



# Biocomputing with R

Niklaus Zemp

June 2020

Genetic Diversity Centre (GDC)

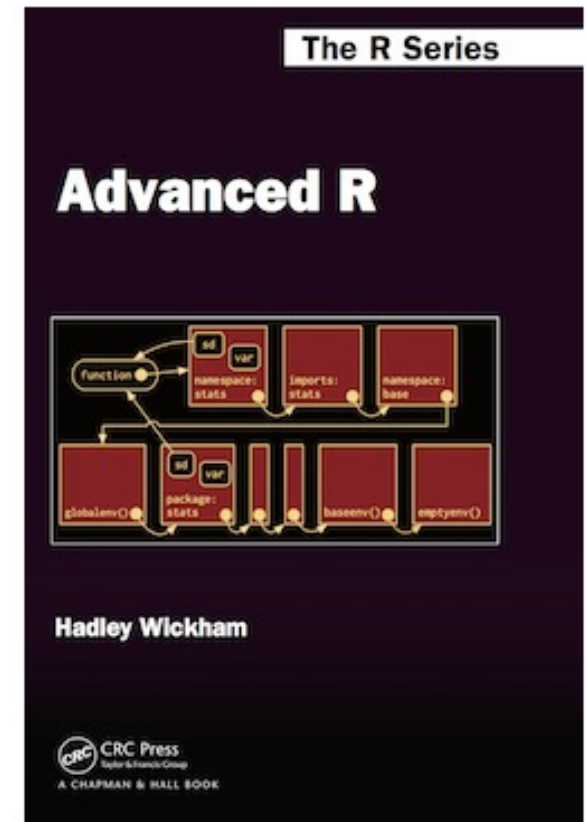
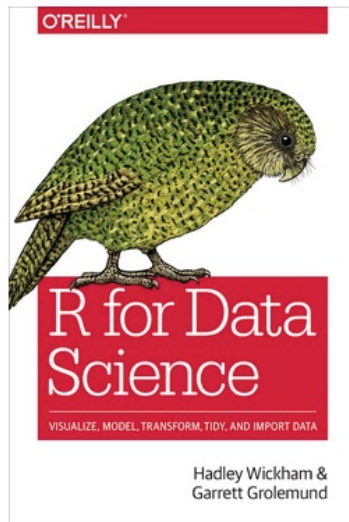
Bioinformatics

ETH Zurich



# Resources

Many many tutorials, forum, YouTube videos posts and books available





# Packages

## Available Packages

Currently, the CRAN package repository features 13884 available packages.

[Table of available packages, sorted by date of publication](#)

[Table of available packages, sorted by name](#)

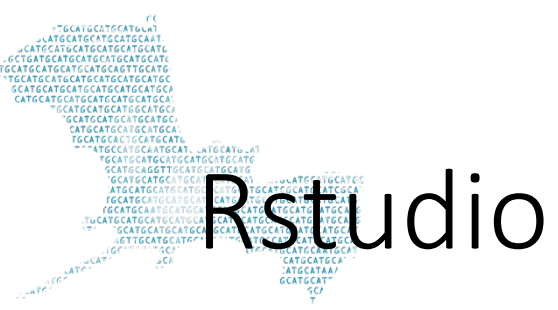



## Install »

- Discover [1649 software packages](#) available in *Bioconductor* release 3.8.

## Get started with *Bioconductor*

- [Install \*Bioconductor\*](#)
- [Get support](#)
- [Latest newsletter](#)
- [Follow us on twitter](#)
- [Install R](#)





# Import tables

```

dat <- read.csv("http://gdc-web.ethz.ch/gdc-analysis-course/2019/data/Students19.txt", header = T, sep="\t")

dat <- read_tsv("http://gdc-web.ethz.ch/gdc-analysis-course/2019/data/Students19.txt")
  
```



data.table::fread



# R Objects

- Data frames
- Lists
- Vectors
- Matrixes

<code>as.logical</code>	TRUE, FALSE, TRUE	Boolean values (TRUE or FALSE).
<code>as.numeric</code>	1, 0, 1	Integers or floating point numbers.
<code>as.character</code>	'1', '0', '1'	Character strings. Generally preferred to factors.
<code>as.factor</code>	'1', '0', '1', levels: '1', '0'	Character strings with preset levels. Needed for some statistical models.

```
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 ...
```

