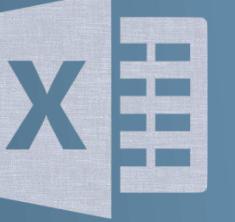


FROM  EXCEL TO 



WHO ARE WE?

1. Center for Health Data Science (HeaDS) - <https://heads.ku.dk/>

SUND Center, which includes a KU data lab

- Courses & Workshops, Seminars, etc.
- Health DS Consultations
- Commissioned Research
- Matchmaking
- Commissioned Supervision



Thilde Terkelsen ¹



Diana Andrejeva ¹



Tugce Karaderi ¹



Henrike Zschach ¹



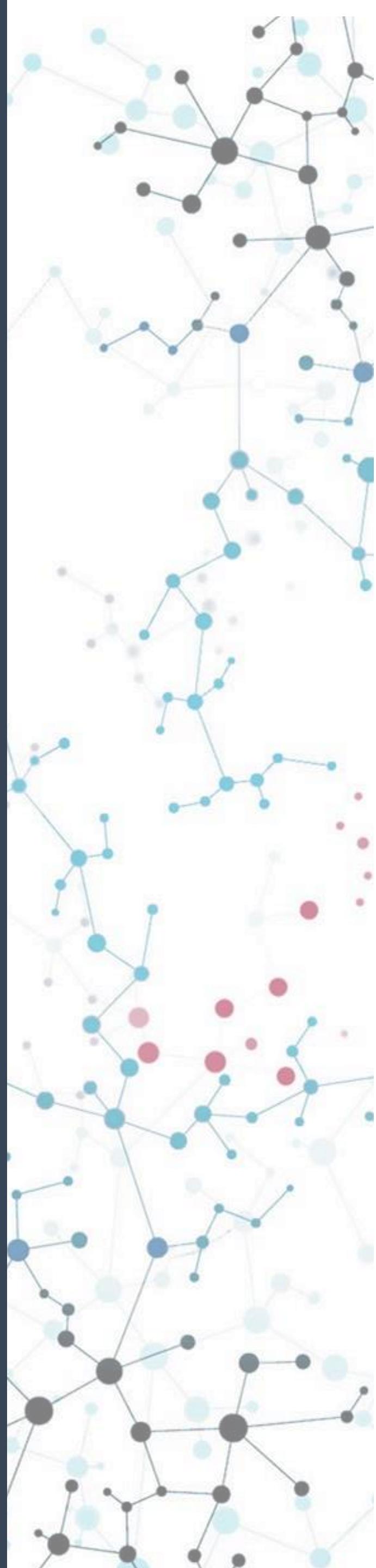
Helene Wegener ¹



Adrija Kalvisa ²

2. ReNEW NNF Center for Stem Cell Medicine

3. Data Science Laboratory (DSL) - <https://datalab.science.ku.dk/>



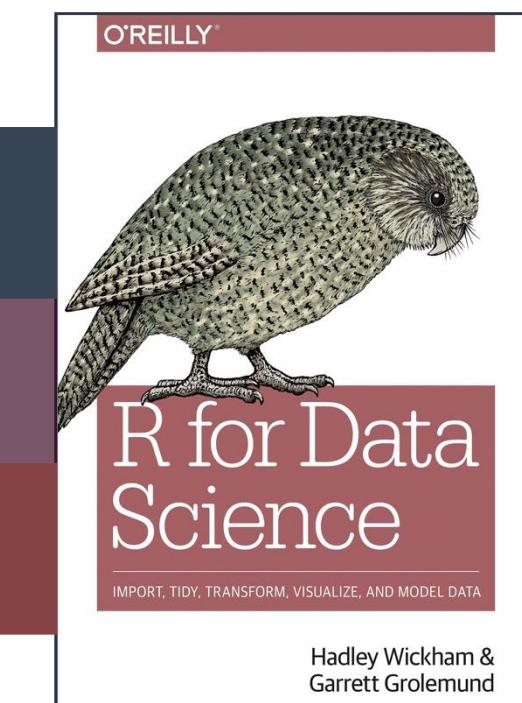
THE PRACTICALS



Two days: 9.00-16.00. There will be coffee breaks, we promise ☕

“R for Data Science” - a generally useful book on R, also for this course

The course is build on hands-on presentations (.R, .qmd) & exercises



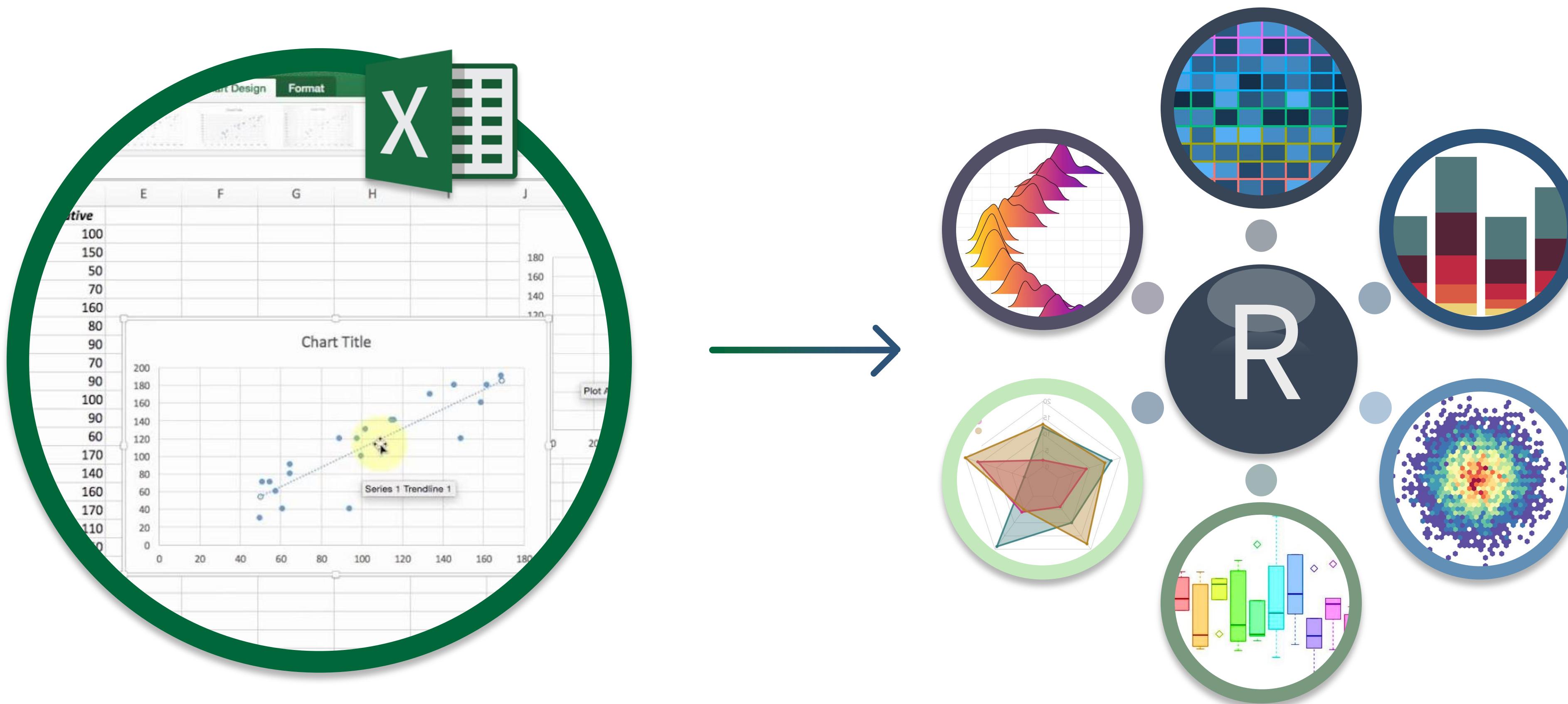
Download and install the newest version of R (<https://cran.r-project.org/>)

Download and install the newest version of R-studio (<https://posit.co/download/rstudio-desktop/>)

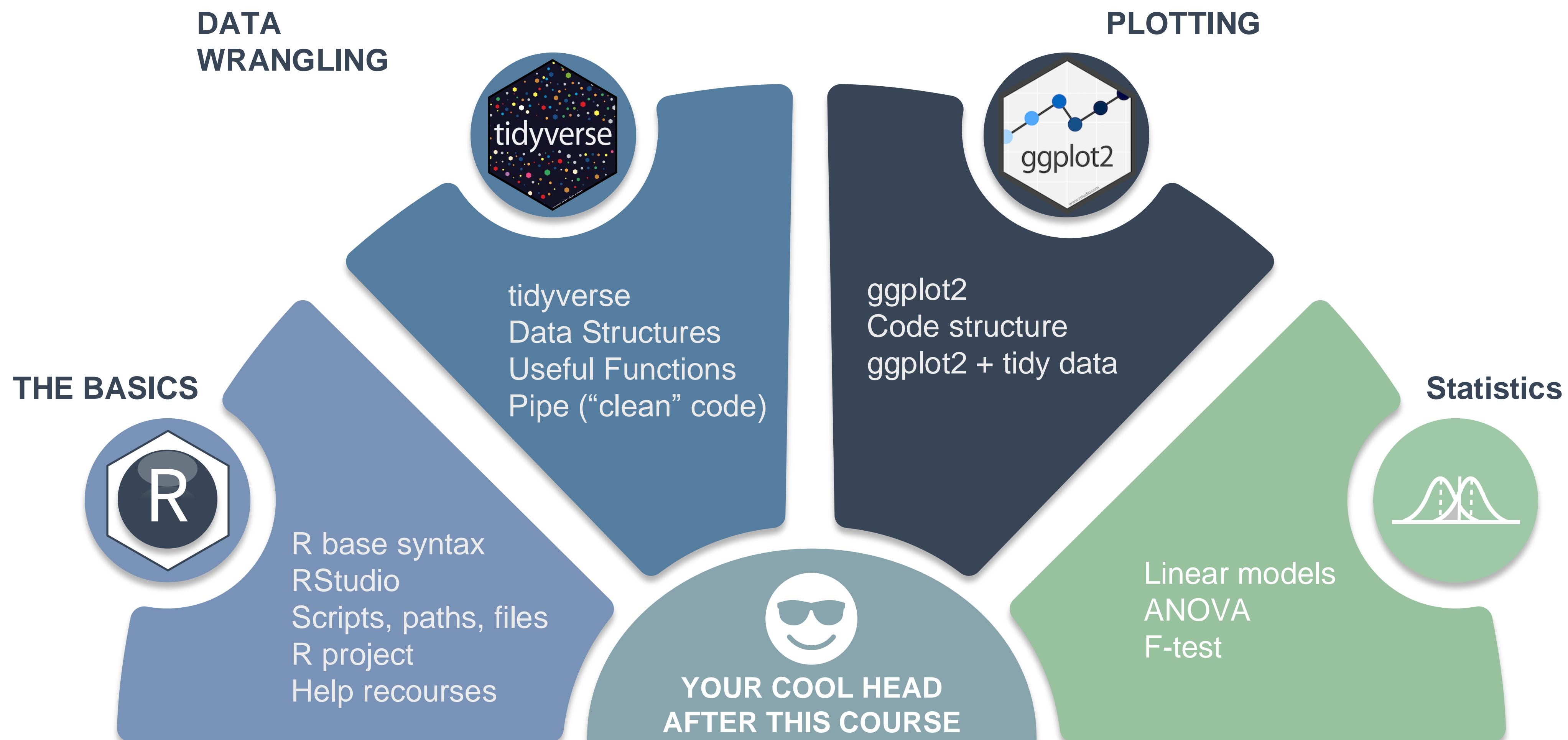
Go to course website: <https://center-for-health-data-science.github.io/FromExceltoR/>



WELCOME TO FROM EXCEL TO R



WHAT WILL YOU LEARN IN THIS COURSE?



PROGRAM

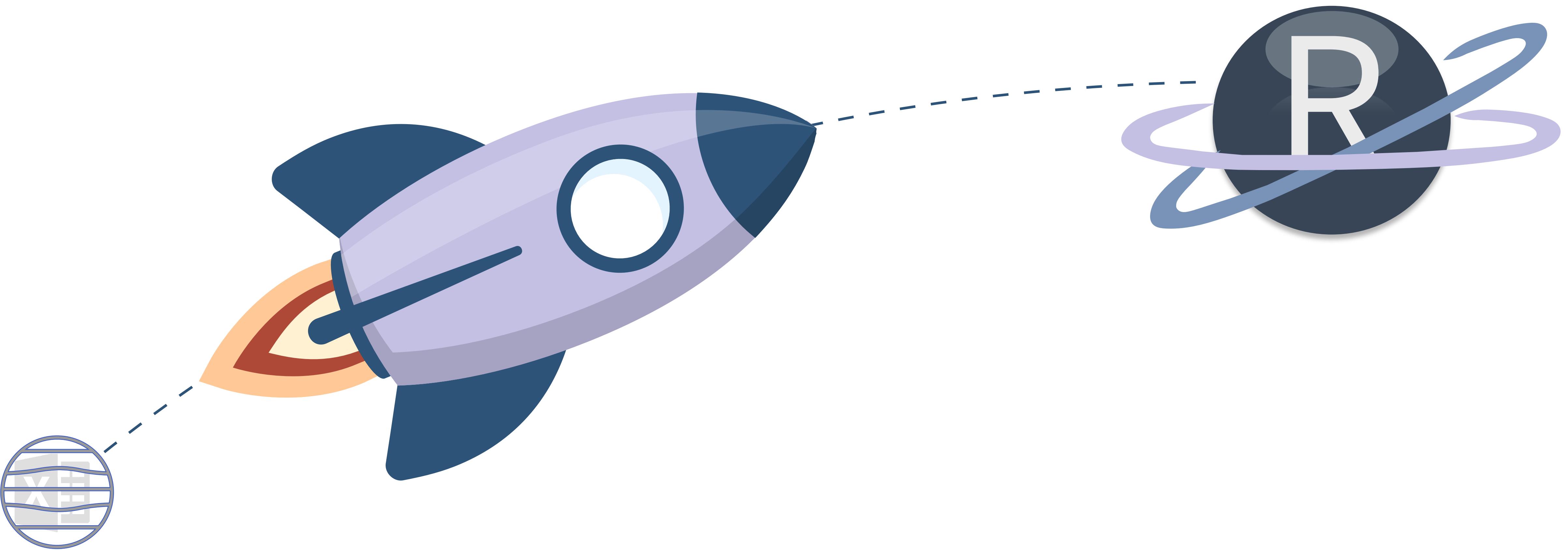
DAY 1

08:30 - Installation Issues + Coffee
09:00 - Introduction to R Basics
10:00 - Rstudio Exercise
10:45 - Break
11:00 - Tidyverse
12:00 - Lunch
13:00 - Tidyverse Exercise
14:30 - Break
14:45 - ggplot2
15:15 - ggplot2 Exercise
16:00 - Q&A - See you tomorrow

DAY 2

08:30 - Coffee + Optional Q&A
09:00 - Applied Statistics
10:00 - Applied Statistics Exercise
10:45 - Break
11:00 - Applied Statistics Exercise (cont.)
12:00 - Lunch
13:00 - Basic Data Analysis Exercise
14:30 - Break
14:45 - Basic Data Analysis Exercise (cont.)
15:45 - Cool things in R + Course Evaluation
16:00 - Bye Bye Bye!

— FROM EXCEL TO R
LET'S GET STARTED



R & FRIENDS



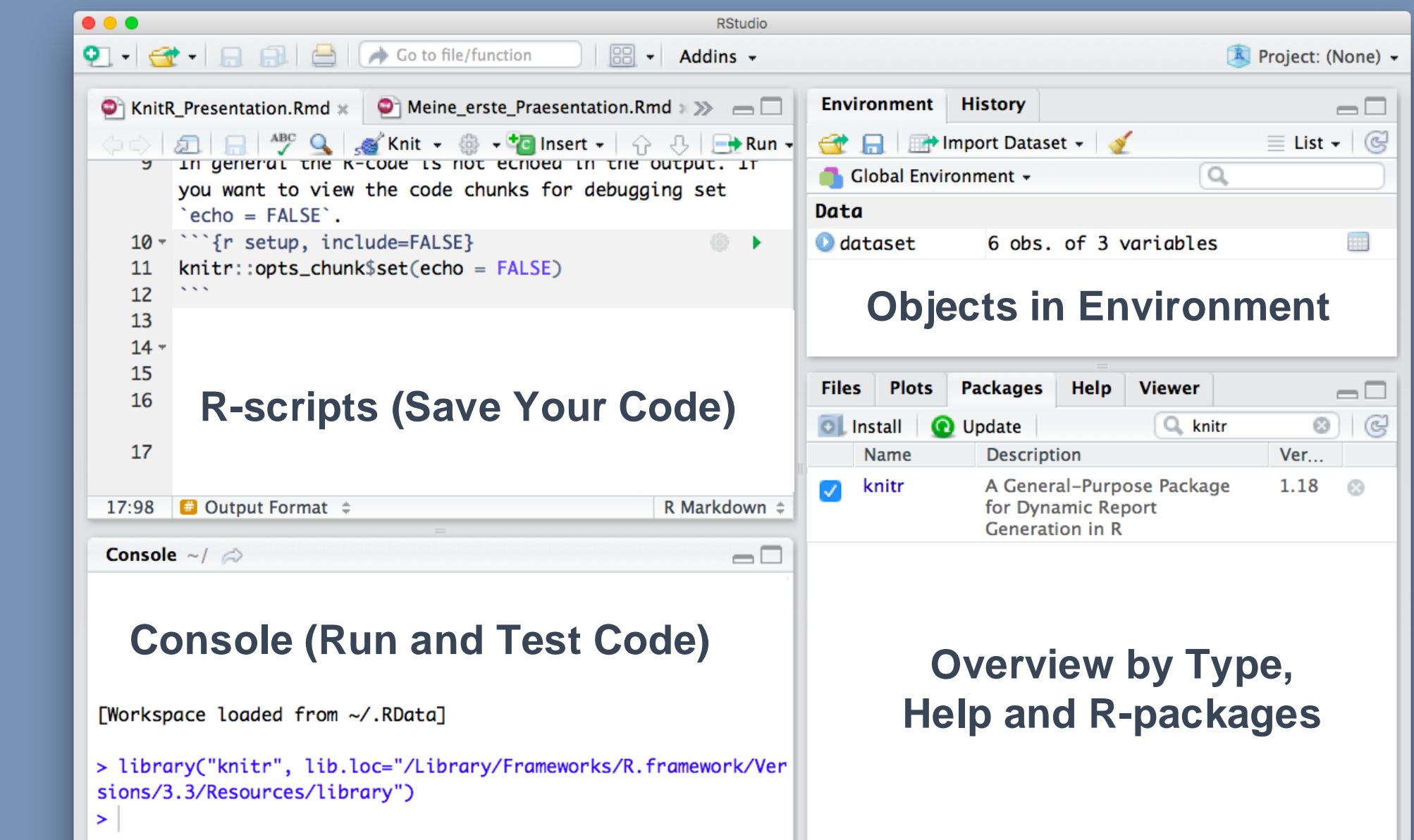
Scripting / Programming Language



Reports (html, pdf,
latex)

