females","Males"),
        col=c(2:3),
        main="Forearm by Sex",
        xlab="Sex",
        ylab="Forearm (mm)",
        ylim=c(0,100))
```




```
PLoad <- tapply(e2m2$ParLoad,factor(format(e2m2$Date,"%m")),mean)
barplot(PLoad)
```



Visualizing Data (Present de data)

R plot() function

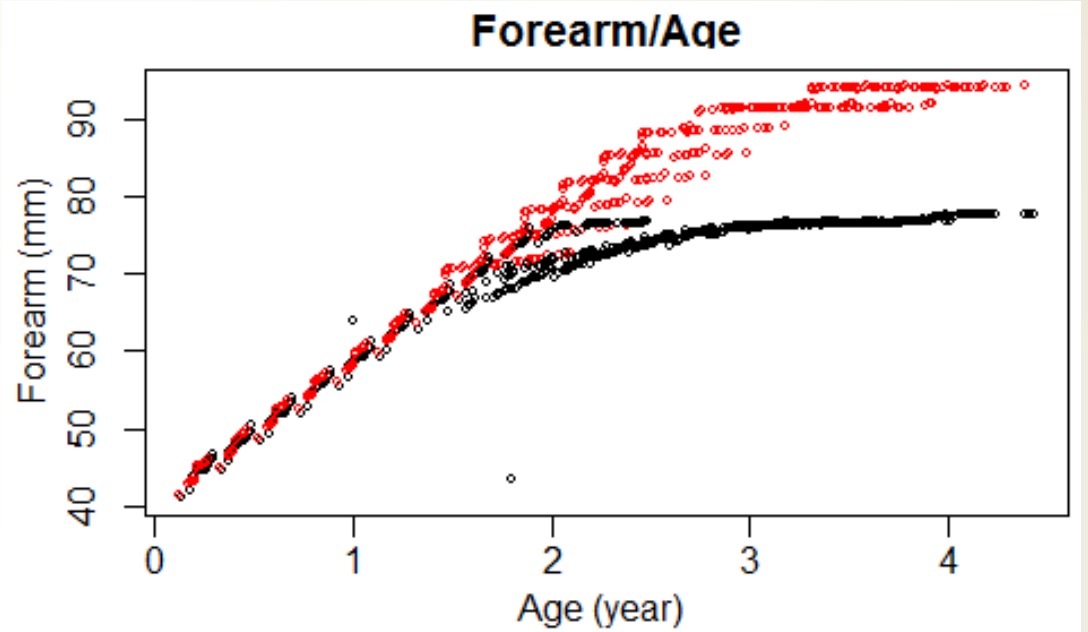
```
plot(e2m2$ Forearm~ e2m2$Age,
       main = "Forearm/Age",
       ylab ="Forearm (mm)", xlab = "Age (year)",
       col=Sex)
```

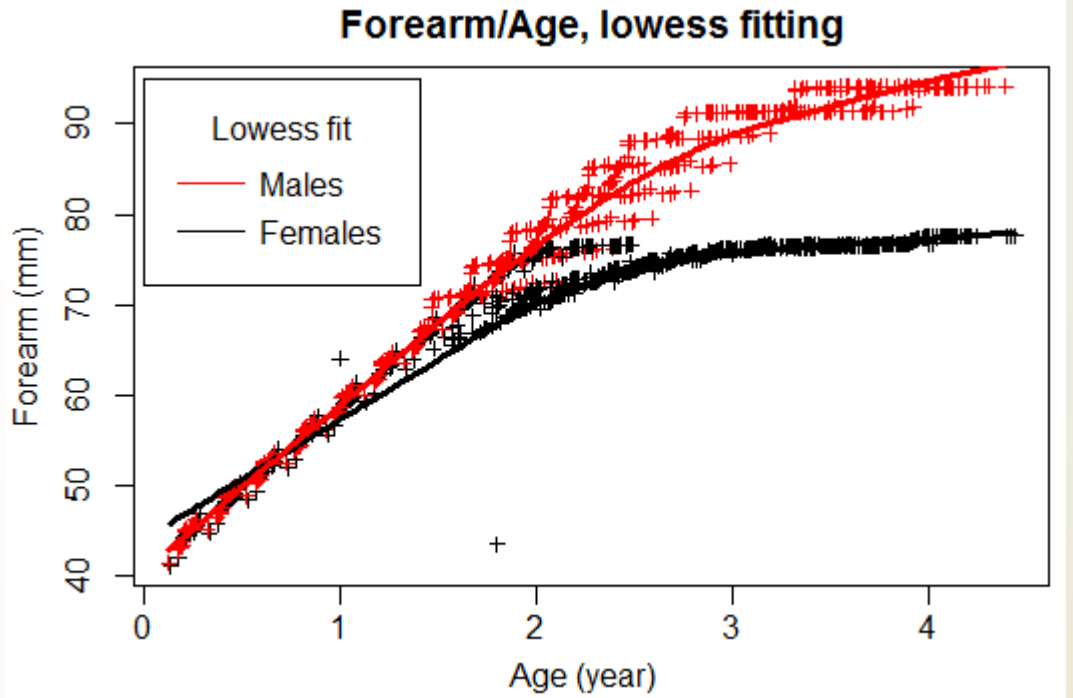
```
fitfem <- lowess(datfem$Forearm~datfem$Age)
fitmal <- lowess(datmal$Forearm~datmal$Age)