# R Objects



tibble

```
as_tibble(iris)
```

```
## # A tibble: 150 x 5
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
##   <dbl>         <dbl>         <dbl>         <dbl> <fct>
## 1         5.1         3.5           1.4           0.2 setosa
## 2         4.9         3             1.4           0.2 setosa
## 3         4.7         3.2           1.3           0.2 setosa
## 4         4.6         3.1           1.5           0.2 setosa
## 5         5           3.6           1.4           0.2 setosa
## 6         5.4         3.9           1.7           0.4 setosa
## 7         4.6         3.4           1.4           0.3 setosa
## 8         5           3.4           1.5           0.2 setosa
## 9         4.4         2.9           1.4           0.2 setosa
## 10        4.9         3.1           1.5           0.1 setosa
## # ... with 140 more rows
```



# Data Manipulation

```
iris.df <- data.frame(iris$Sepal.Length, iris$Sepal.Width, iris$Species)
```

## Replace patterns

```
iris.df$Species2 <- gsub("setosa", "Setosa", iris.df$iris.Species)
```

## subset

```
iris.df.sub <- subset(iris.df, iris.df$iris.Species == "setosa")
```

## order table

```
iris.df.orderd <- iris.df[order(iris.df$iris.Sepal.Length), ]
```



# Data Manipulation

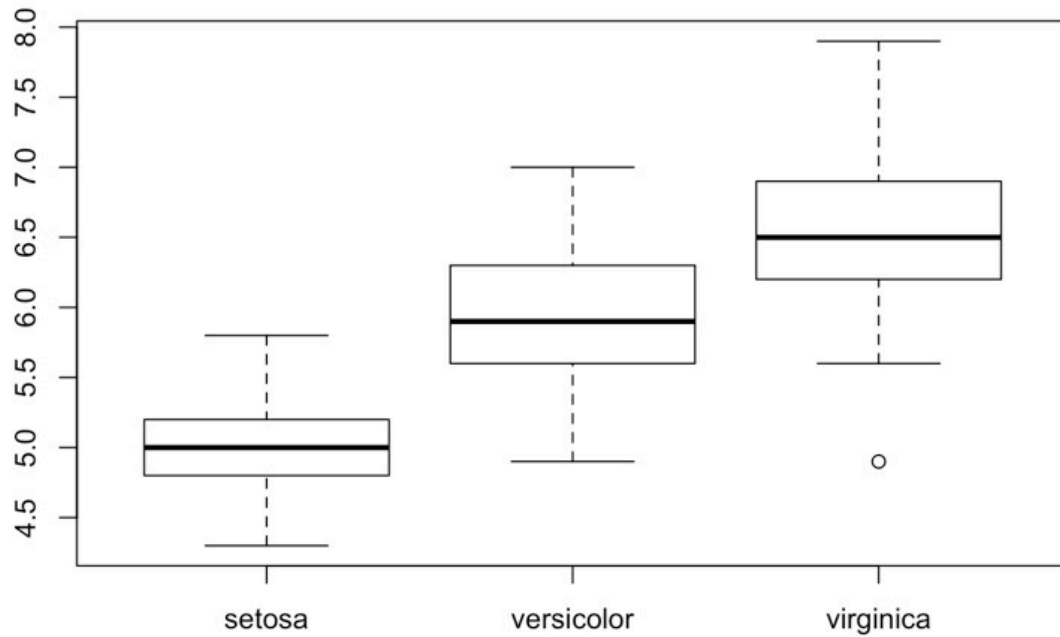
```

iris.tb <- as_tibble(iris) %>%
  select(Species, Sepal.Length, Sepal.Width, Petal.Length) %>%
  dplyr::filter(Species != "setosa") %>%
  mutate(Species2 = gsub("versicolor", "Versicolor", Species)) %>%
  arrange(., Sepal.Length)
  