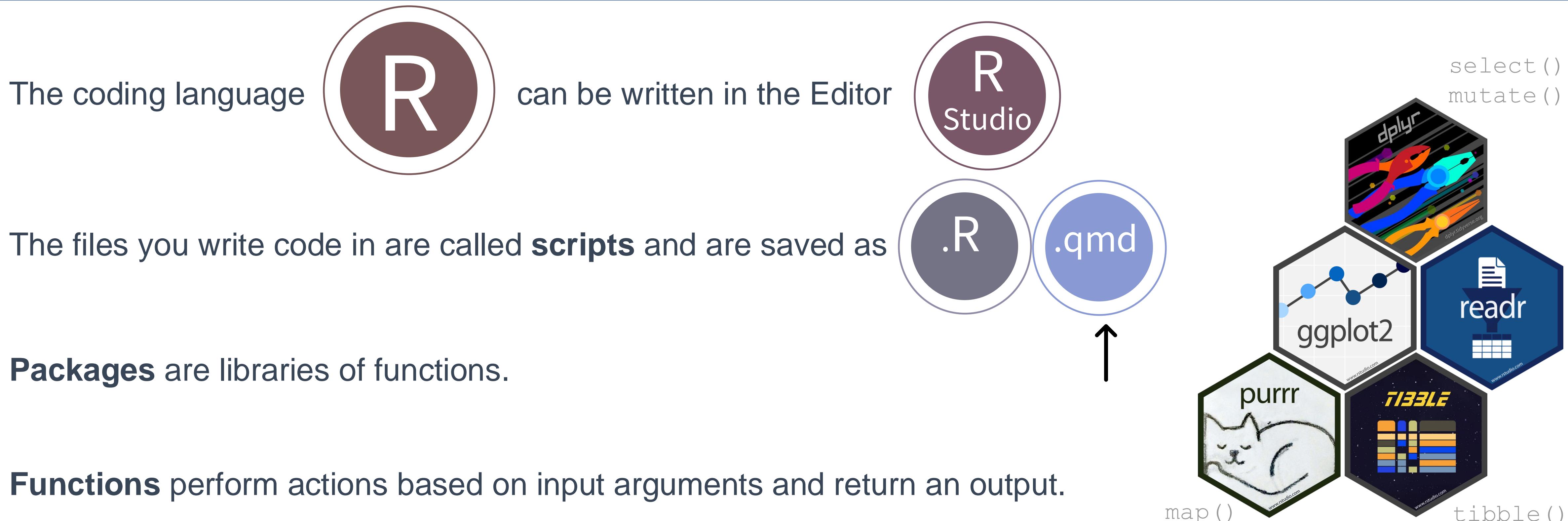


R Studio



R Code Interpreter and Editor

THE ANATOMY OF R



ONLINE RESOURCES FOR R

<https://www.r-project.org/>



GET STARTED

<https://rseek.org/>

https://rstudio.com/resources/c_heatsheets/

<http://www.cookbook-r.com/>

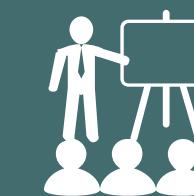
https://www.statmethods.net/r_tutorial/index.html



GRAPHICS

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

http://r-statistics.co/Top50_Ggplot2-Visualizations-MasterList-R-Code.html



BOOKS & COURSES

<https://www.r-bloggers.com/best-books-to-learn-r-programming/>

<https://www.datacamp.com/>

<https://www.codecademy.com/>

<https://www.coursera.org/>



OTHER RESOURCES

<https://github.com/trending/r>

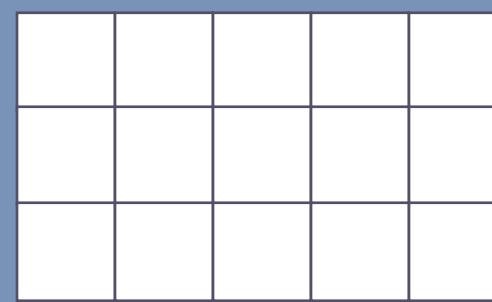
<https://blog.revolutionanalytic s.com/>

<https://stackoverflow.com/qu estions/tagged/r>



R DATA TYPES & STRUCTURES

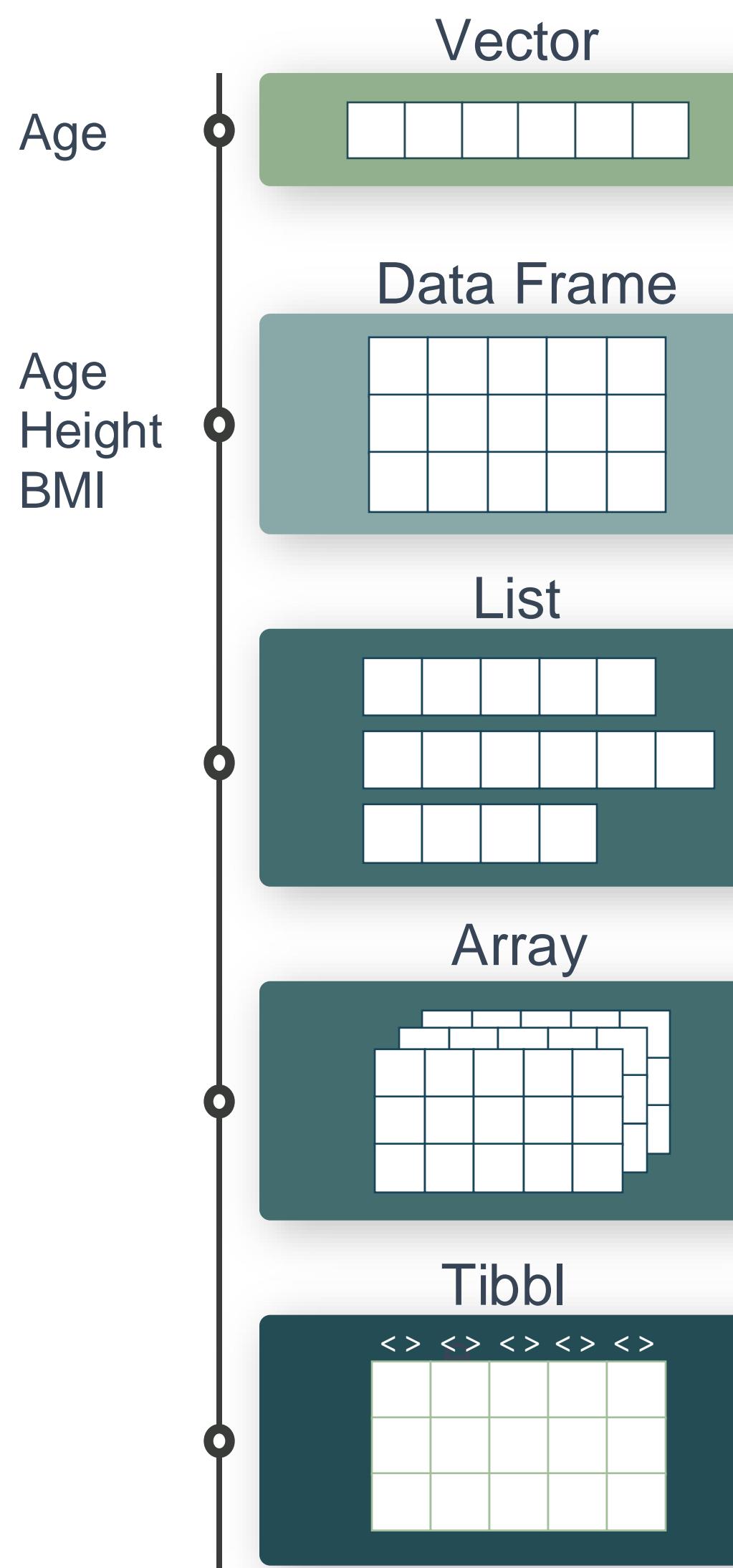
VARIABLES



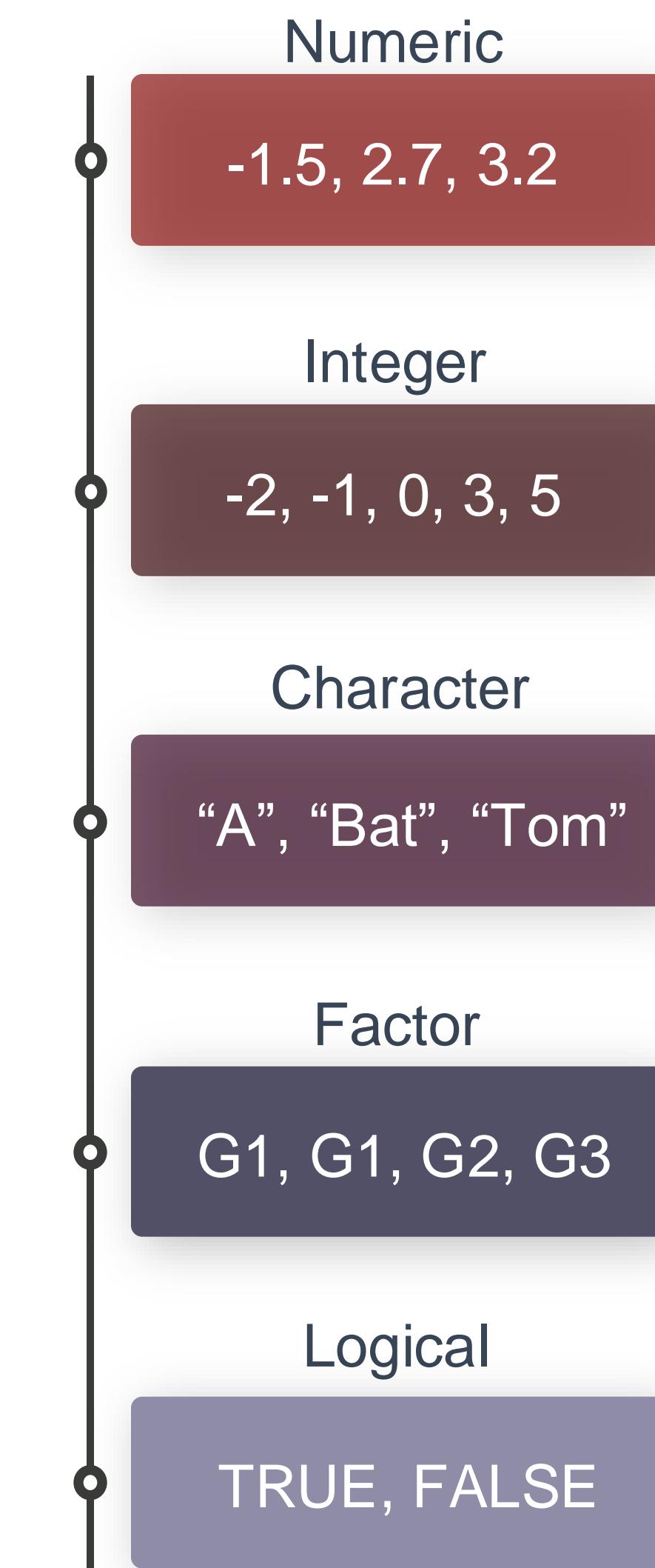
OBSERVATIONS

— FROM EXCEL TO R

DATA STRUCTURES

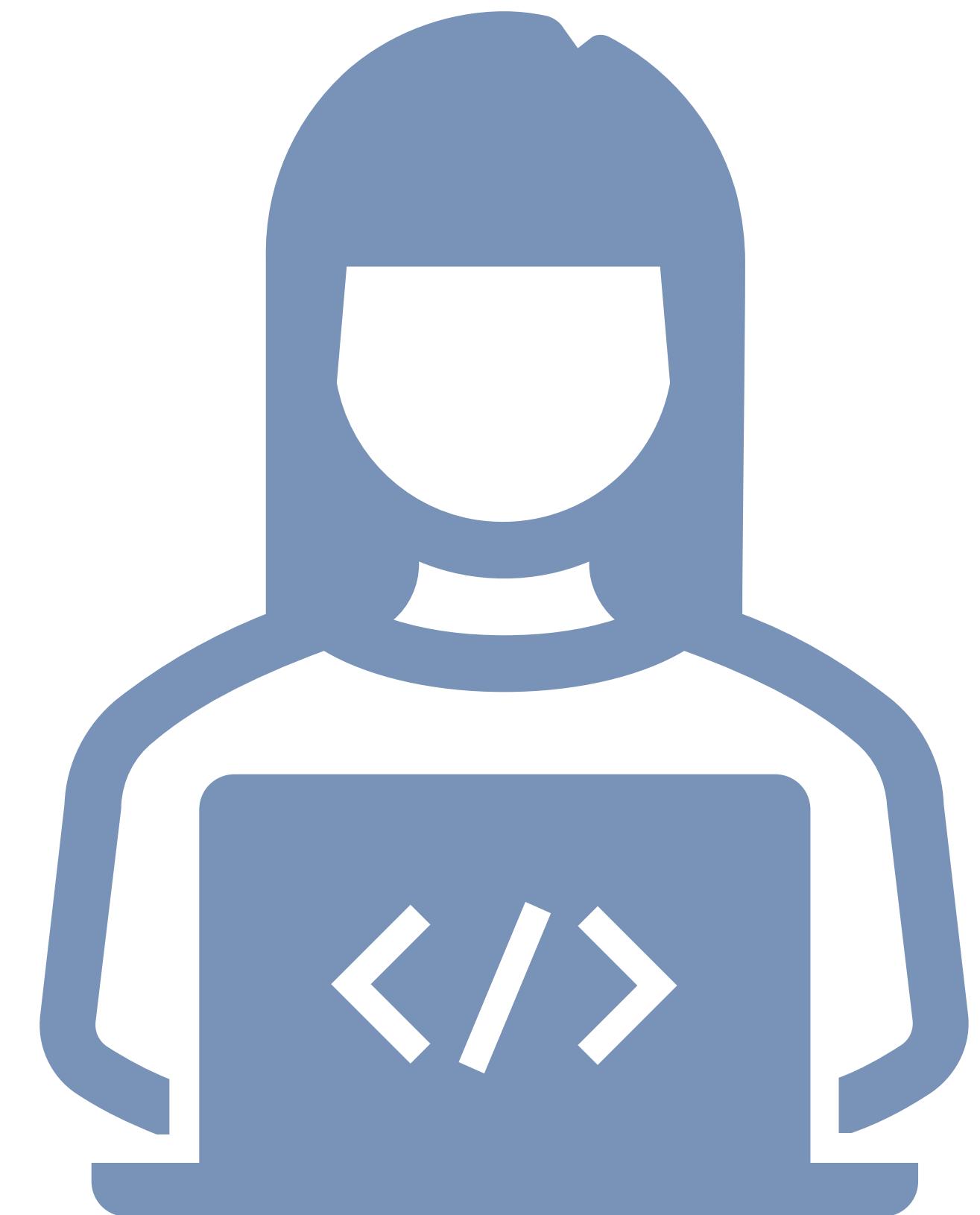


DATA TYPES



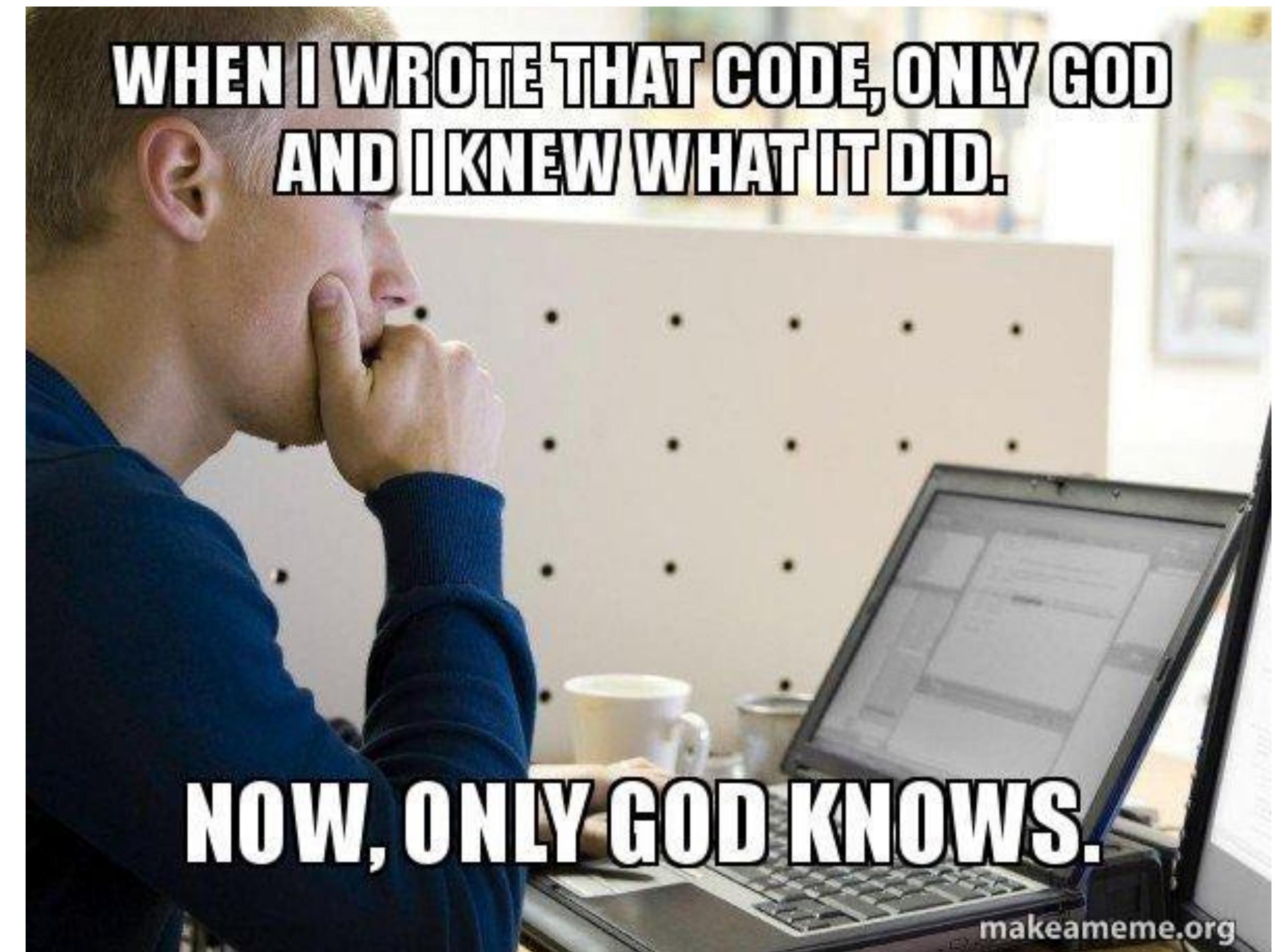
— FROM EXCEL TO R

LIVE CODING 1 – INTRO TO R



QUARTO STRUCTURE

- Separate sections with headers
- Do not overfill code chunks
- Think about when to use text and comments



INTRO CHEAT SHEET

Basics:

```
getwd() # location  
install.packages("package_name") # install packages  
library(package_name) # load packages
```

Overview:

```
head(df, n=10), tail(df, n=10) # first or last 10 rows  
unique(), table(), count() # unique vals, count vals
```

Type/Class:

```
class() # get data type/class  
is.numeric(x), is.character(), is.factor(), is.integer() # get data type  
as.numeric(x), as.character(), as.factor(), as.integer() # change data type
```

Summary statistics:

```
summary() # summary statistics  
mean(), median(), sd(), sum()  
n()
```

Read in data:

```
read.xlsx("name.xlsx") # library(readxl)  
read.delim("name.txt", sep ="\t")  
read.csv("name.csv", sep=";")
```

```
view() # view data as table  
df$col1 # extract column from dataframe  
nrow(df), ncol(df) # number of rows/columns
```

Plots:

```
plot(x)  
plot(x, y) or plot(col1, col2, df) # scatter  
hist(x) # histogram
```

GETTING STARTED

OVERVIEW

DATA TYPES

STATISTICS & BASE PLOTS

QUATO CHEAT SHEET

YAML parameter:

```
---
```

```
title: My Project Name
```

```
output:
```

```
    html_document (pdf_document, ...)
```

```
---
```

Code Chunk:



Source mode:

```
```{r}
```

```
some R code
```

```
```
```

Visual mode:

```
{r}
```

```
# some R code
```

Code Options:

```
{r echo = FALSE} # don't print code (default is TRUE)
```

```
{r eval = FALSE} # don't run code (default is TRUE)
```

```
{r error = FALSE} # don't display error message (default is TRUE)
```

(Can also be set for warning and message)

Figure Options:

```
fig.align (='left', 'right', 'center')
```

```
fig.cap (= 'my figure caption')
```

```
fig.height (= n), fig.width (= n)
```

GETTING STARTED

CHUNK OPTIONS

Source mode:

Header

Header size ranging from largest (one #)
to smallest (six #):
my.text, ## my.text, ### my.text, etc.