plot(e2m2$Forearm~e2m2$Age,
    main="Forearm/Age, lowess fitting",
    xlab="Age (year)",ylab="Forearm (mm)",
    type="p",pch=3,cex=0.7,
    col=e2m2$Sex)

lines(fitfem,col="black",lwd=3)
lines(fitmal,col="red",lwd=3)

legend(x=0, y=95,legend=c("Males","Females"),
      col=c("red","black"),title="Lowess fit",
      lty=1,x.intersp = .5,y.intersp = .8)
```





Visualizing Data (Present de data)

Data Plotting with (ggplo2)

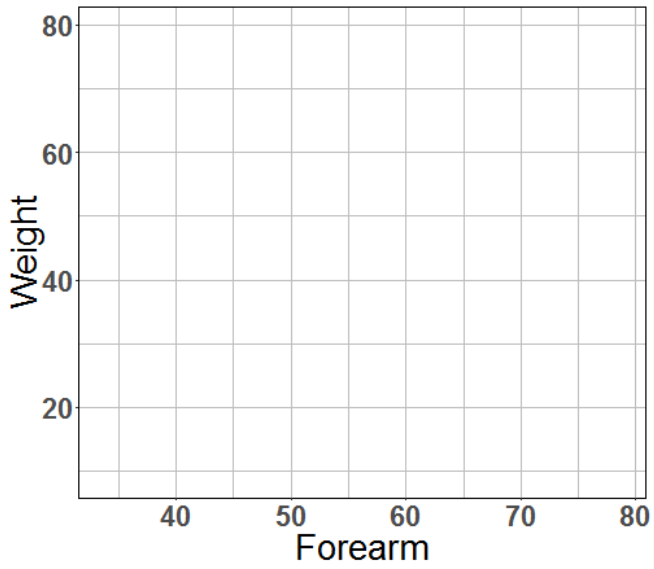
Base plot

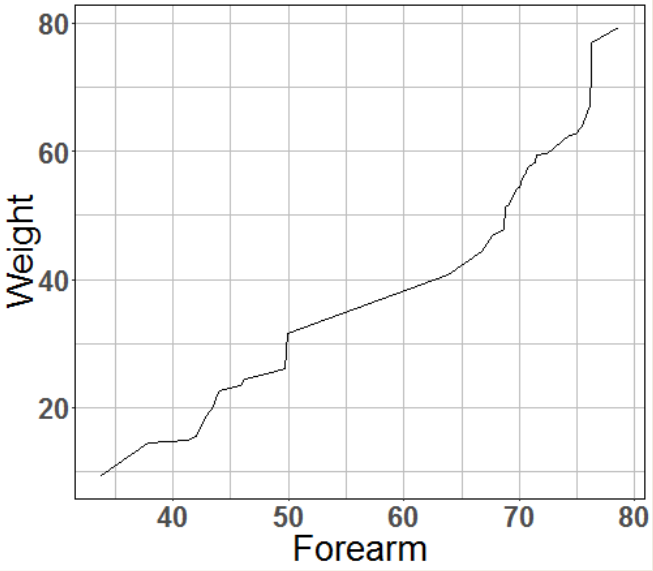
```
ggplot(data,aes(x,y))
```

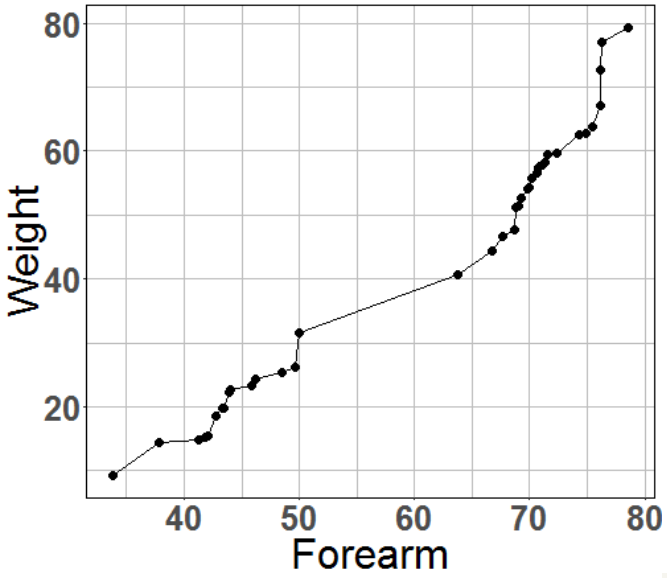
+ geom

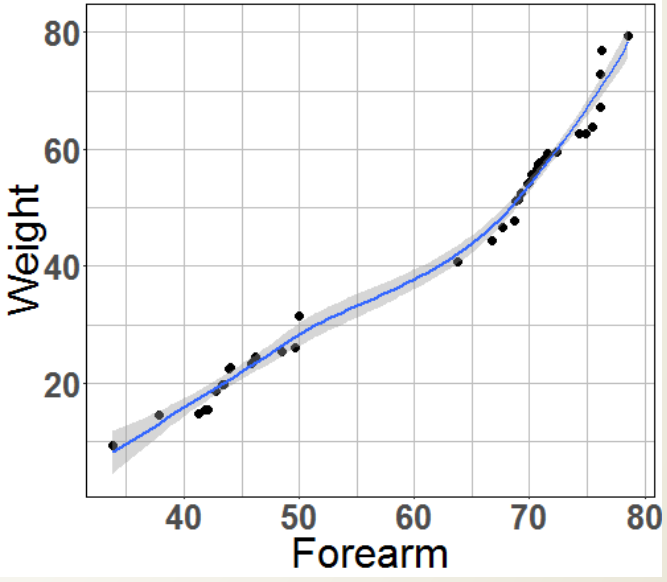
```
geom_line()
geom_point()
geom_boxplot()
geom_bar()
```