```

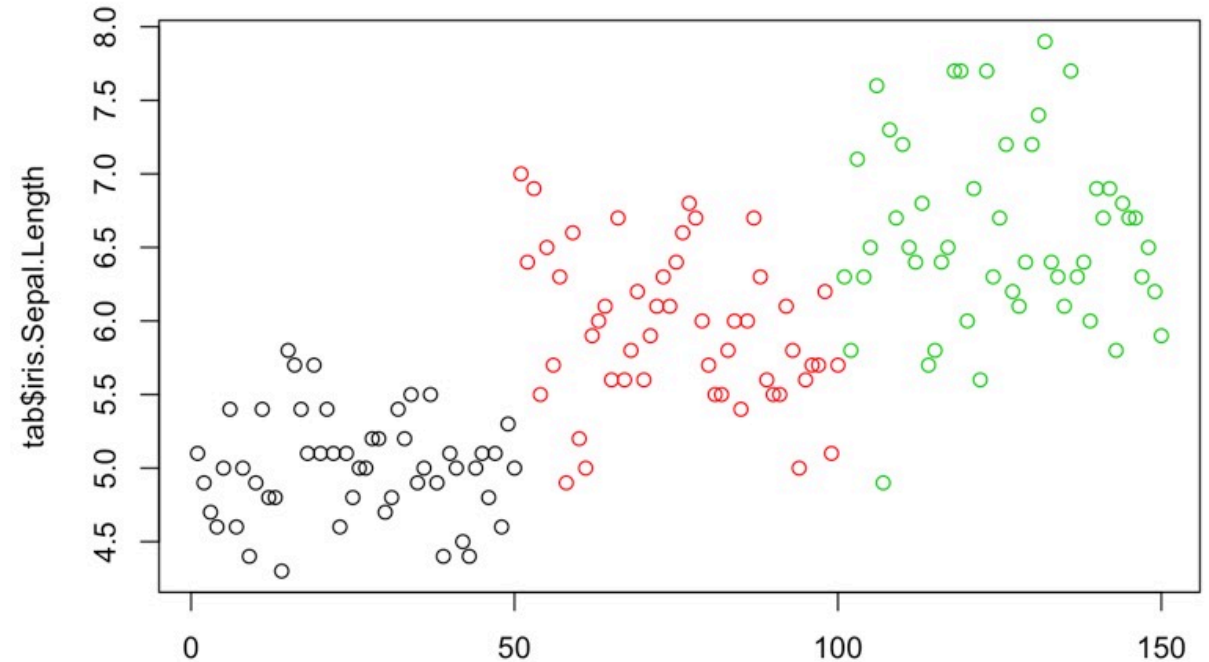


# Data visualization

```
boxplot(tab$iris.Sepal.Length~tab$iris.Species)
```



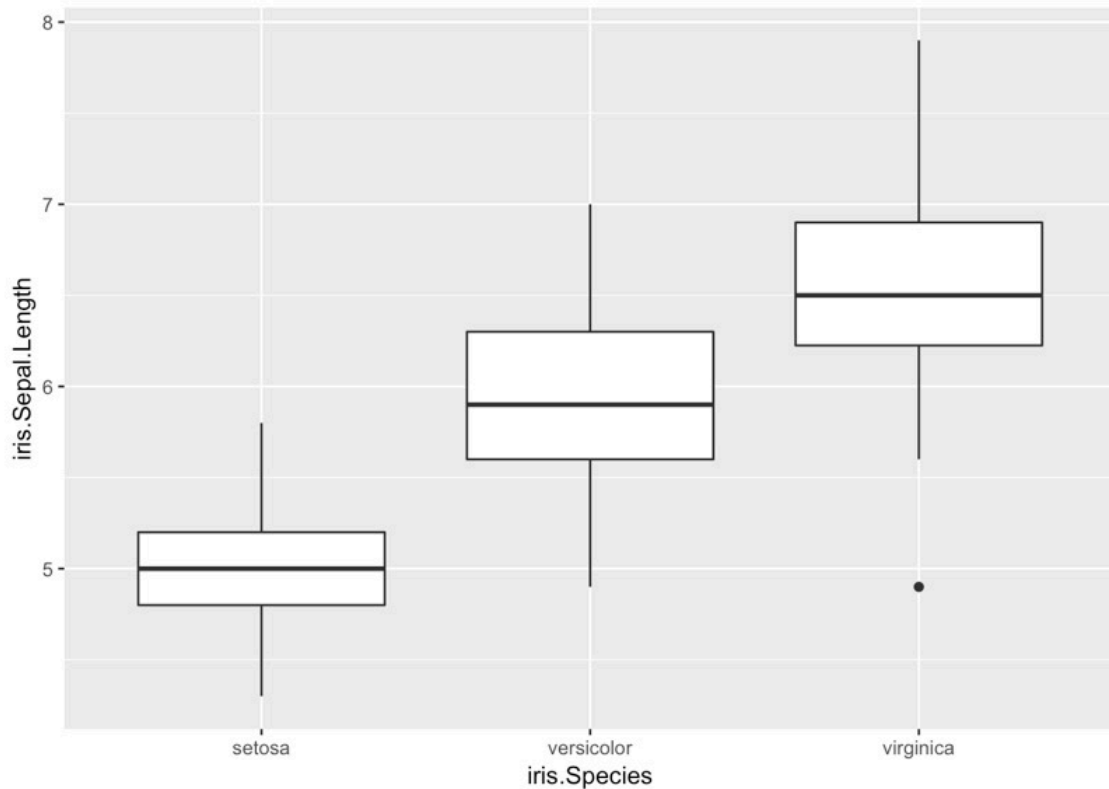
```
plot(tab$iris.Sepal.Length, col=tab$iris.Species)
```