Text

italics

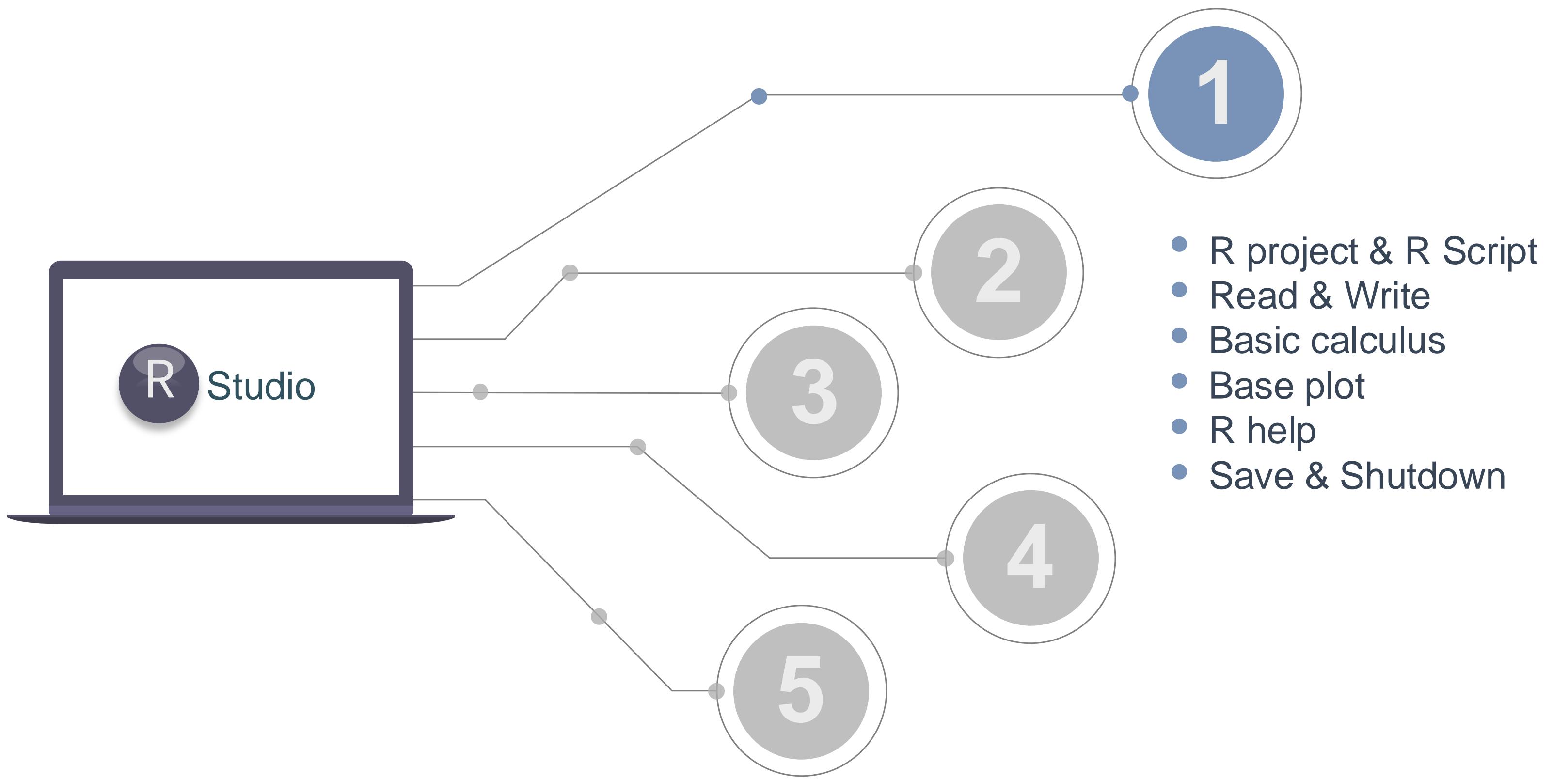
bold

`highlighted`

Lists

- * List item1 (filled dot)
 - + sub-item1 (open dot)
- 1. List item1 (numbered)
 - i) sub-item1 (roman)

TEXT



FUNDAMENTALS EXERCISE 1

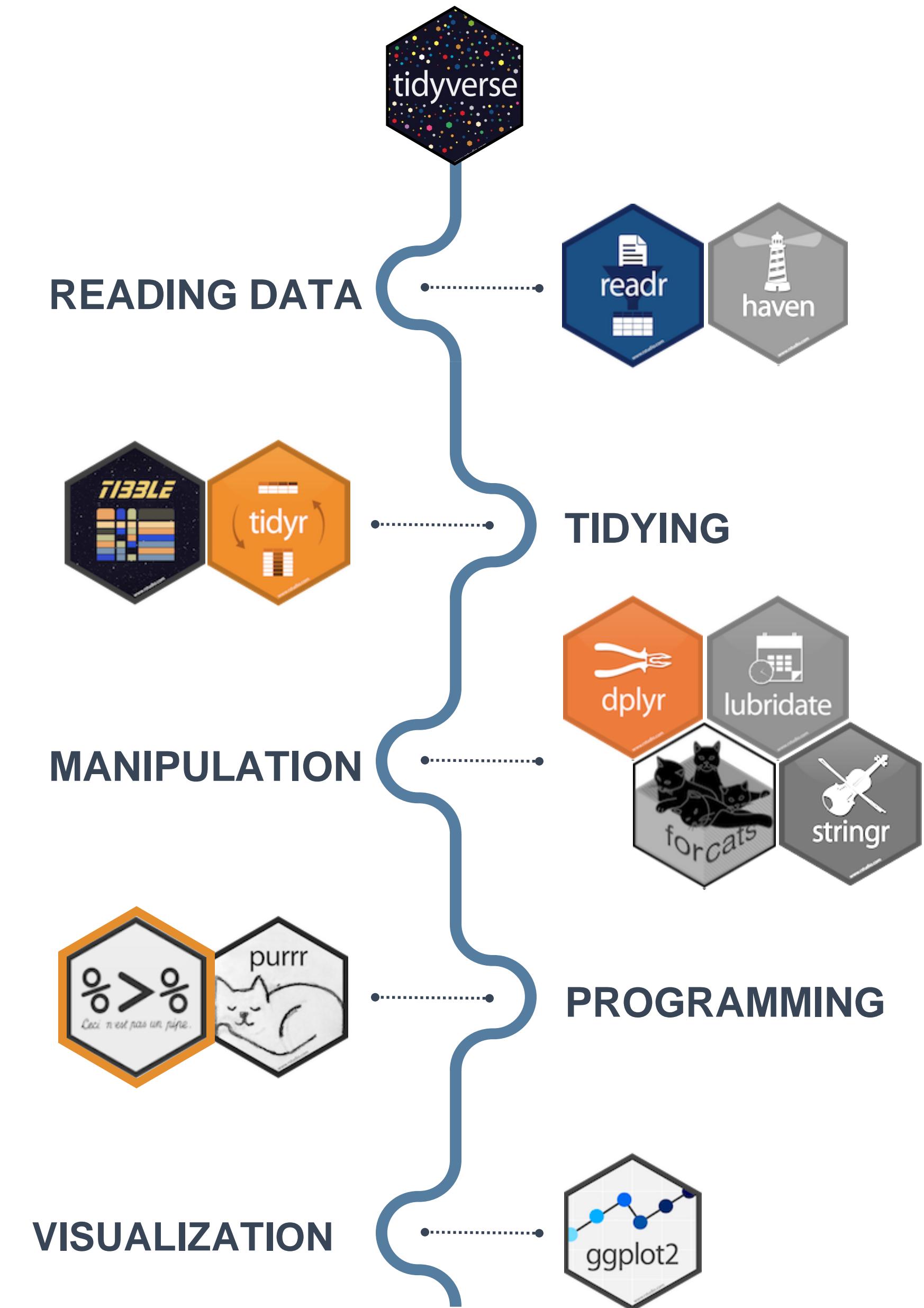
TIDYVERSE

<https://www.tidyverse.org/>

tidyverse is a collection of R packages for data science

“The packages share an underlying design philosophy, grammar, and data structures.” *Wickham and Grolemund*

tidyverse is used to “tidy up” your datasets, so they are easy to work with



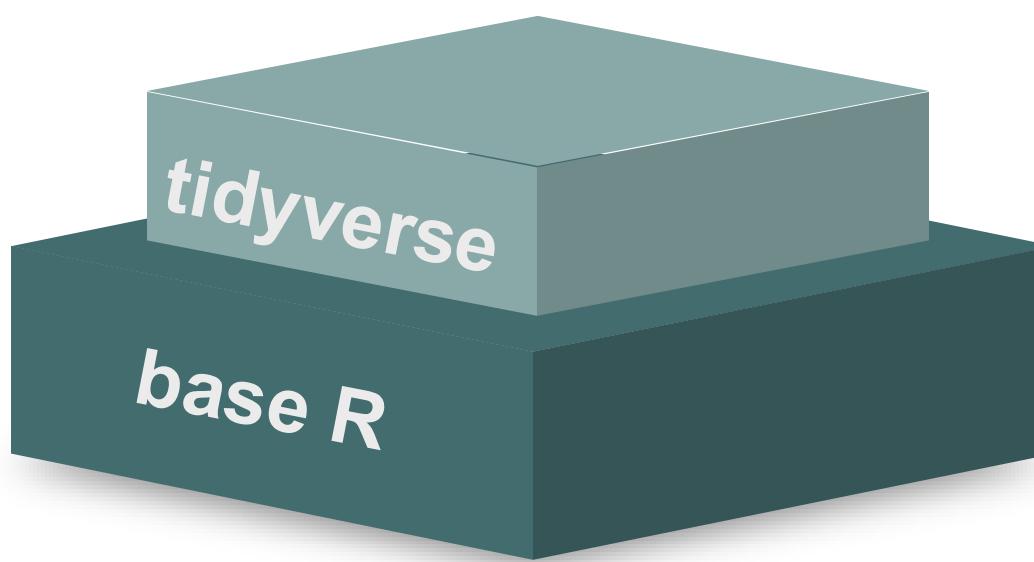
CECI N'EST PAS UNE PIPE

%>%

- You do NOT have to “choose” between tidyverse and base R

BENEFITS

- Short & well-organised code
- Tidy datasets, easy to work with
- Great documentation
- Functions with logical names & inputs



CONSIDERATIONS

- Can be less stable
- “Different syntax”
- Remember what tidyverse is made for!

base R

```
# think from the inside out  
g(f(x), z)
```

tidyverse

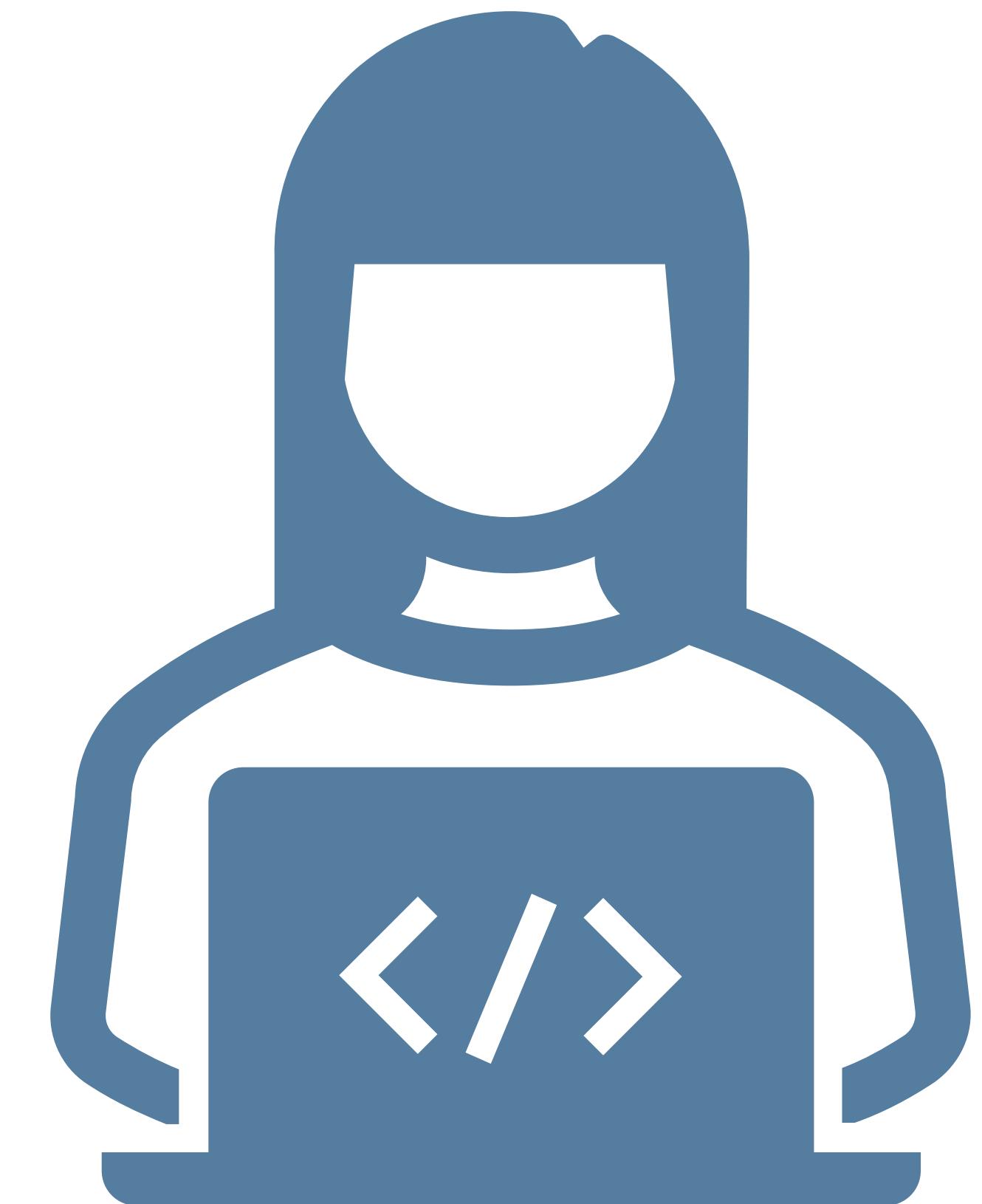
```
# no brain acrobatics  
x %>% f(y) %>% g(z)
```



pipe symbol

— FROM EXCEL TO R

LIVE CODING 2 – TIDYVERSE



TIDYVERSE CHEAT SHEET

readr, tidyr, dplyr, ...

Read Data (*readr*)

Reading tabular data

There are solutions for multiple data types
`read_excel()` # using *readxl* package
`read_table()`
`read_csv()`

Useful arguments

Skip lines: `read_csv(file, skip=1)`
Read subset: `read_csv(file, n_max=1)`

Data types

readr guesses the types of each column and tells you about it
("Parsed with column specifications: ...")

HELP

R Documentation (e.g. enter `?dplyr::filter` and see examples)

Much more info and detailed cheat sheets:

<https://brianward1428.medium.com/introduction-to-tidyverse-7b3dbf2337d5>

It also helps to google "tidyverse + whatever you want to do"

Workflow

Tidyverse workflow

```
df %>%  
  select(col1)  
  
df %>%  
  filter(col1 < x)  
  
df %>%  
  summarize(n_1 = n(),  
            avg_1 = mean(col1),  
            sd_1 = sd(col1))
```

Select column

`df$col1`

Data Manipulation (*dplyr*)

Summary

`summarize()`
`count()`

Group

`group_by()`

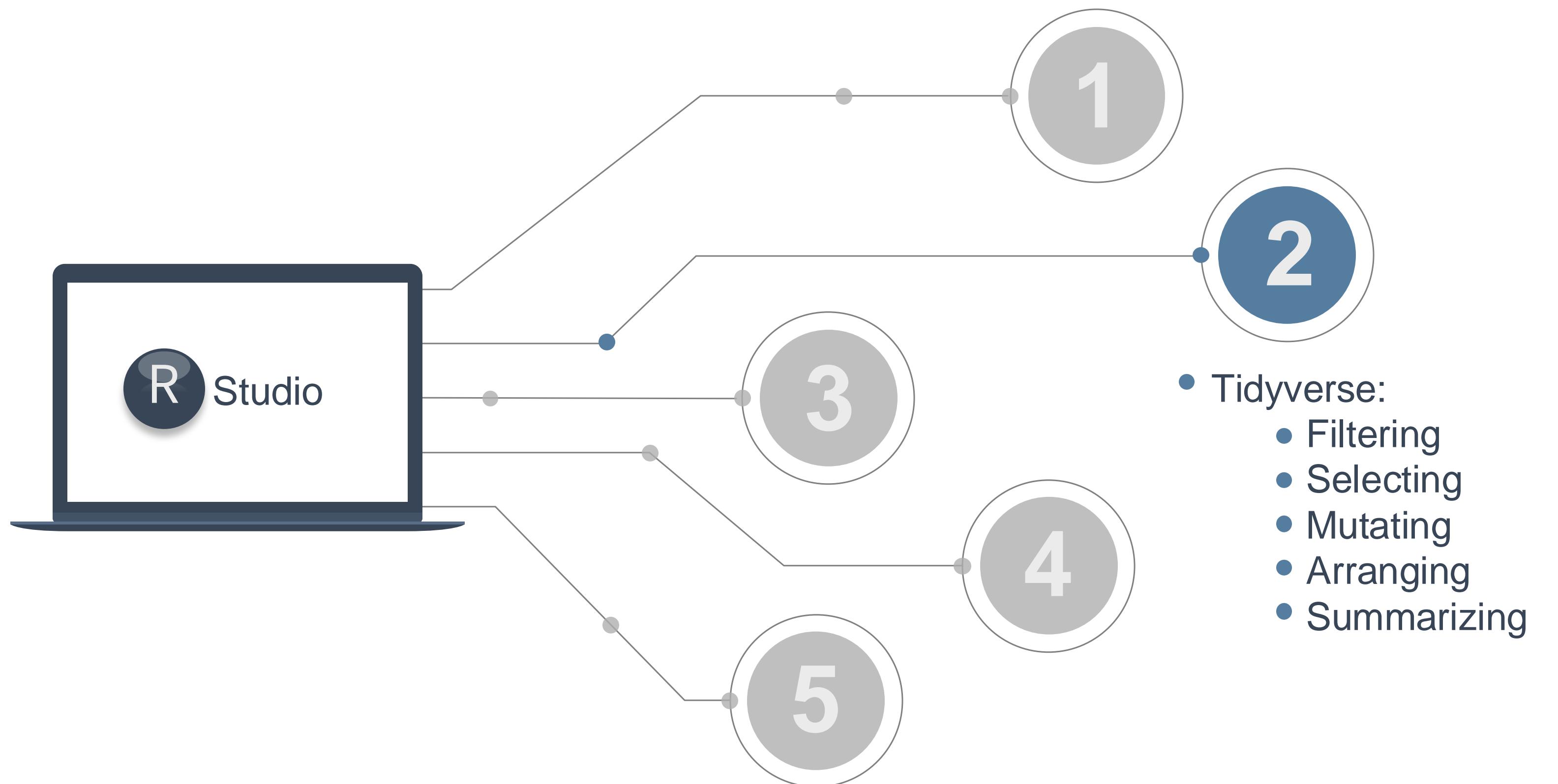
Functions will manipulate each group separately and combine results.