```
ggplot(fb_male,aes(Forearm,Weight))
  + geom_line()
  + geom_point()
  + geom_smooth()
```





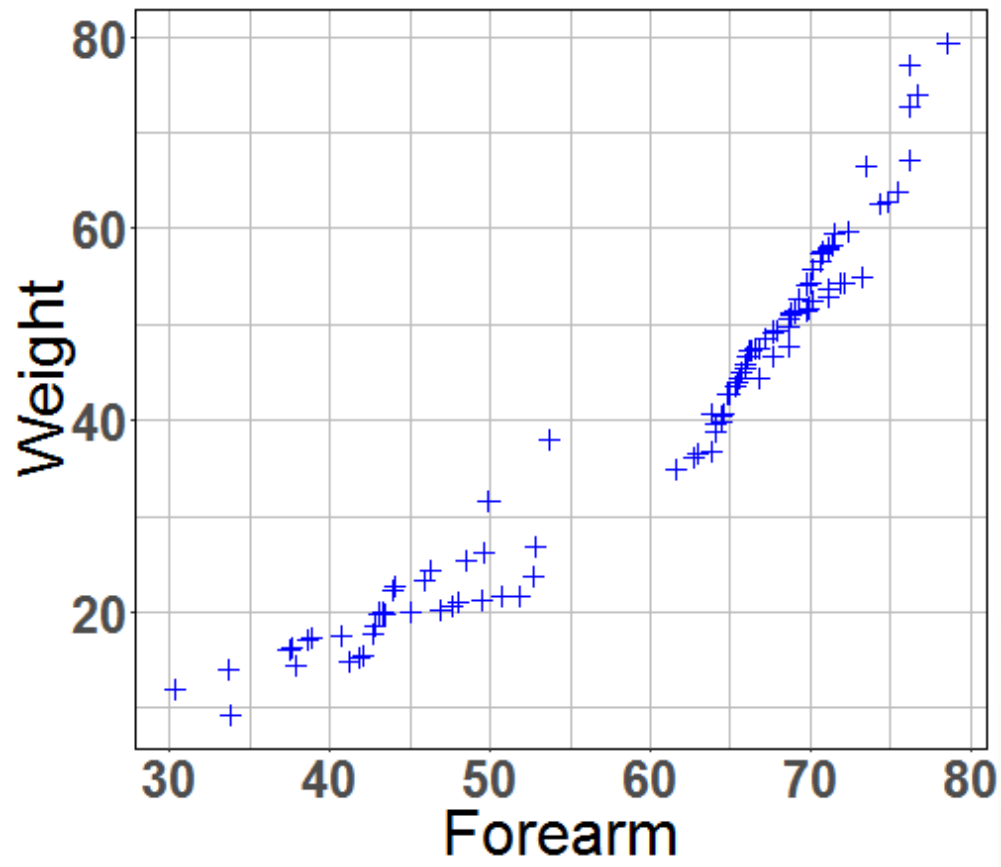