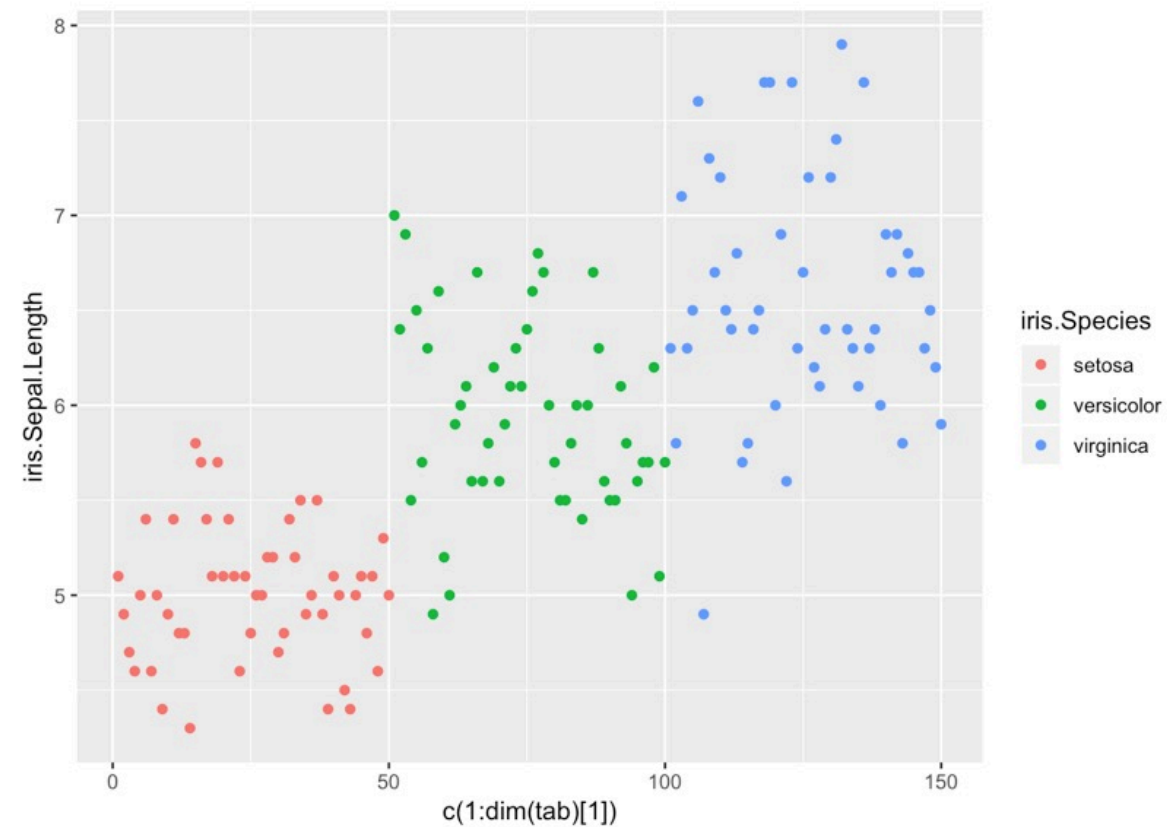
# Visualization-ggplot2



```
library(ggplot2)
ggplot(tab, aes(x=iris.Species,y=iris.Sepal.Length))+
  geom_boxplot()
```



```
ggplot(tab, aes(x=c(1:dim(tab)[1]),y=iris.Sepal.Length, colour=iris.Species))+
  geom_point()
```