Extract and sort observations (rows)

`filter()` # subset rows by condition
`distinct()` # subset to unique values
`top_n()` # subset by position
`arrange()` # sort low->high, other way with `desc()`

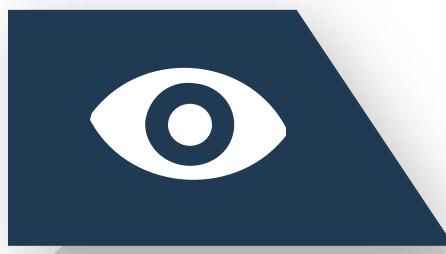
Manipulate variables (columns)

`select()` # subset rows by condition
`mutate(new_name = f(column))`
`mutate(new_name = ifelse(col1 < x, "Yes", "No"))`
`mutate(new_name = col1 + col2)`

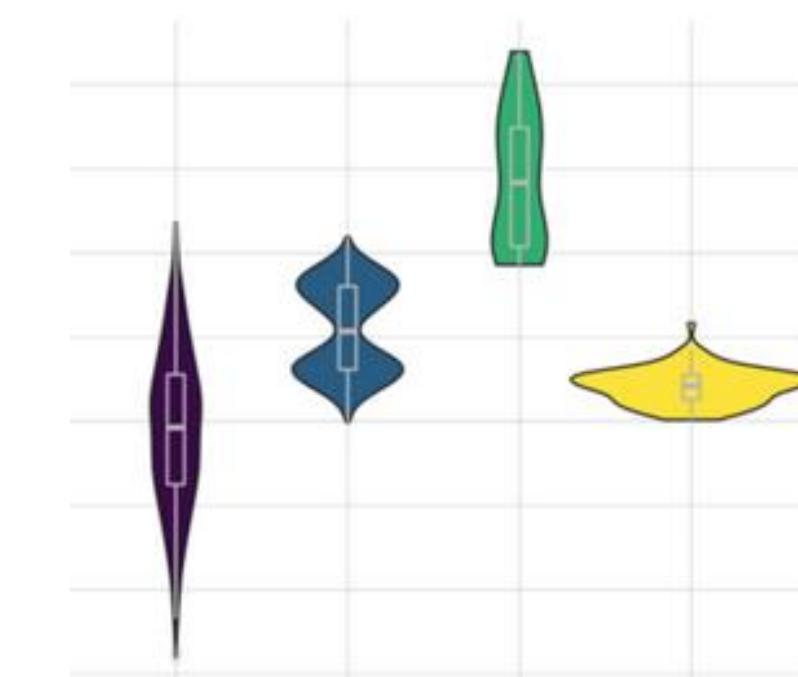


TIDYVERSE EXERCISE 2

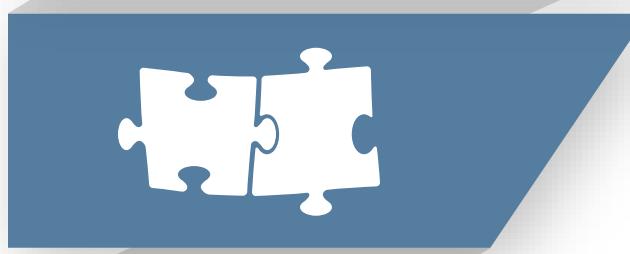
GGPLOT2 - EASY GRAPHICS



Aesthetically pleasing graphics.



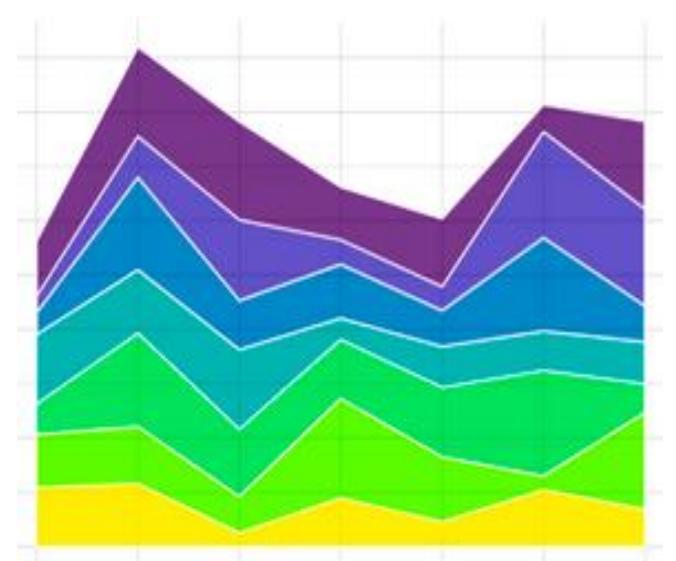
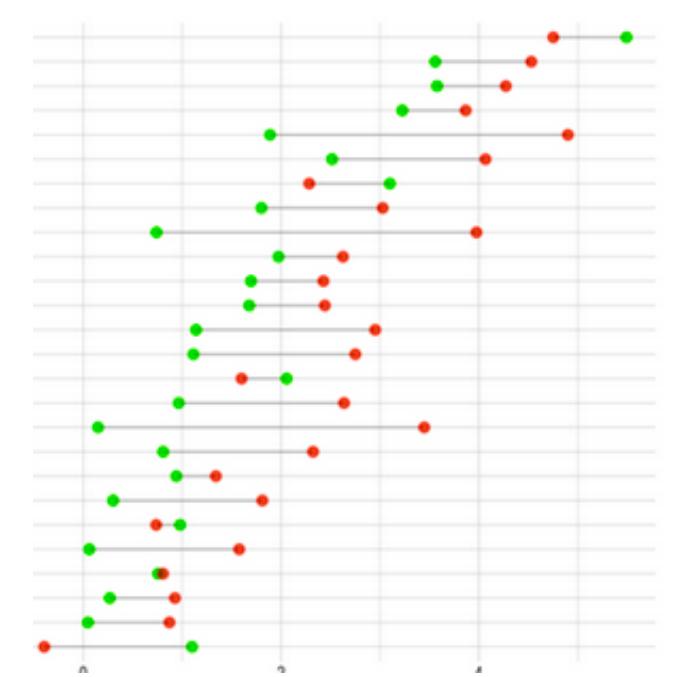
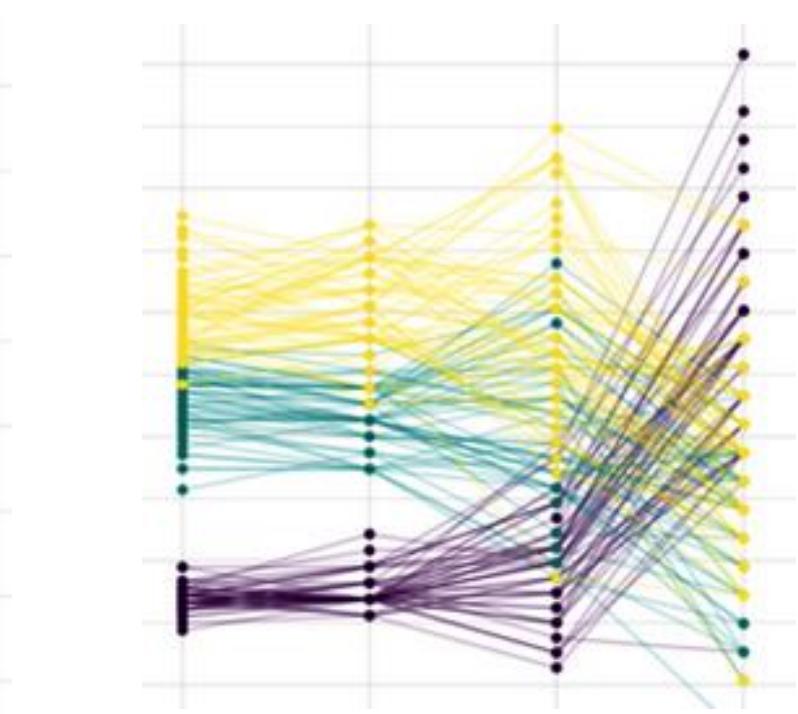
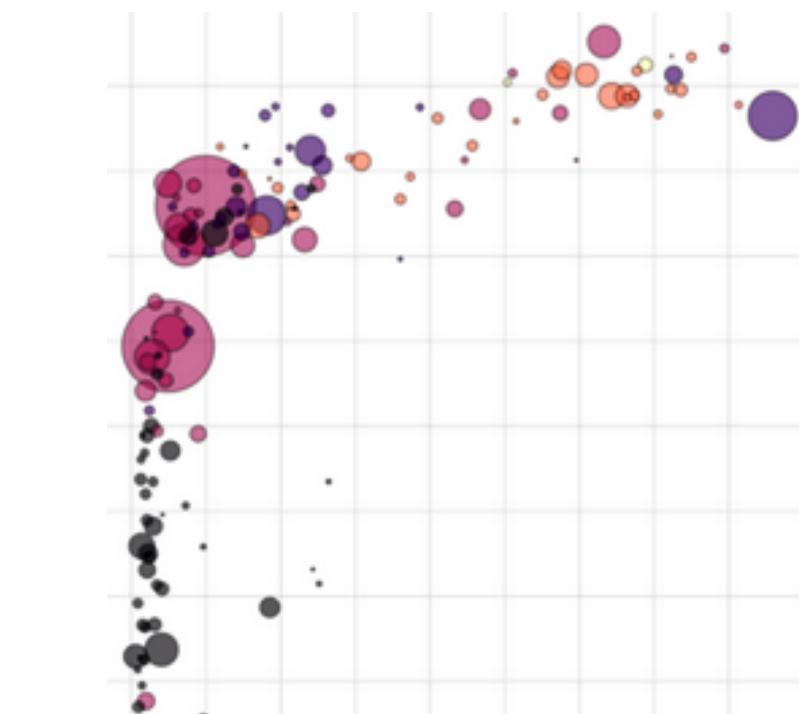
Well-defined “additive” (+) structure.



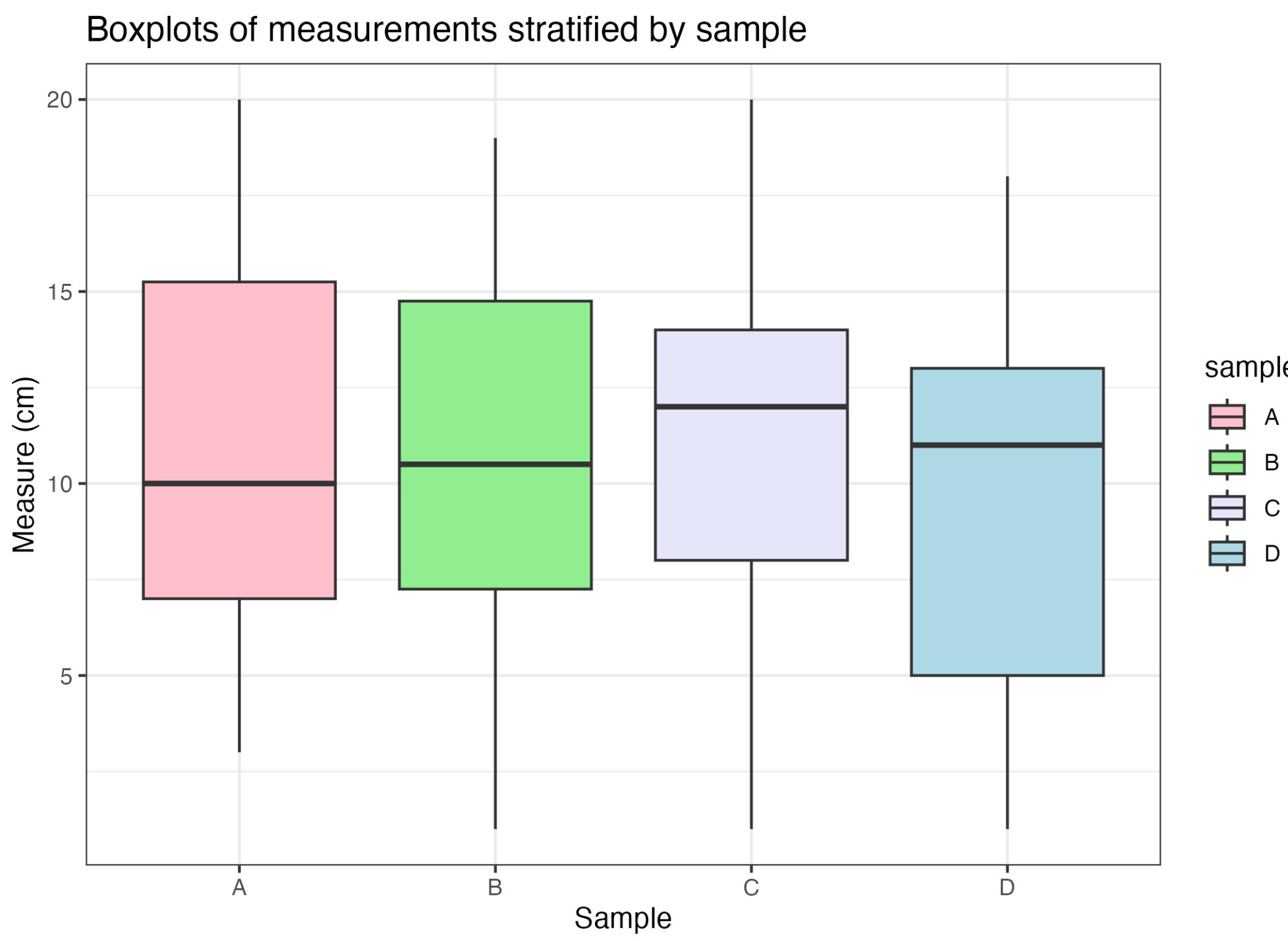
Integrates perfectly with tidy data.



Great documentation & community



GGPLOT2 ADDITIVE STRUCTURE



Dataset, Samples & Observations



```
ggplot(df,  
       aes(x = Sample,  
            y = Measure))
```

Define Plot Type



```
ggplot(df,  
       aes(x = Sample,  
            y = Measure)) +  
  geom_boxplot()
```

Color by Group



```
ggplot(df,  
       aes(x = Sample,  
            y = Measure,  
            fill = Sample)) +  
  geom_boxplot()
```

Title and Legend



```
... +  
  labs(title = "Boxplots of  
measurements stratified by sample")
```

Custom Colors



```
... +  
  scale_fill_manual(values =  
c("pink", "lightgreen", "lavender",  
"lightblue"))
```

Background

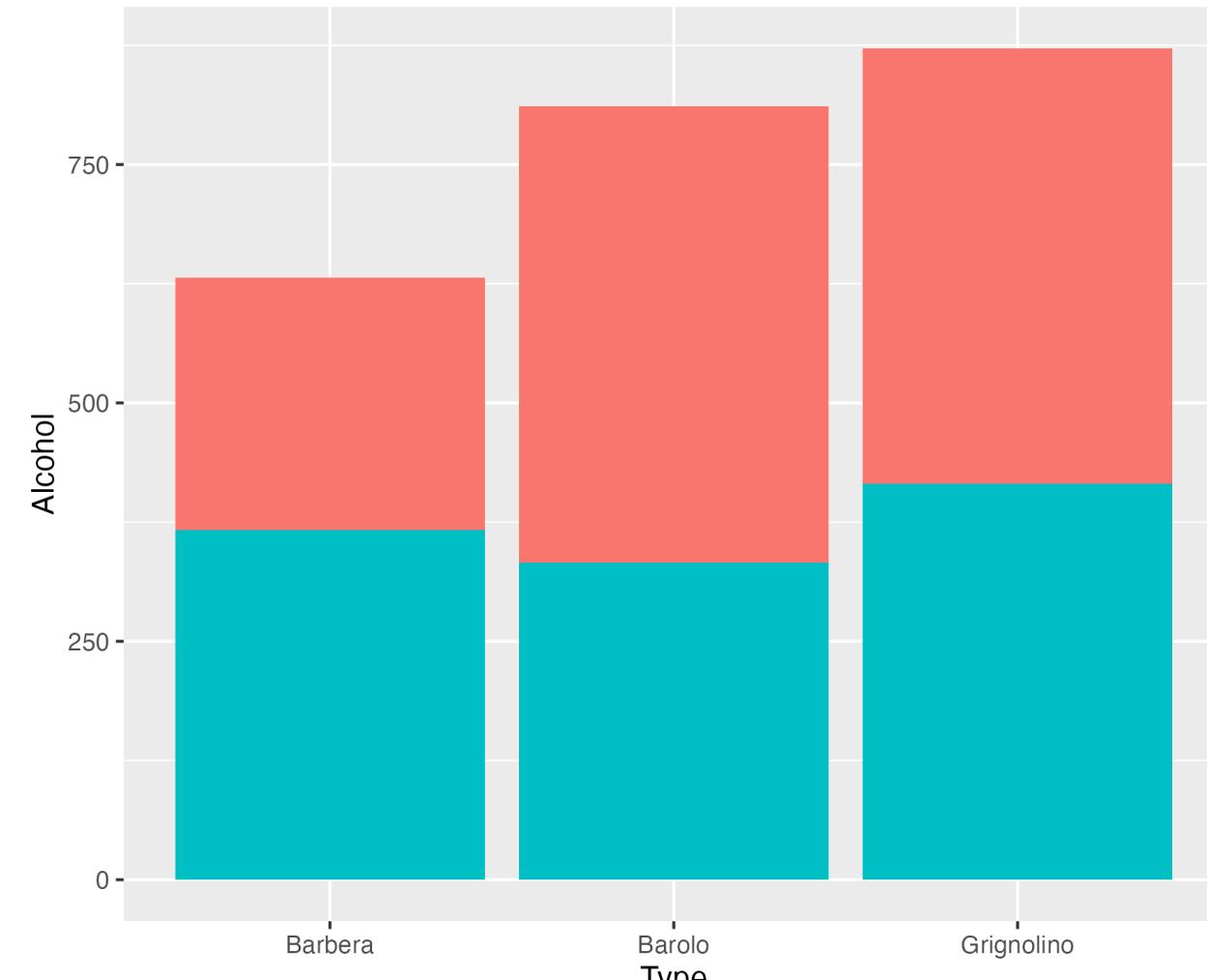


```
... +  
  theme_bw()
```

GGPLOT BASIC STRUCTURE

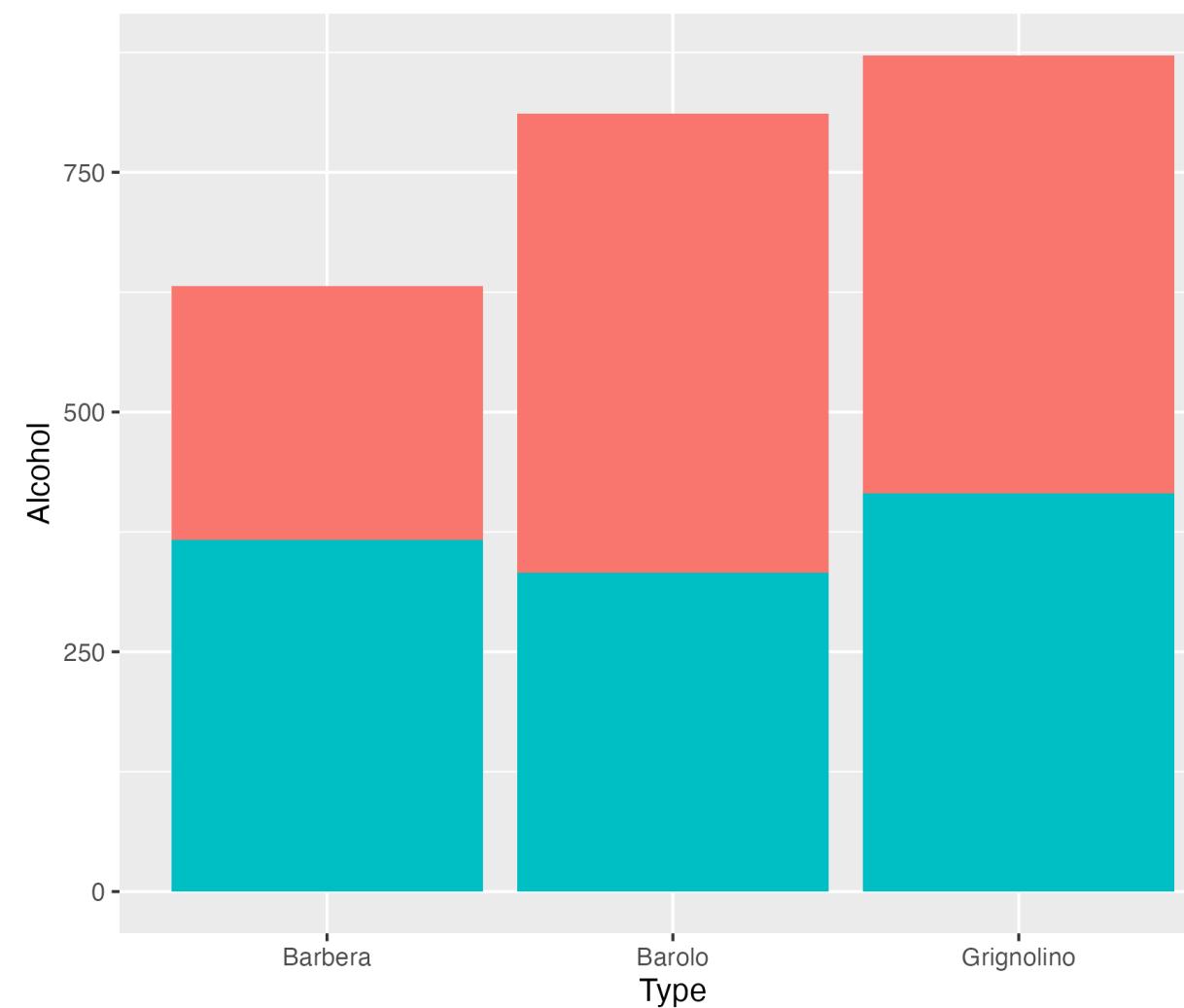
Geoms inherit the parameters from the ggplot they are added to:

```
ggplot(my_wine,
       aes(x = Type,
           y = Alcohol,
           fill = Country)) +
  geom_col()
```