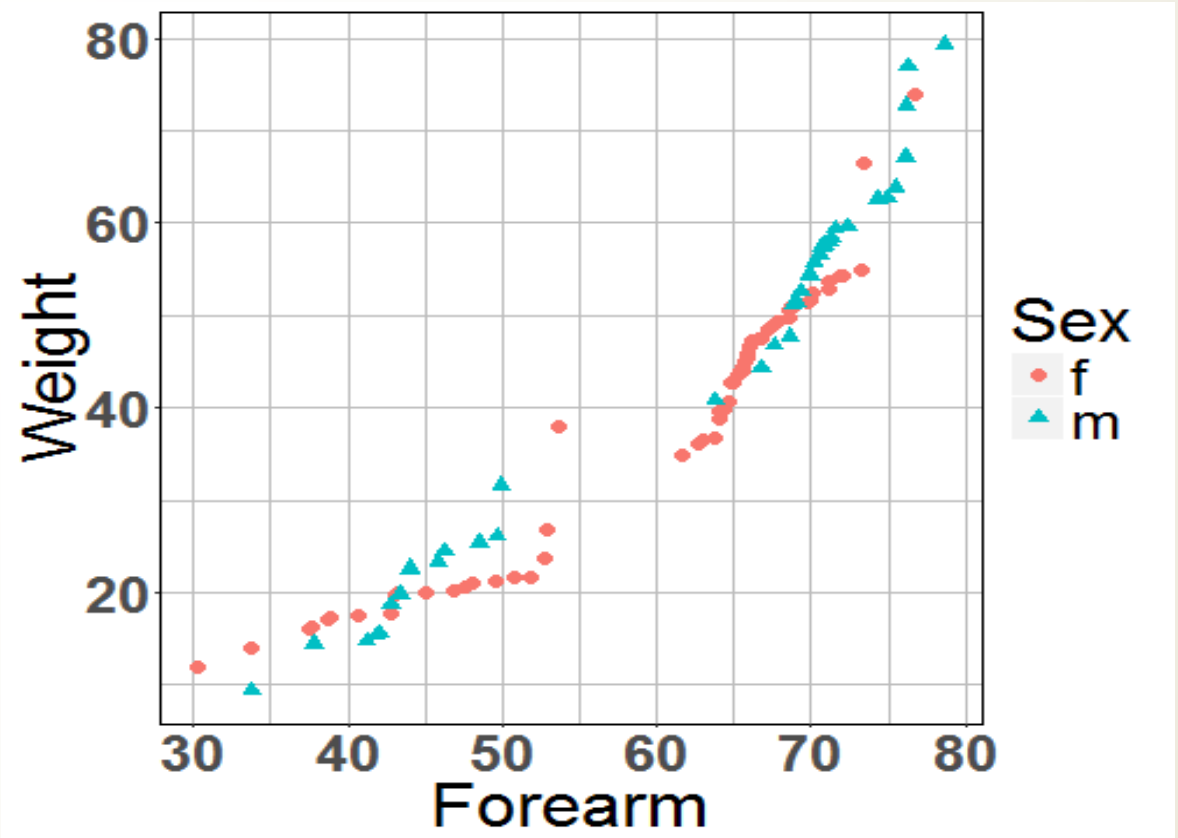
Visualizing Data (Present de data)

Mapping aesthetic vs fixed value