## Distribution



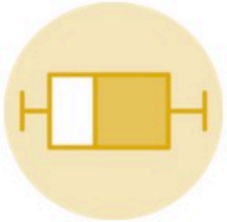
Violin



Density



Histogram



Boxplot



Ridgeline / Joyplot

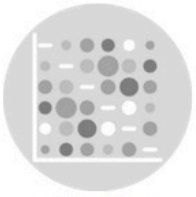
## Correlation



Scatter



Heatmap



Correlogram



Bubble



Connected Scatter



Density 2D

## Rankings



Barplot



Spider / Radar



Wordcloud



Parallel



Lollipop / Stem



Circular Barplot

## Part of a whole



Treemap



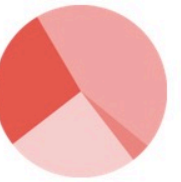
Dendrogram



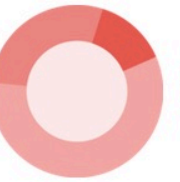
Venn Diagram



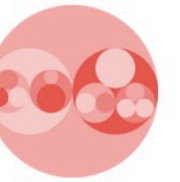
Stacked Bar



Pie Chart



Doughnut



Circular Packing

## Evolution



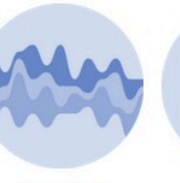
Stacked Area



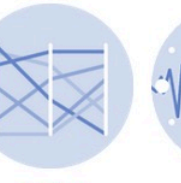
Line



Area



Streamgraph



Parallel



Time series

## Maps



Background Map



Connection



Choropleth



Bubble



Hexbin Map

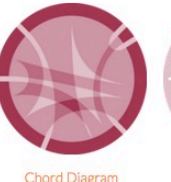


Cartogram

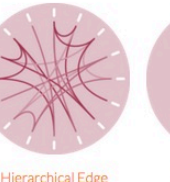
## Flow



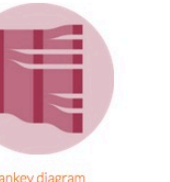
Network



Chord Diagram



Hierarchical Edge  
Bundling



Sankey diagram

## Other



Interactive



Colour



Cheat sheet



3D

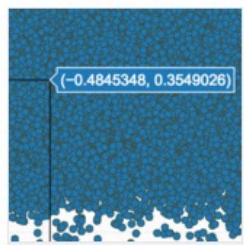


Animation

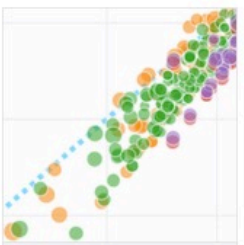


Bad Charts

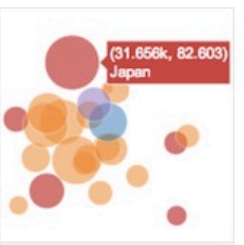
<https://www.r-graph-gallery.com/>



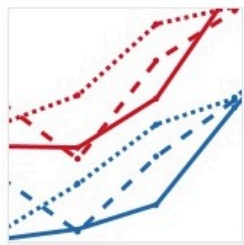
WebGL vs SVG in R



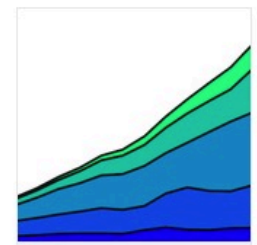
Scatter and Line Plots



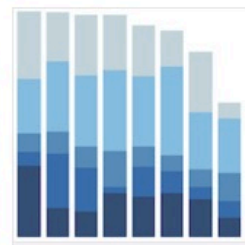
Bubble Charts



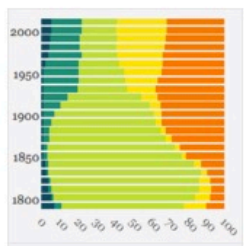
Line Plots



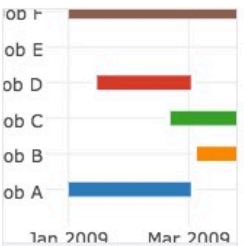
Filled Area Plots



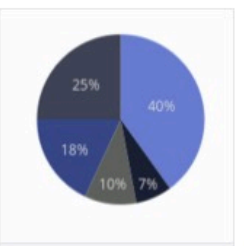
Bar Charts



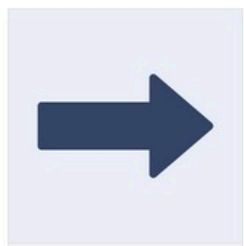
Horizontal Bar Charts



Gantt Charts



Pie Charts



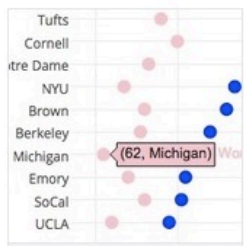
Graphing Multiple Chart Types



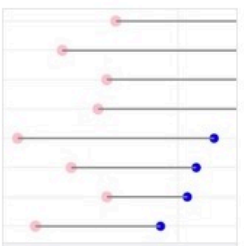
Sunburst Charts



Tables



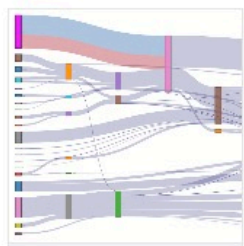
Dot Plots



Dumbbell



Gauge Charts



Sankey

You've collected the data, agonized over the right model and now you've GOT to come up with a management plan....