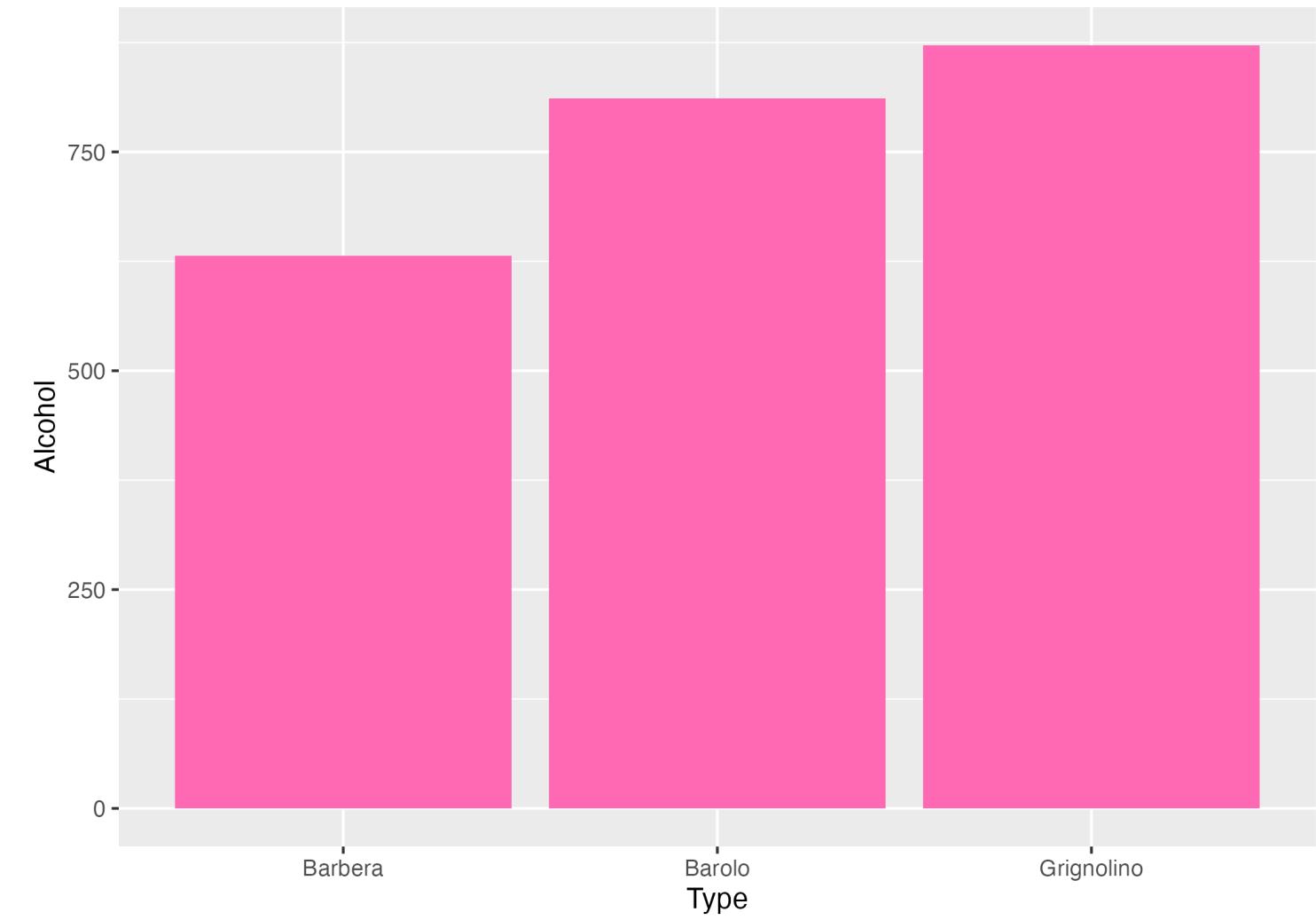
Alternatively, you can specify additional parameters directly in the geom:

```
ggplot(my_wine,
       aes(x = Type,
           y = Alcohol)) +
  geom_col(aes(fill = Country))
```



Things outside the aes are applied to everything!

```
ggplot(my_wine,
       aes(x = Type,
           y = Alcohol)) +
  geom_col(fill = "hotpink")
```



GGPLOT CHEAT SHEET

Define Plot:

```
ggplot(data = my.data,  
       aes(x = x.var,  
            y = y.var))
```

Add Plot Type:

```
... + geom_point() # scatter plot  
... + geom_line()  
... + geom_boxplot()
```

```
... + geom_col()  
... + geom_density()  
... + geom_histogram()
```

One Color:

```
ggplot(..., aes(...),  
       color = "green")
```

Color Fill by Group:

```
ggplot(..., aes(...,  
               fill = z.var))
```

Custom Colors:

```
... + scale_*_manual(values = c())  
ex: scale_color_manual(values = c("blue", "pink"))
```

More Colors:

```
... + scale_fill_grey(start = 0.2, end = 0.8)  
... + scale_fill_gradient(low="white", high="red")
```

Labels:

```
... + labs(title = "Title",  
          x = "X label",  
          y = "Y label")
```

Theme:

```
... + theme_bw()  
... + theme_minimal()  
... + theme_dark()  
... + theme_classic()
```

Text:

```
... + theme(legend.position = *)  
* = "none", "top", "bottom", "left", "right"
```

GET
STARTED

COLORS

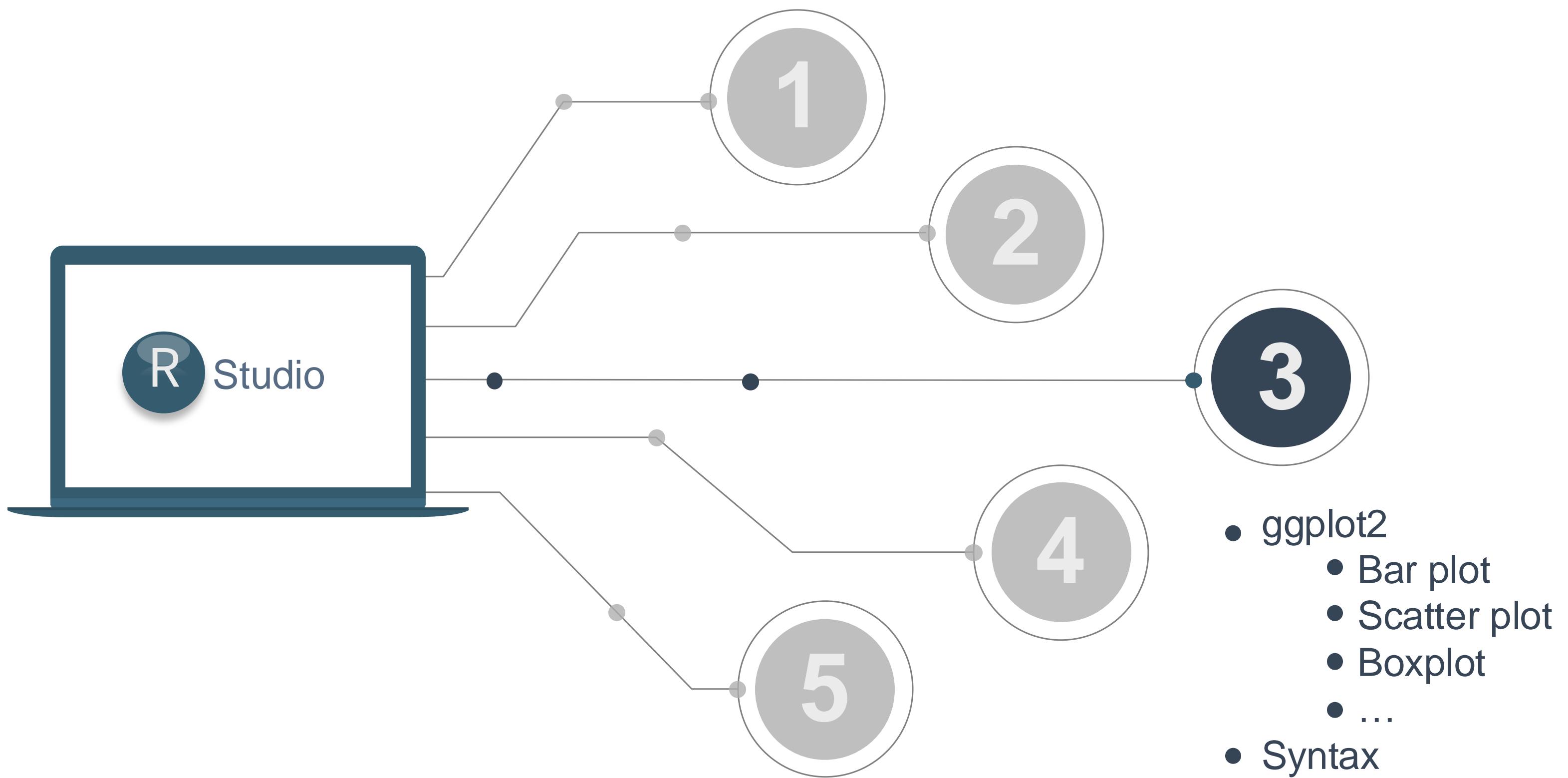
COLOR SCALES
& THEMES

TEXT

— FROM EXCEL TO R

LIVE CODING 3 – GGPLOT 2



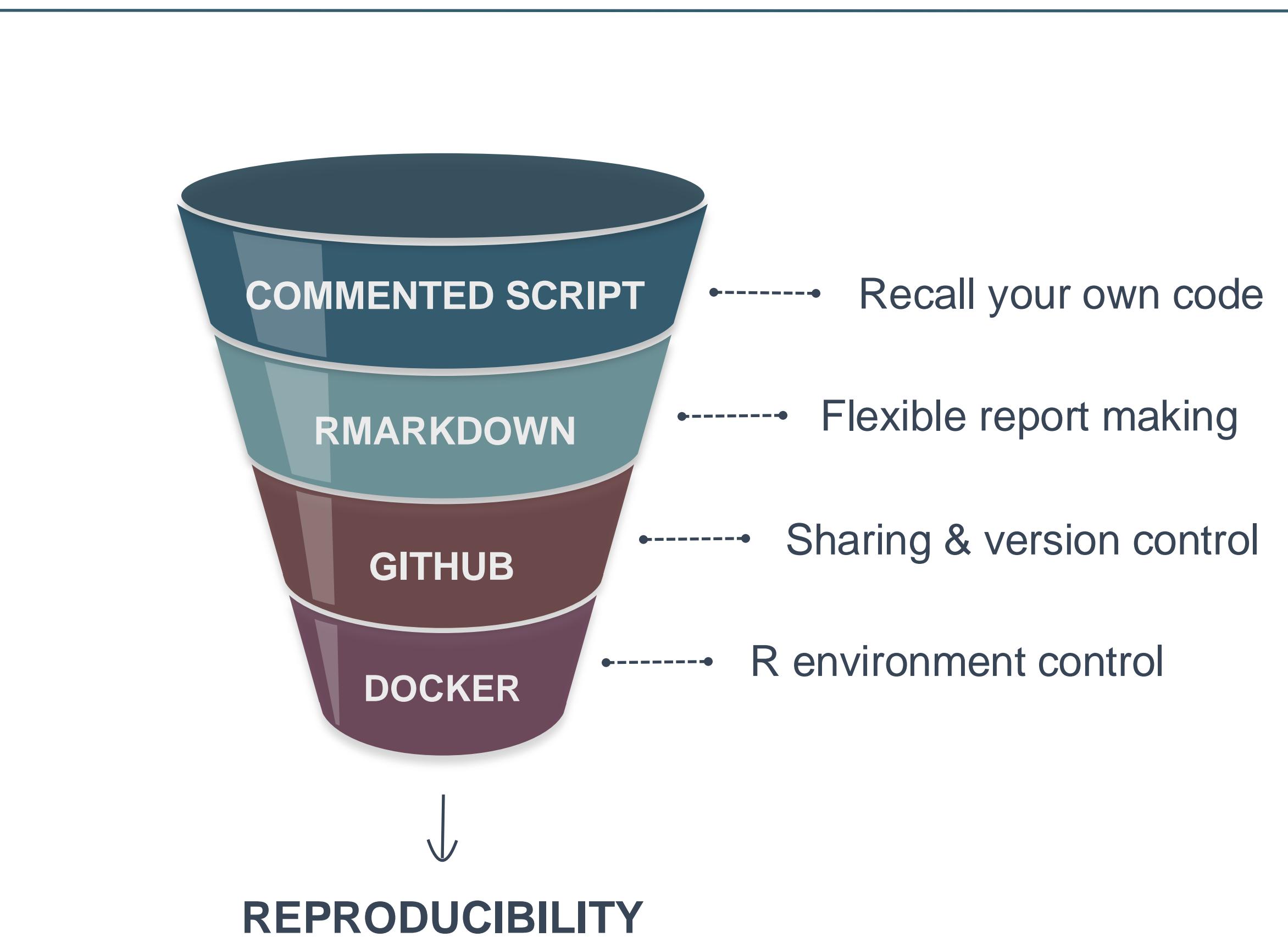
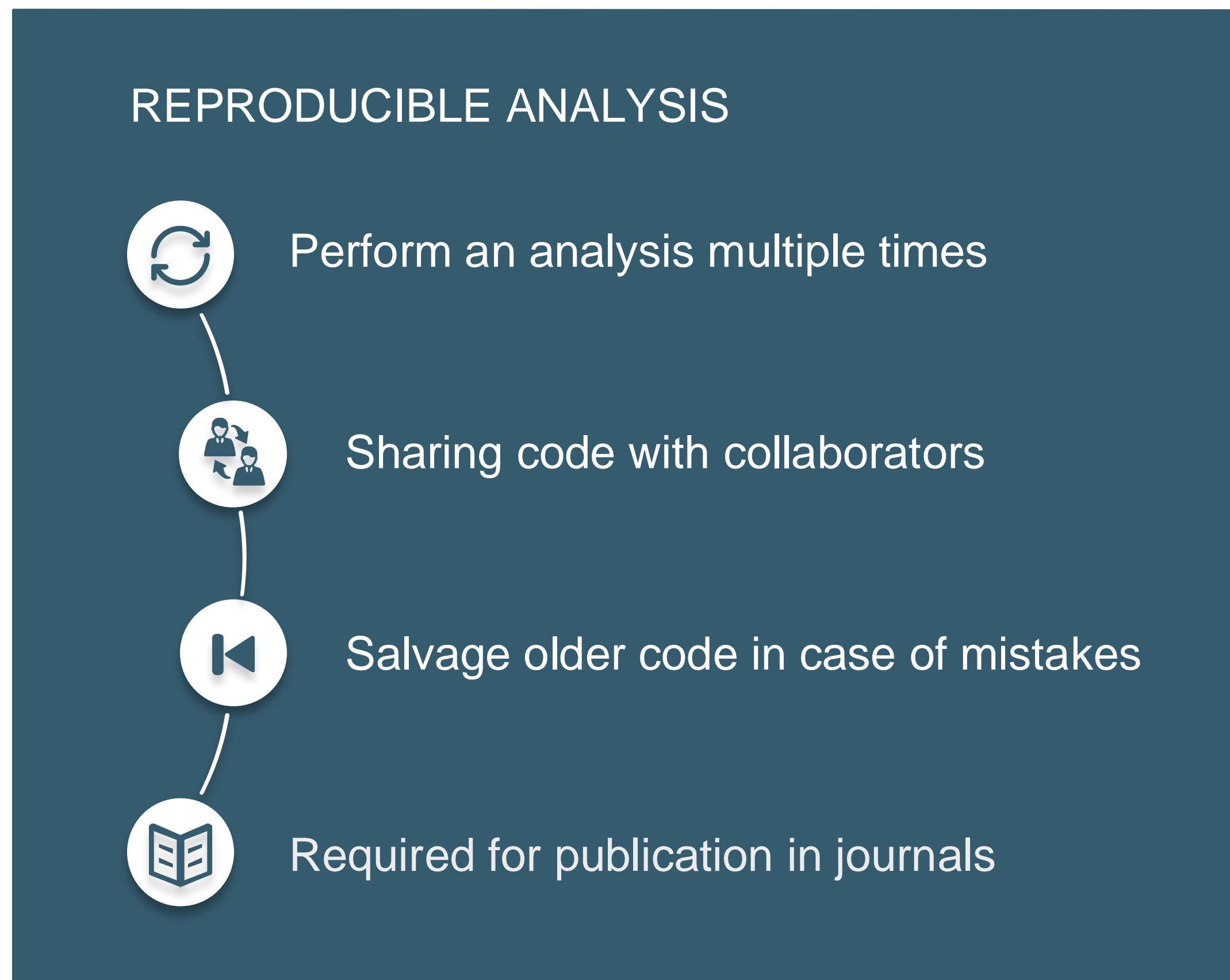


— GG PLOT2 EXERCISE 3

FROM EXCEL TO R

APPLIED STATISTICS

REPRODUCIBILITY IN R





- A markup language
 - Plain text format
 - Converted into HTML, PDF, LATEX
- Finds code embedded in text
 - Evaluates the code
 - “Knits” the results into the document

— <https://www.markdowntutorial.com/>

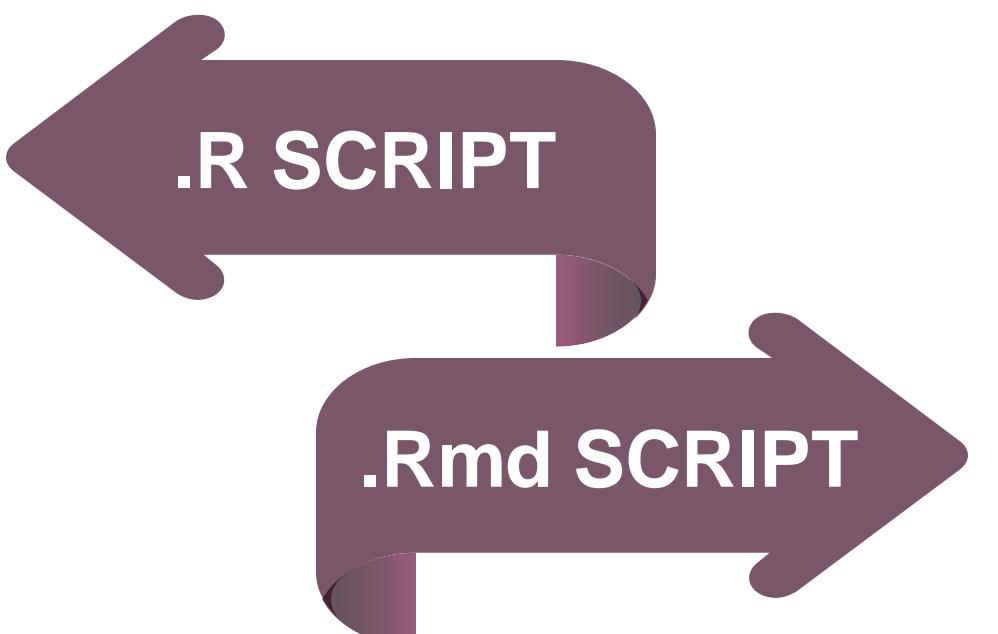
R SCRIPT & RMARKDOWN

HOW TO

- Write code as normal
- Comment text with #

USE FOR

- Testing new code
- Big data analysis
- Software development



HOW TO

- Write text as normal
- Embed code `'{r} my.code'`

USE FOR

- Reports for yourself
- Reports for collaborators
- Tutorials

RMarkdown

The diagram illustrates the RStudio interface for R Markdown. It features two main windows side-by-side, connected by a curved arrow pointing from the left pane to the right.