```
ggplot(e2m2_FB,aes(Forearm,Weight)) +
geom_point(color="blue", shape=3)
```




```
ggplot(e2m2_FB,aes(Forearm,Weight)) +
geom_point(aes(color=Sex, shape=Sex))
```



Visualizing Data (Present de data)

Polish the plot

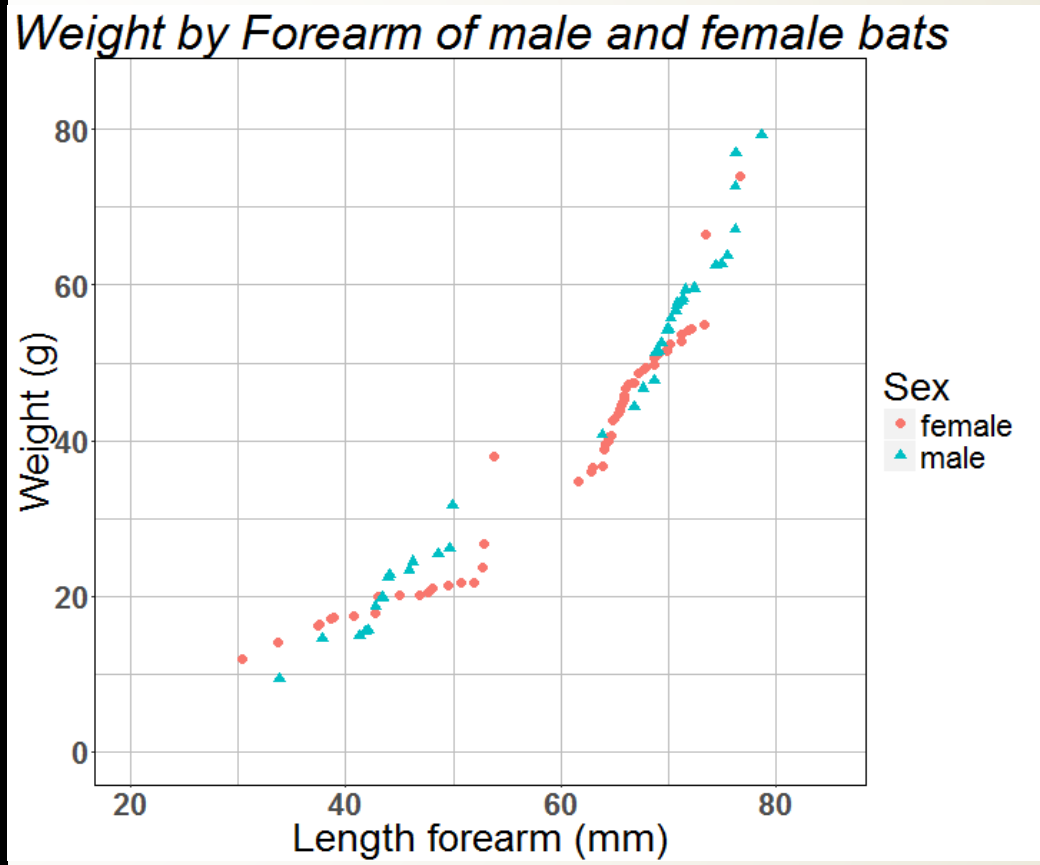
```
ggtitle("Weight by Forearm of male and female")+

scale_x_continuous(name="Length forearm (mm)",
                   limits=c(20,85))+

scale_y_continuous(name="Weight (g)",
                   limits=c(0,85))+

scale_color_discrete(name="Sex",
                     breaks=c("f","m"),
                     label=c("female","male"))+

scale_shape_discrete(name="Sex",
                     breaks=c("f","m"),
                     label=c("female","male"))
```



Conclusion

R software:

- Powerful data management
- Simple syntax
- Large graphic vocabularies
- Packages to fit needs