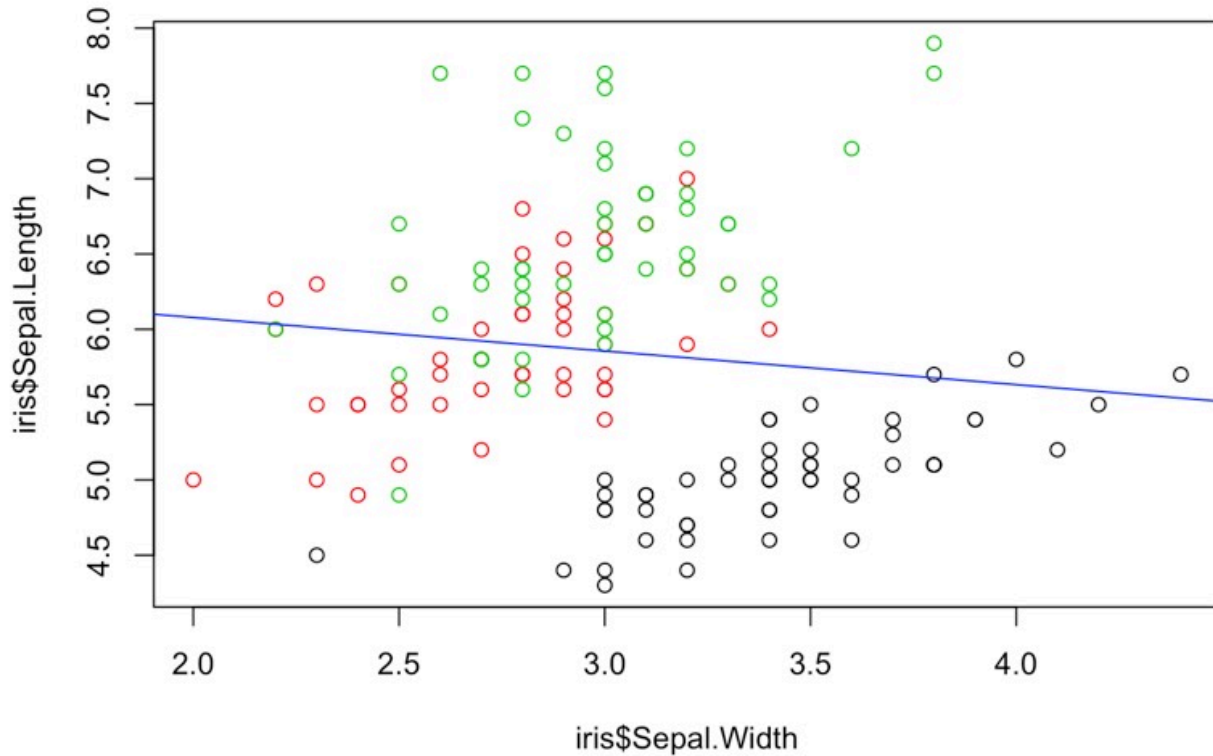
# Statistics using R-Linear models

```

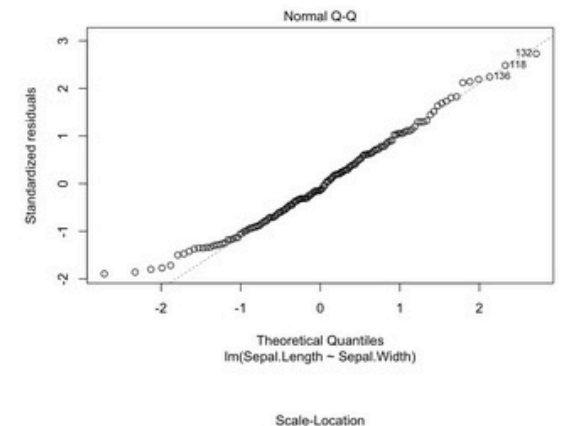
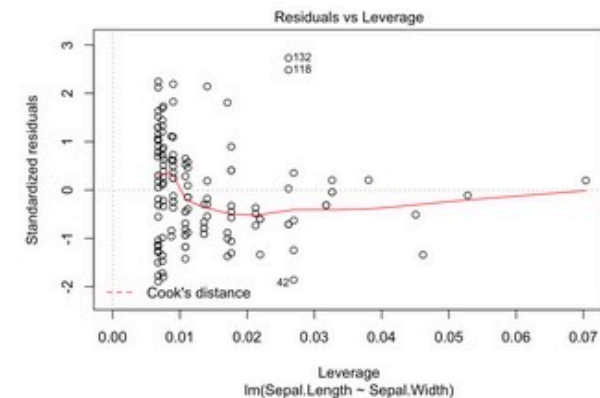
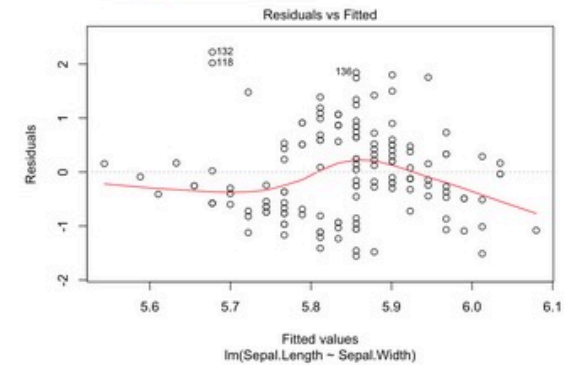
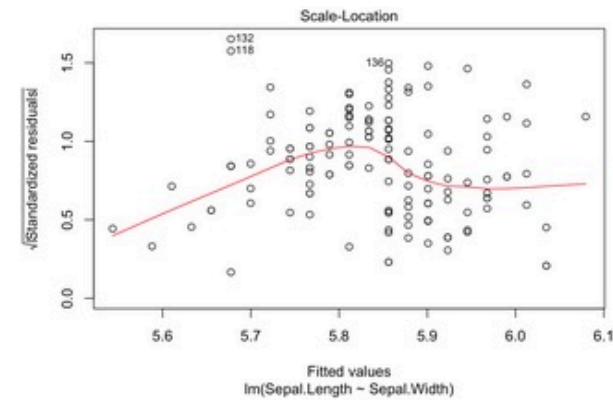
ml<-lm(Sepal.Length~Sepal.Width,data=iris)

plot(iris$Sepal.Length~iris$Sepal.Width, col=iris$Species)
abline(ml$coefficients, col="blue")