Left Pane (R Markdown Editor):

- Toolbar:** Includes icons for back, forward, file operations, ABC, search, and a prominent **Knit** button.
- Code Area:** Displays the R Markdown code. The code includes a YAML header, R code chunks, and text sections. A snippet of text is highlighted in blue.
- Status Bar:** Shows the file name "R Markdown" and the line number "4:18".

Right Pane (HTML Preview):

- Toolbar:** Includes icons for Open in Browser, Find, Publish, and a refresh symbol.
- Content Area:** Displays the generated HTML document. It includes a title section ("Data Science Lab" and "9/14/2020"), a main heading ("R Markdown"), and a descriptive paragraph about R Markdown. It also shows an R code chunk and its output (summary of the cars dataset).

Code and Output Snippets:

```
1 ---  
2 title: "Rcourse"  
3 author: "Data Science Lab"  
4 date: "9/14/2020"  
5 output: html_document  
---  
6  
7 ```{r setup, include=FALSE}  
8 knitr::opts_chunk$set(echo = TRUE)  
9 ```  
10  
11 ## R Markdown  
12  
13 This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. The document is a flat text type document which can be read without opening it in RStudio. Making a markdown document is easy, For example, if you want to make something bold use two stars, For italics use underscore, etc. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.  
14  
15 When you click the Knit button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:  
16  
17 Here is a summary of the cars dataset  
18 ```{r cars}  
19 summary(cars)  
20 ```  
21  
22  
23 Here is some math  
24 ```{r, eval=TRUE}  
25  $((5+6+7)/3)*12$   
26 ```  
27  
28 Here is a plot
```

summary(cars)

```
##      speed      dist  
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

Here is some math

```
 $((5+6+7)/3)*12$ 
```

```
## [1] 72
```

RMarkdown Cheat Sheet

Begin .Rmd:

```
---
```

```
title: My Project Name
```

```
output:
```

```
    html_document (pdf_document, ...)
```

```
---
```

Code Chunk:

```
```{r}  
some R code
```
```

Global Option:

```
```{r setup, include=FALSE}  
knitr::opts_chunk$set(echo = TRUE)
```
```

Code Options:

```
echo (= TRUE or FALSE - print my code)  
eval (= TRUE or FALSE - run my code)  
warning (= TRUE or FALSE display warning messages)
```

Figure Options:

```
fig.align (= 'left', 'right', 'center')  
fig.cap (= 'my figure caption')  
fig.height (= n), fig.width (= n)
```

GETTING
STARTED

CHUNK
OPTIONS

Header:

Header size ranging from largest (one #) to smallest (six #):
my.text, ## my.text, ### my.text, etc.

Text:

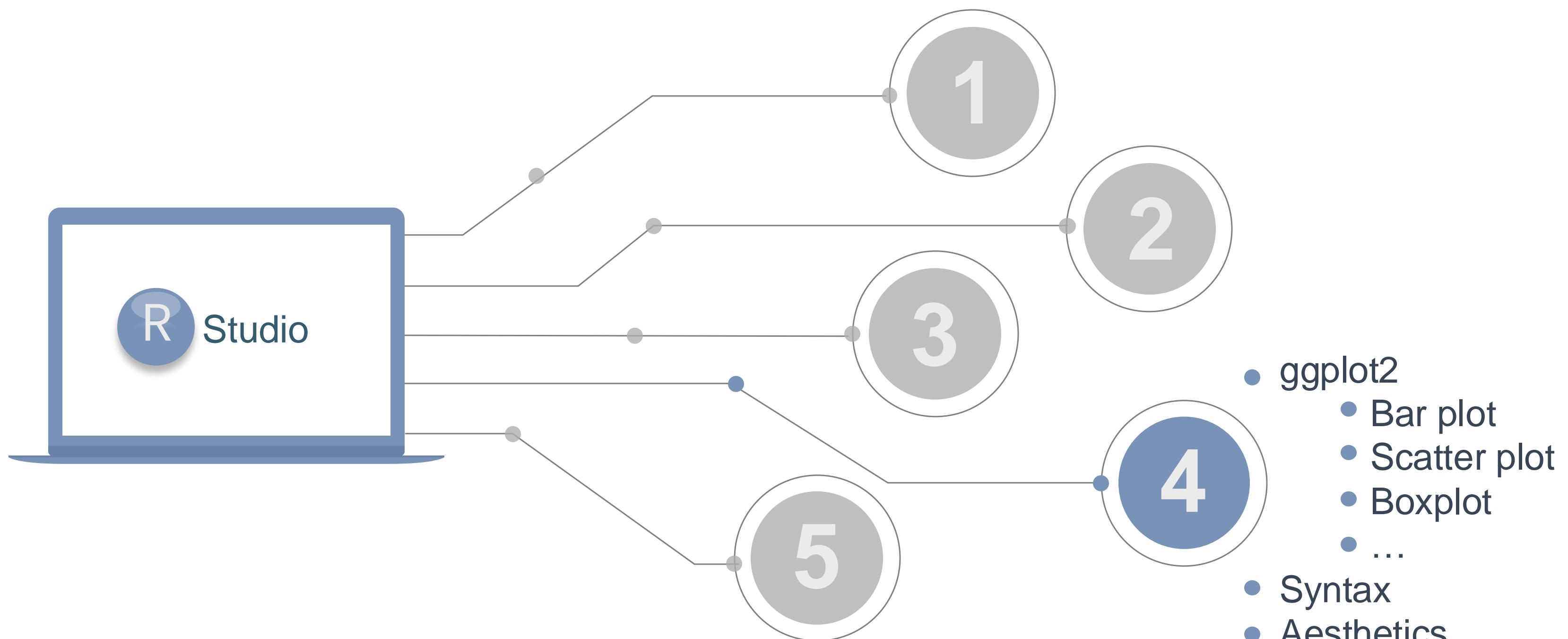
italics
bold
`highlighted`

Lists:

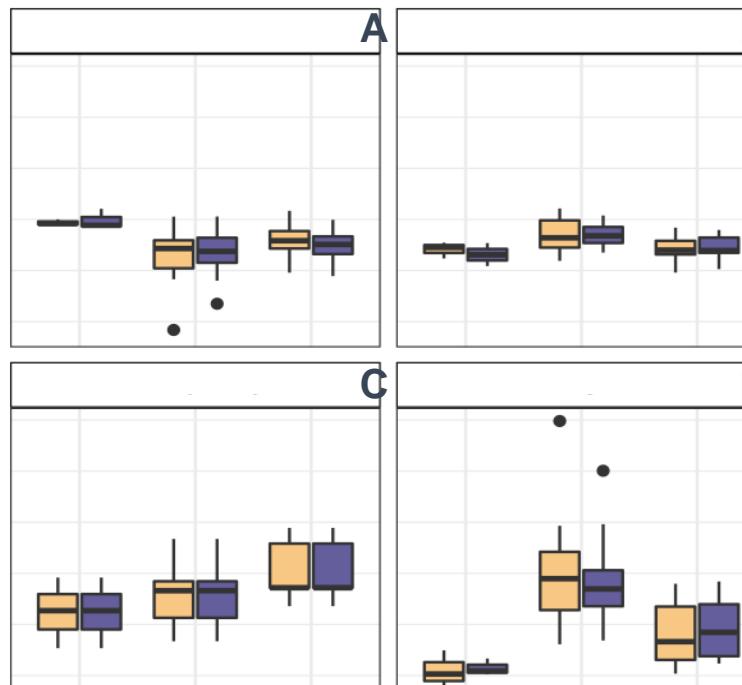
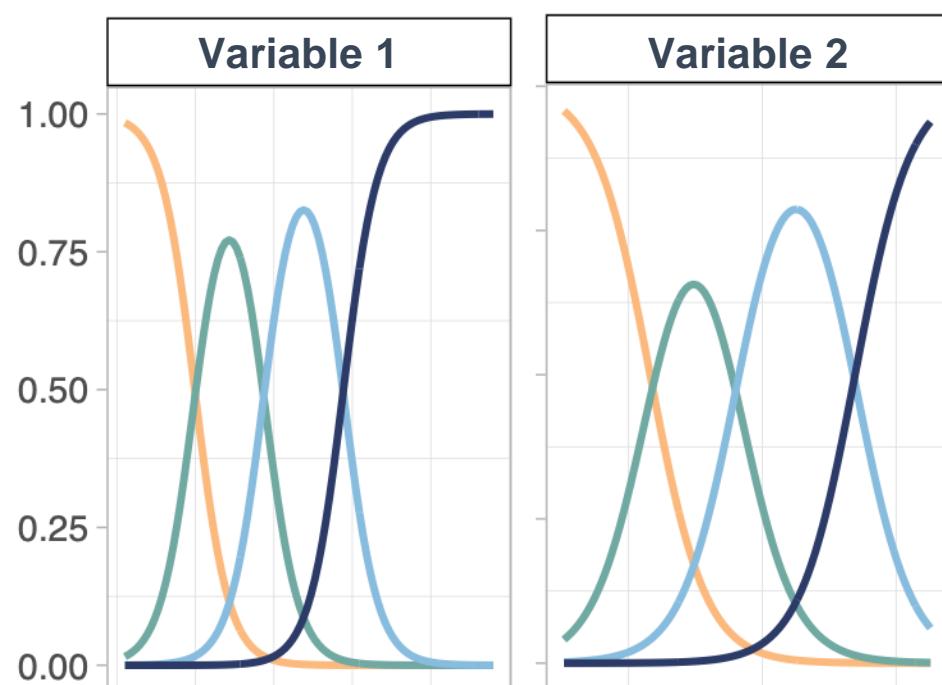
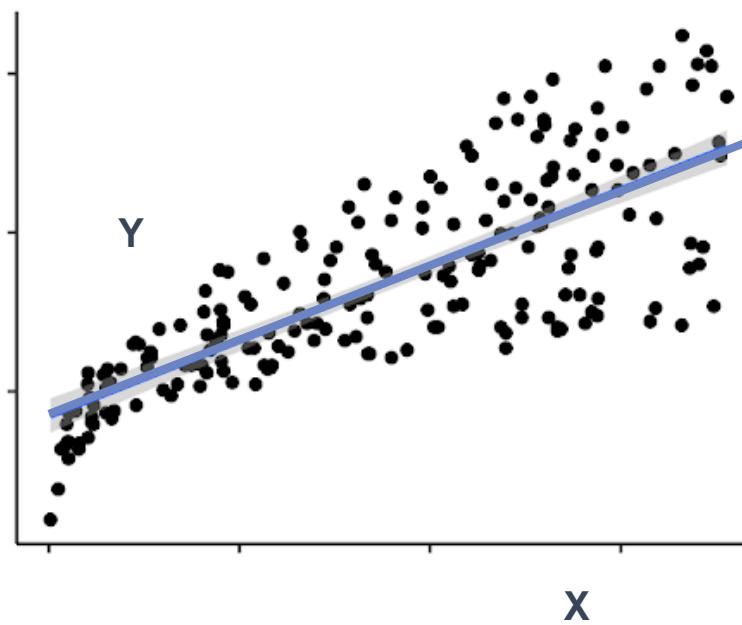
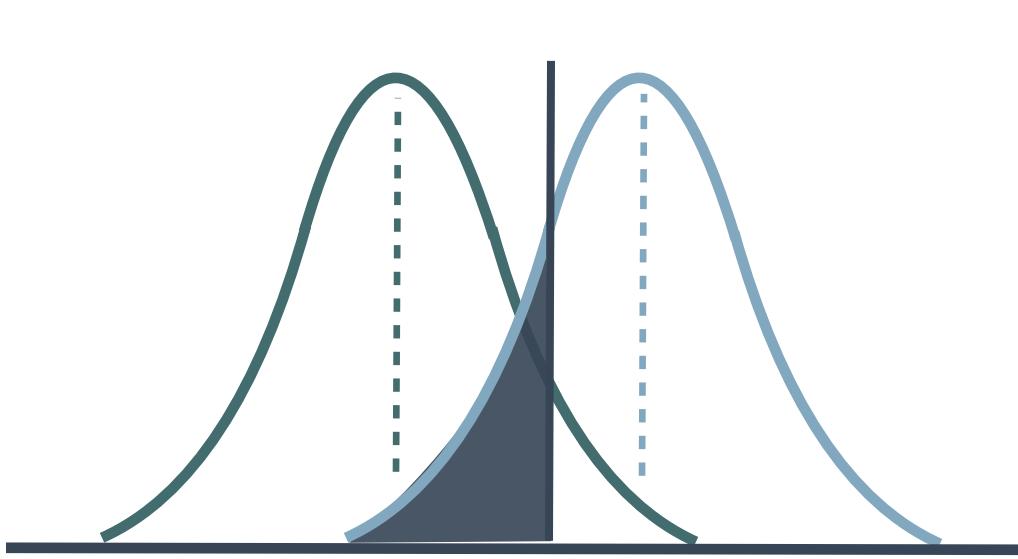
- * List item1 (filled dot)
 - + sub-item1 (open dot)
- 1. List item1 (numbered)
 - i) sub-item1 (roman)

TEXT

— STATISTIC EXERCISE 4



R - A STATISTICAL SCRIPTING LANGUAGE



MODEL FUNCTIONS
`lm()`, `glm()`
`lmer()`, `glmer()`,
`nls()`, ...

EMMEANS PACKAGE
`emmeans()`,
`pairs()`, `cld()`

APPLY TO MODEL
`summary()`, `anova()`,
`confint()`, `predict()`,
`drop1()`, `update()`,
`step()`, ...

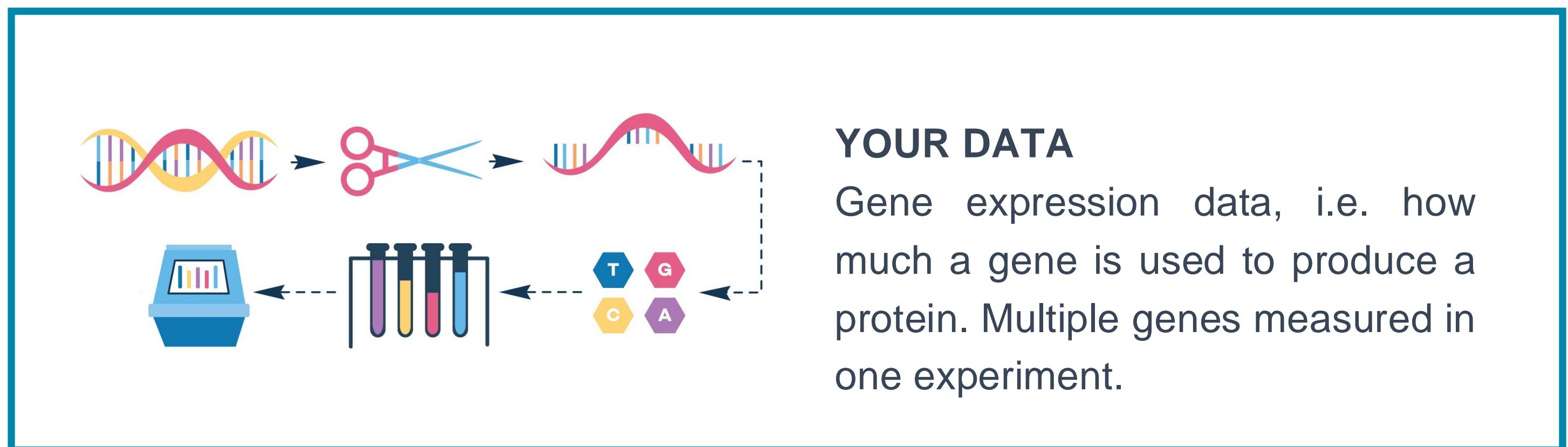
MORE FUNCTIONS
`t.test()`, `cor()`,
`cor.test()`, `aov()`,
`quantile()`,
`p.adjust()`,
`rank()`, ...

Let's use R in a statistical analysis



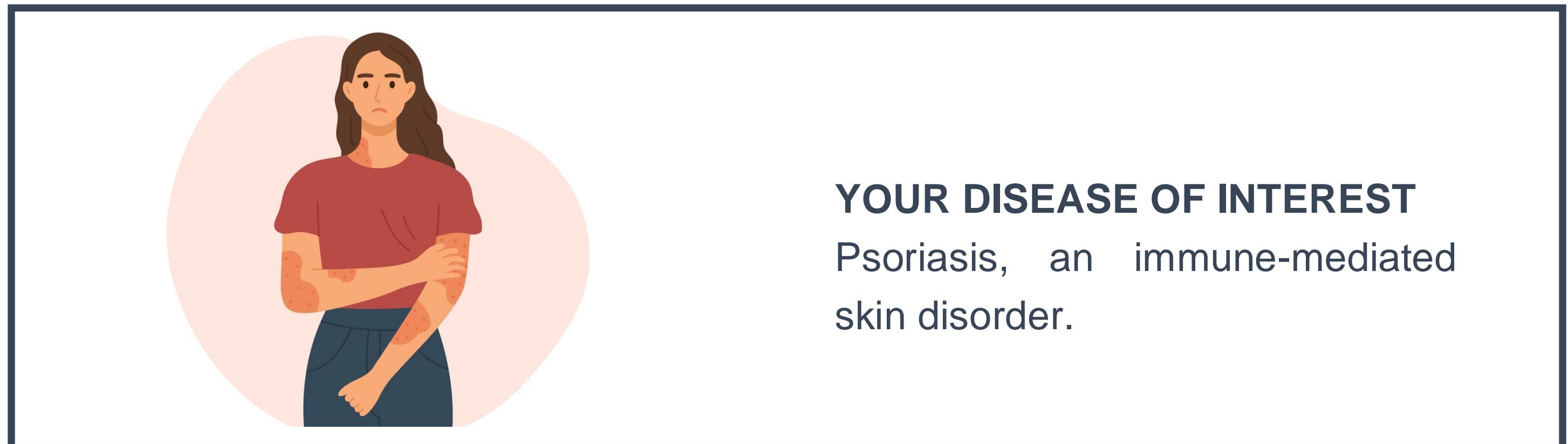
YOU

The researcher with R skills!



YOUR DATA

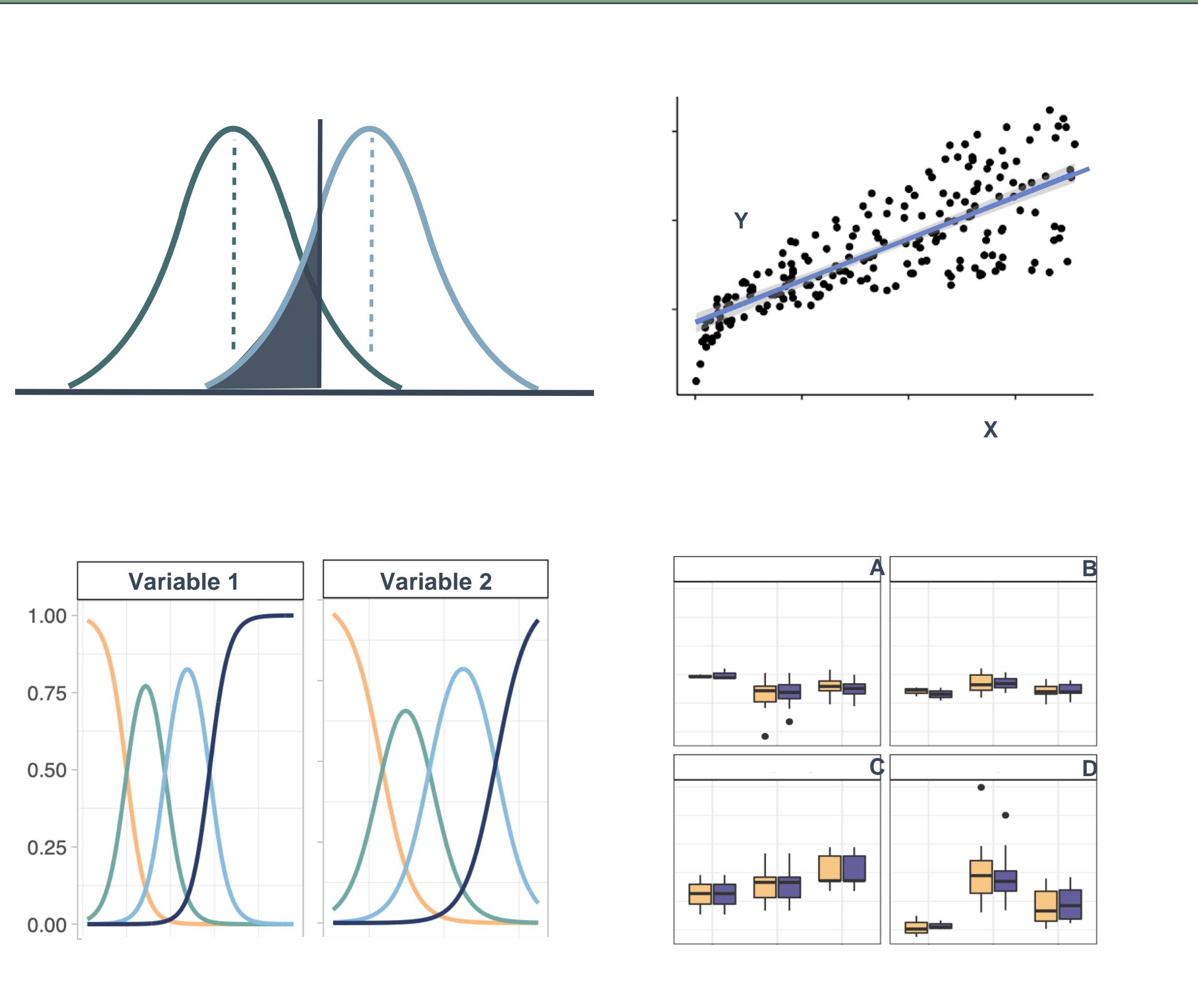
Gene expression data, i.e. how much a gene is used to produce a protein. Multiple genes measured in one experiment.



YOUR DISEASE OF INTEREST

Psoriasis, an immune-mediated skin disorder.

R - A STATISTICAL SCRIPTING LANGUAGE



During this session:

- *Cooperatively* discuss and share ideas about the data
- Apply steps of basic statistical analysis for hypothesis testing consistent with the given data
- Suggest conclusions based on your analysis, regarding the association between psoriasis and gene expression levels

<https://rstudio.com/resources/cheatsheets/> for various relevant cheat sheets.
Other example: <https://www.dummies.com/programming/r/statistical-analysis-with-r-for-dummies-cheat-sheet/>

STATS CHEAT SHEET

Import Data:

```
read_excel("my.data.xlsx")
```

Overview of Data:

```
summary(my.data)  
nrow(my.data)
```

```
length(my.data)  
names(my.data)
```

Linear:

```
lm(y~x, data=my.data)  
confint(model)
```

Logistic:

```
glm(y~x,  
data=my.data)
```

Linear Mixed:

```
lmer(y~x + (1|z),  
data=my.data)
```

Check Model:

```
summary(model)  
par(mfrow=c(2,2))  
plot(model)
```

ANOVA:

```
anova(model2, model1)
```

F-Test:

```
drop1(model, test="F")
```

Emmeans:

```
emmeans(model, ~x)  
pairs(emmeans(model, ~x))
```

Check Type:

```
table(my.data$x)  
is.numeric(my.data$x)  
is.factor(my.data$x)
```

Change Type:

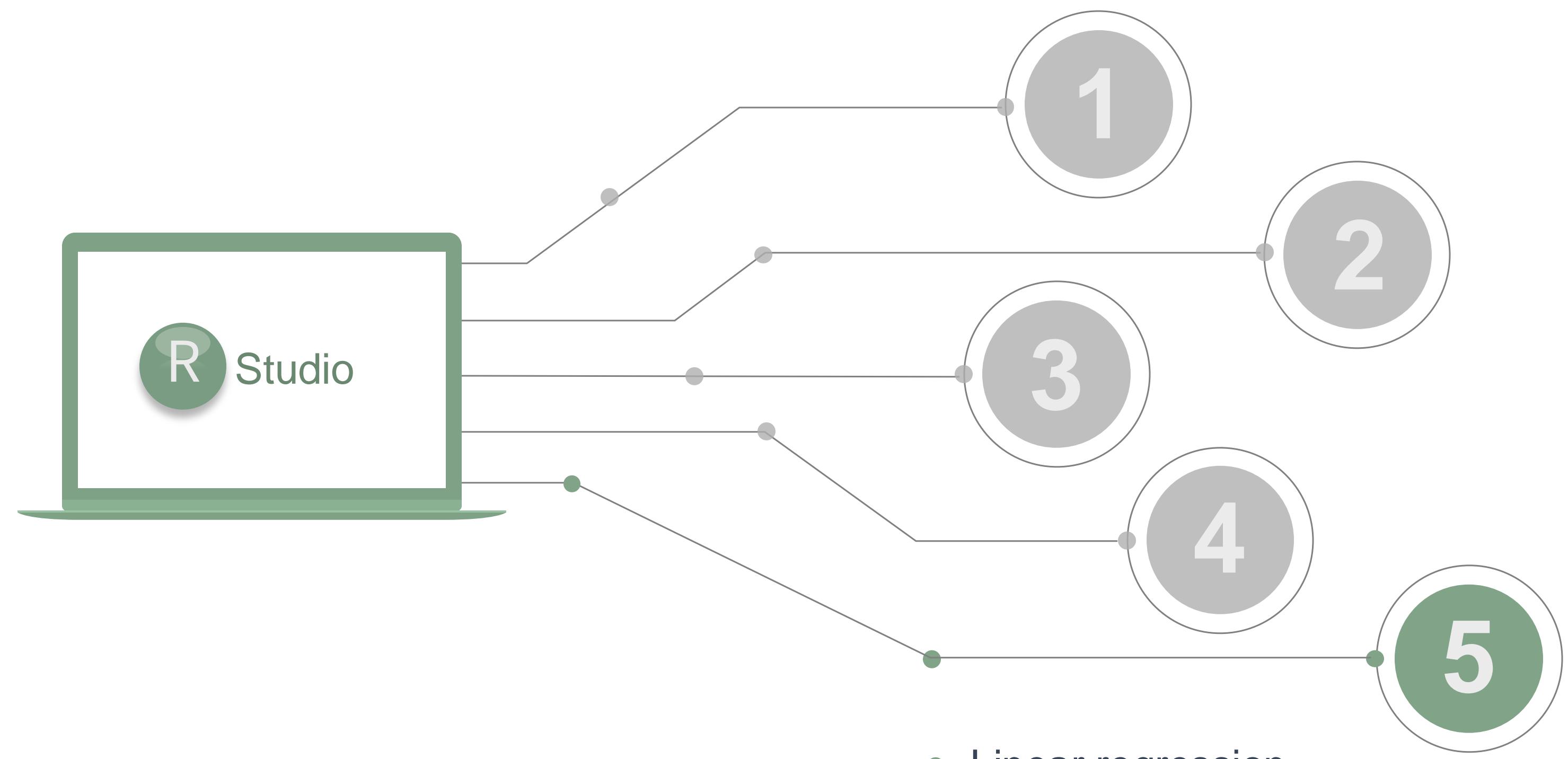
```
my.data <- mutate(my.data, x = factor(x))  
my.data$z <- as.numeric(my.data$z)
```

GET
STARTED

REGRESSION
MODELS

TESTS/COMPARISONS

VARIABLES

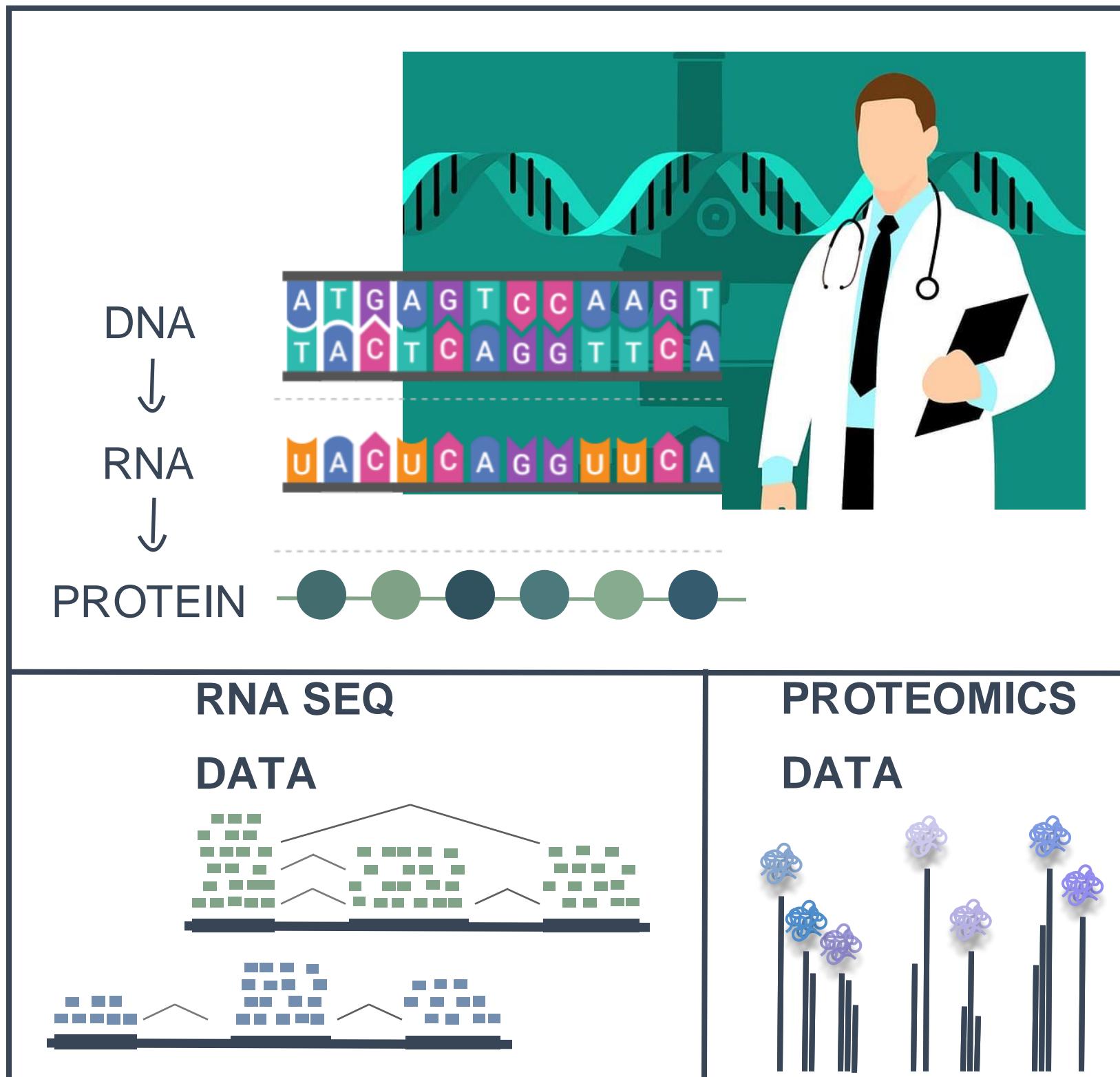


- Linear regression
- Summary Statistics
- ANOVA
- Logistic regression
- Clustering
- Correlation

— Statistics in R
EXERCISE 5

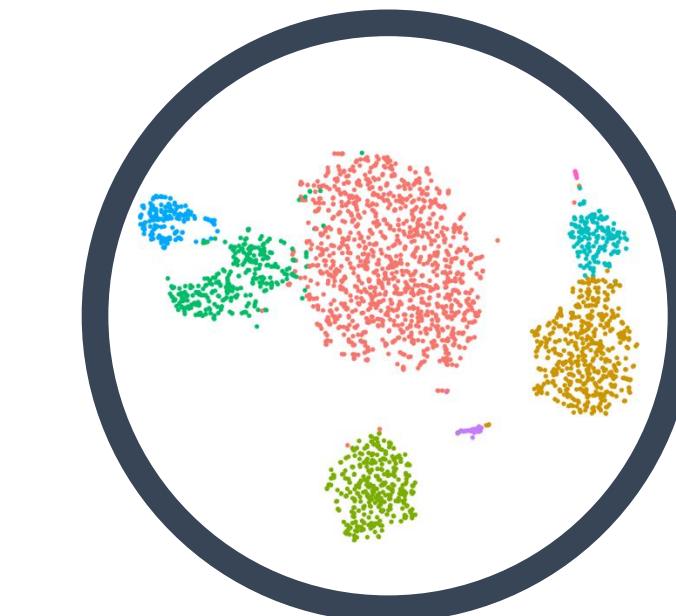
BIOINFORMATICS IN R

HIGH THROUGHPUT DATA



BIOINFORMATIC ANALYSIS

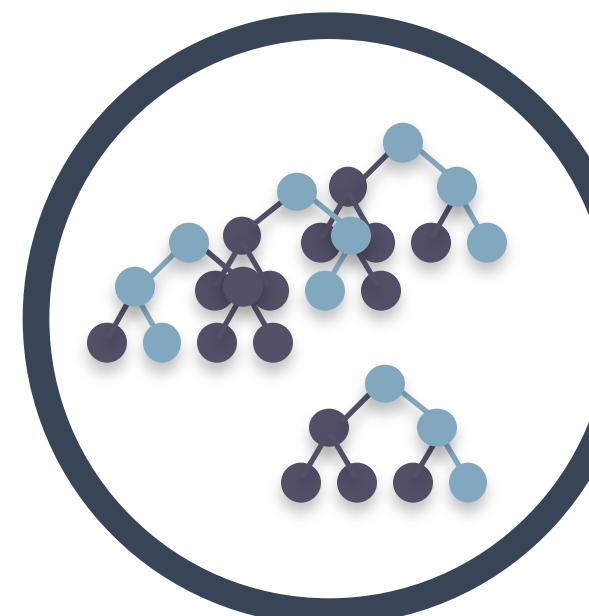
DIMENSIONALITY REDUCTION



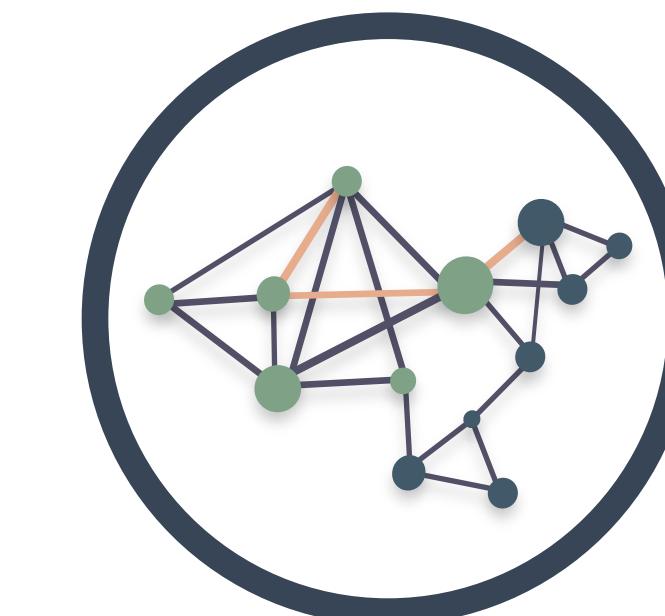
CLUSTERING



MACHINE LEARNING



NETWORK ANALYSIS



THE TOP OF THE R ICEBERG



STATISTICAL ANALYSIS

Statistical models (linear, generalized, mixed, ...)

Statistical tests (t-test, chisq, anova, ...)

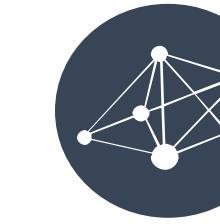
Survival analysis (Cox, Kaplan meier)



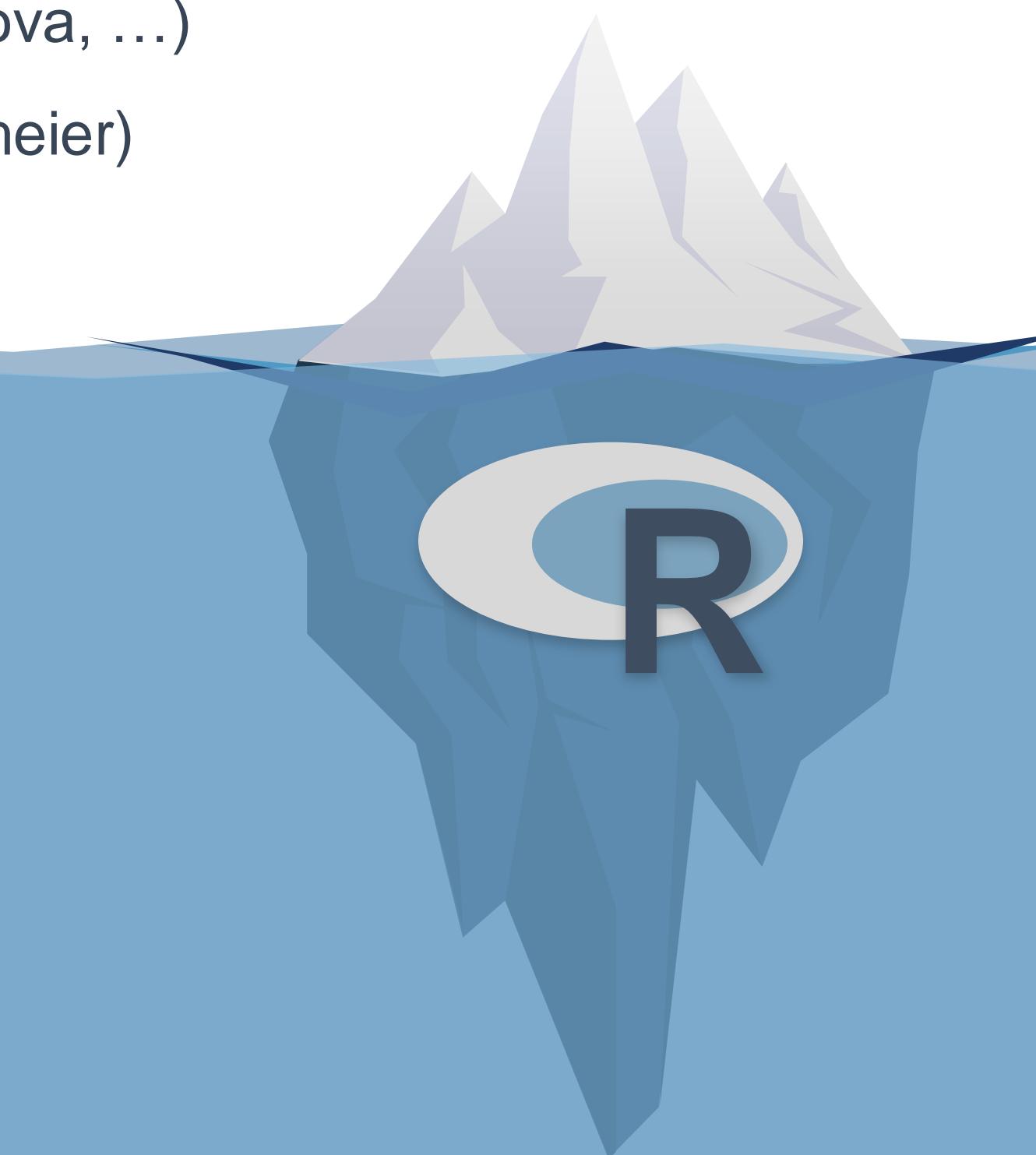
DATA MANGEMENT



EASY PLOTTING



BIOINFORMATIC ANALYSIS



— FROM EXCEL TO R

WANT MO-R-E?



The Section for Biostatistics offers a number statistics-oriented R courses:

Spring

[Basic Statistics for Health Researchers \(Danish course\)](#)

ECTS: 9,0

[Epidemiological methods in medical research](#)

ECTS: 7

[Advanced topics in health research B](#)

ECTS: 2,8

[Statistical methods in bioinformatics](#)

ECTS: 3,5

[Statistical analysis of survival data](#)

ECTS: 4,9

<https://publichealth.ku.dk/about-the-department/biostat/>

[Programming and statistical modelling in R](#)

ECTS: 1,6

[Bayesian methods in biomedical research](#)

ECTS: 2,4