```

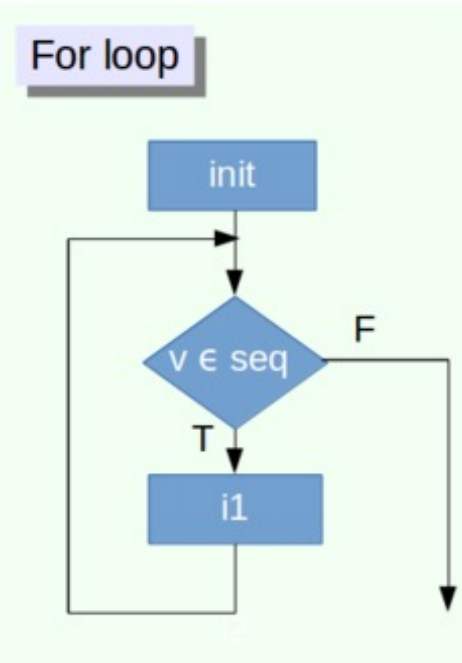
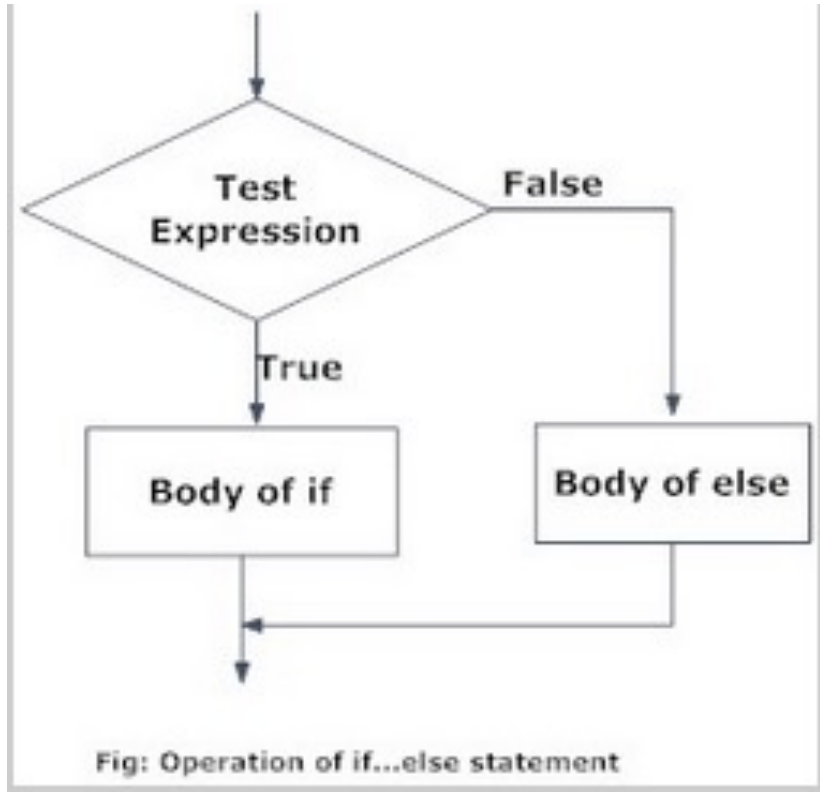


```
plot(ml)
```





# Programming in R



**Hands-On Programming with R**  
<https://rstudio-education.github.io/hopr/>



# Customized functions

Myfunction <- function(variables) {

Function1

Function2

value/plot

}

```

ploting <- function(n_samples) {
  dat <- rnorm(n_samples, 100, 5)
  plot(dat)
}

```

```
ploting(10000)
```

## Customized packages





# Take home message

- Fastly developing
- Many resources for biologists
- Also suitable for programming