[Psychometric validation of patient reported outcome measures](#)

ECTS: 2,5

[Introduction to validation of patient reported outcome measures](#)

[Causal inference I](#)

ECTS: 2,5

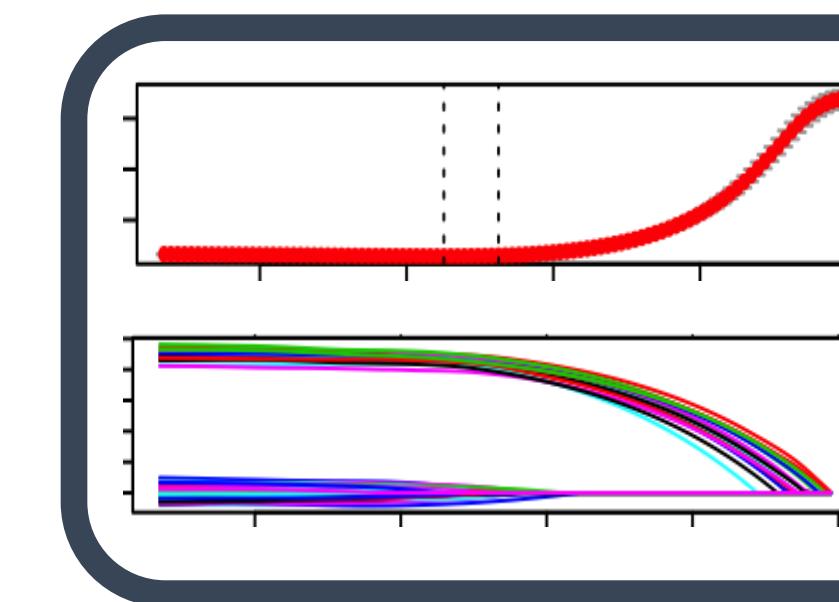
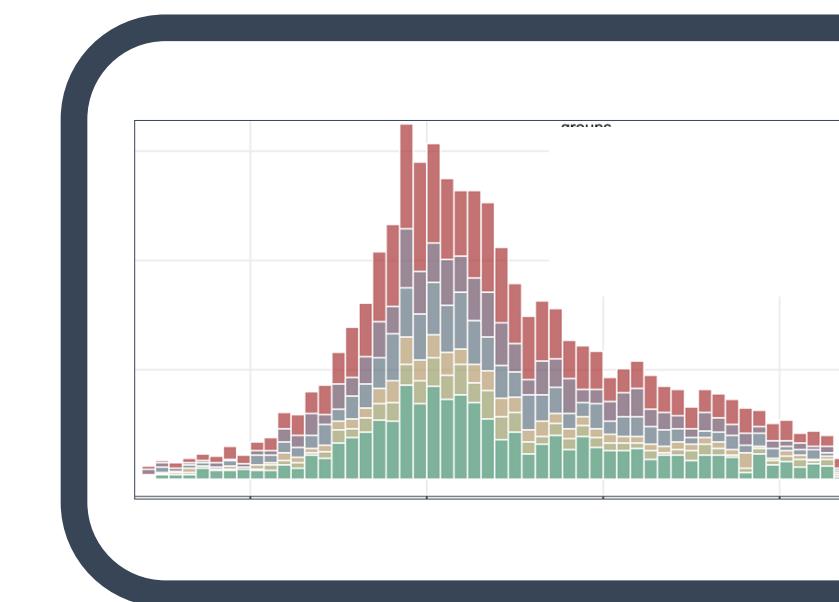
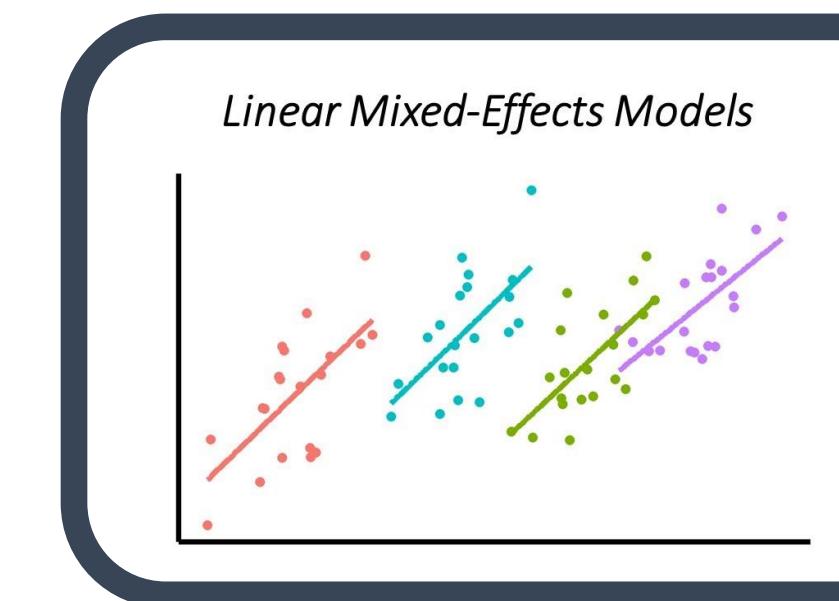
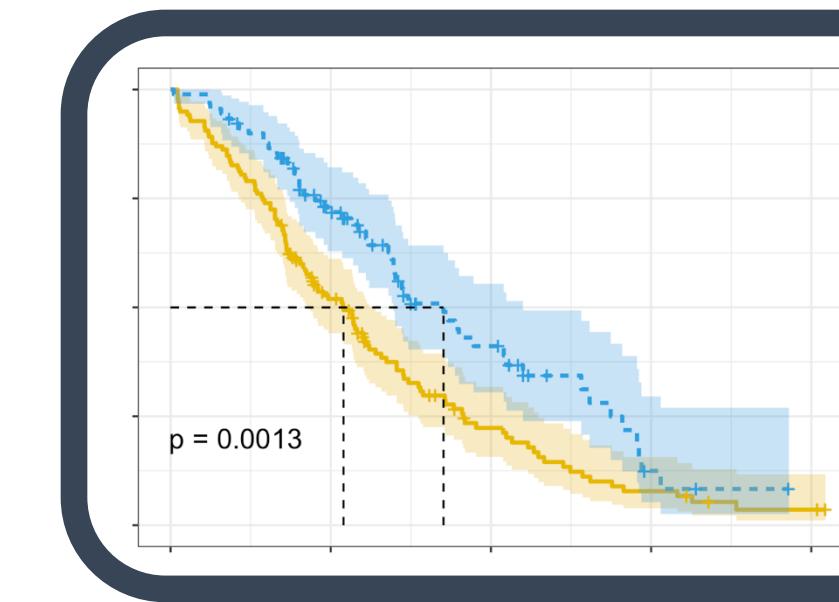
[Use of the statistical software R](#)

ECTS: 2,1

* These are screenshots. Go to the website and scroll down to 'Teaching'



— TEASER STATISTICS in R



Survival Analysis

survival: <https://rviews.rstudio.com/2017/09/25/survival-analysis-with-r/>
survminer: <https://cran.r-project.org/web/packages/survminer/survminer.pdf>
(<https://rpkgs.datanovia.com/survminer/>)

Mixed-Effects Models

lme4: <https://cran.r-project.org/web/packages/lme4/vignettes/lmer.pdf>
<https://cran.microsoft.com/snapshot/2017-08-01/web/packages/sjPlot/vignettes/sjplmer.html>
glmmTMB: <https://cran.r-project.org/web/packages/glmmTMB/index.html>

Epidemiological Analysis

Epi: <https://cran.r-project.org/web/packages/Epi/index.html>
pubh: <https://rviews.rstudio.com/2020/03/05/covid-19-epidemiology-with-r/>
https://cran.r-project.org/web/packages/incidence/vignettes/customize_plot.html
<https://rviews.rstudio.com/2020/03/05/covid-19-epidemiology-with-r/>

Elastic-Net Regression

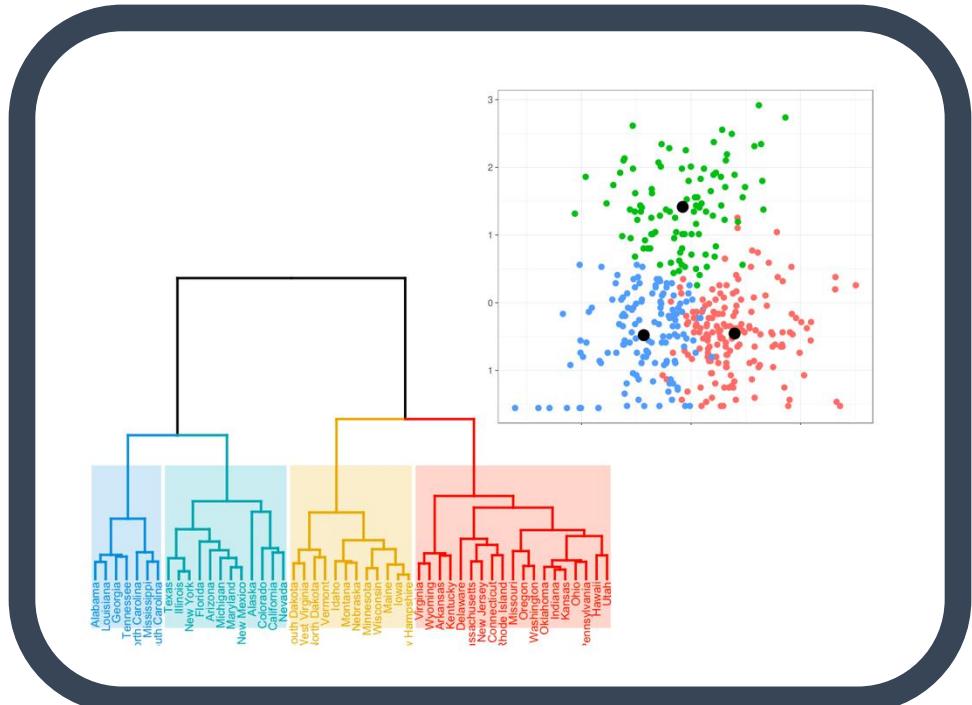
(R
glmnet: <https://cran.r-project.org/web/packages/glmnet/glmnet.pdf>
elasticnet: <https://cran.r-project.org/web/packages/elasticnet/elasticnet.pdf>
<https://www.datacamp.com/community/tutorials/tutorial-ridge-lasso-elastic-net>

TEASER

Machine Learning

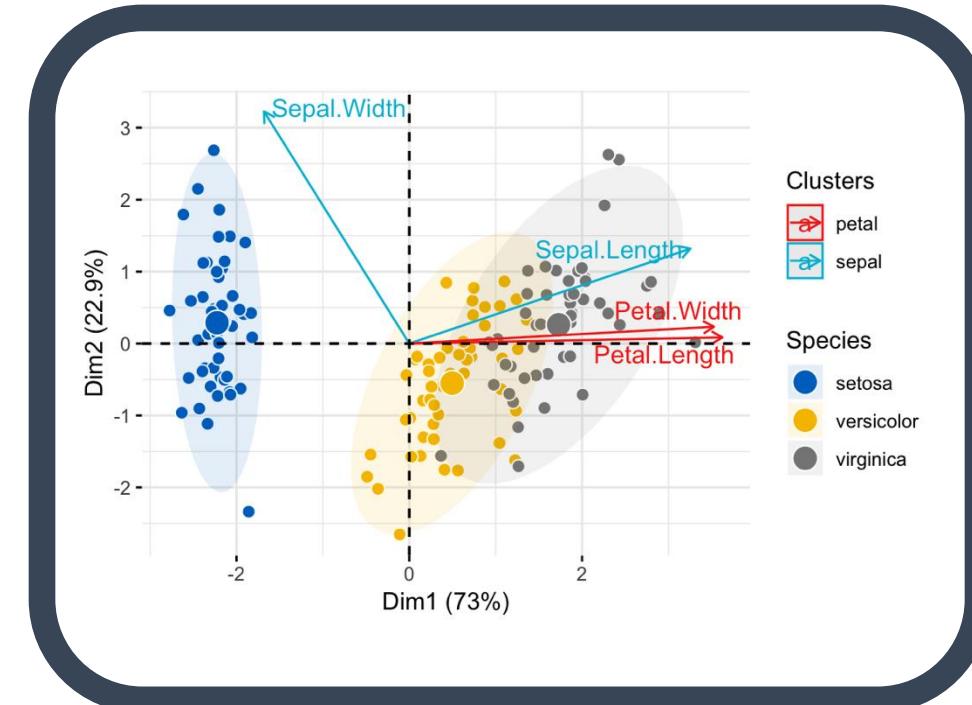
<https://lgatto.github.io/IntroMachineLearningWithR/an-introduction-to-machine-learning-with-r.html>

Clustering



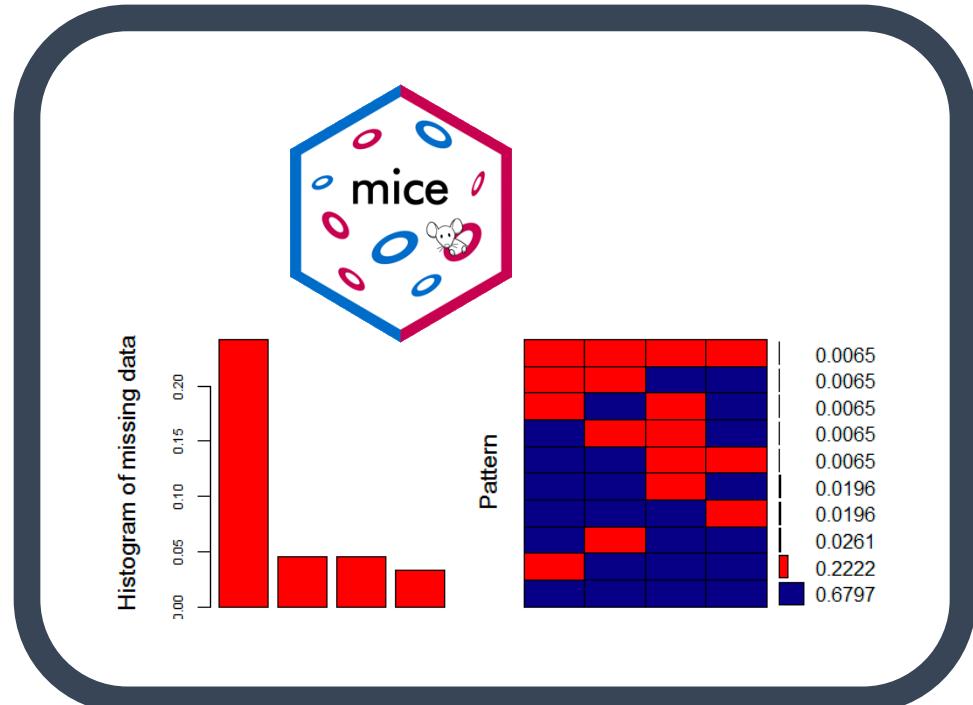
<https://statsandr.com/blog/clustering-analysis-k-means-and-hierarchical-clustering-by-hand-and-in-r/>

Feature Selection: PCA



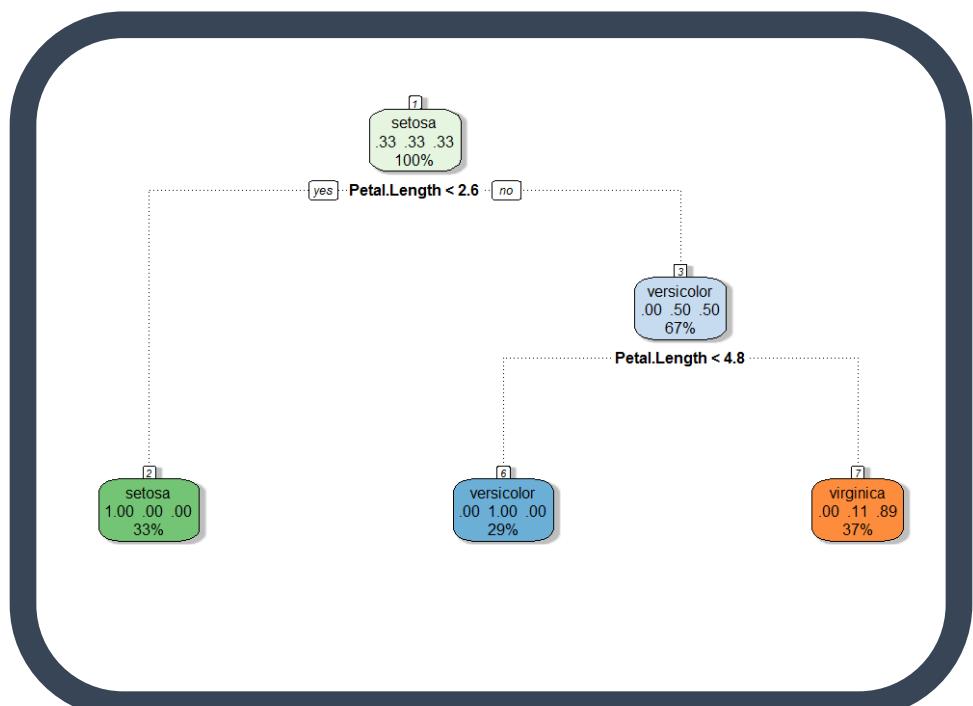
<https://bioconductor.org/packages/release/bioc/vignettes/PCAtools/inst/doc/PCAtools.html>

Missing Data



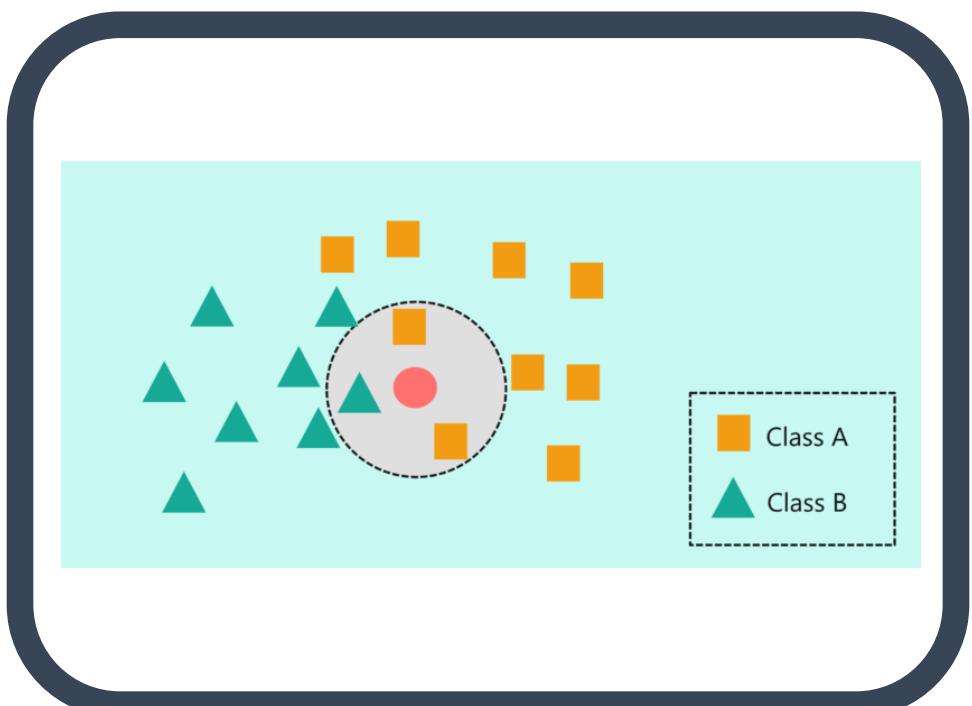
<https://amices.org/mice/>
<https://datascienceplus.com/imputing-missing-data-with-r-mice-package/>

Random Forest



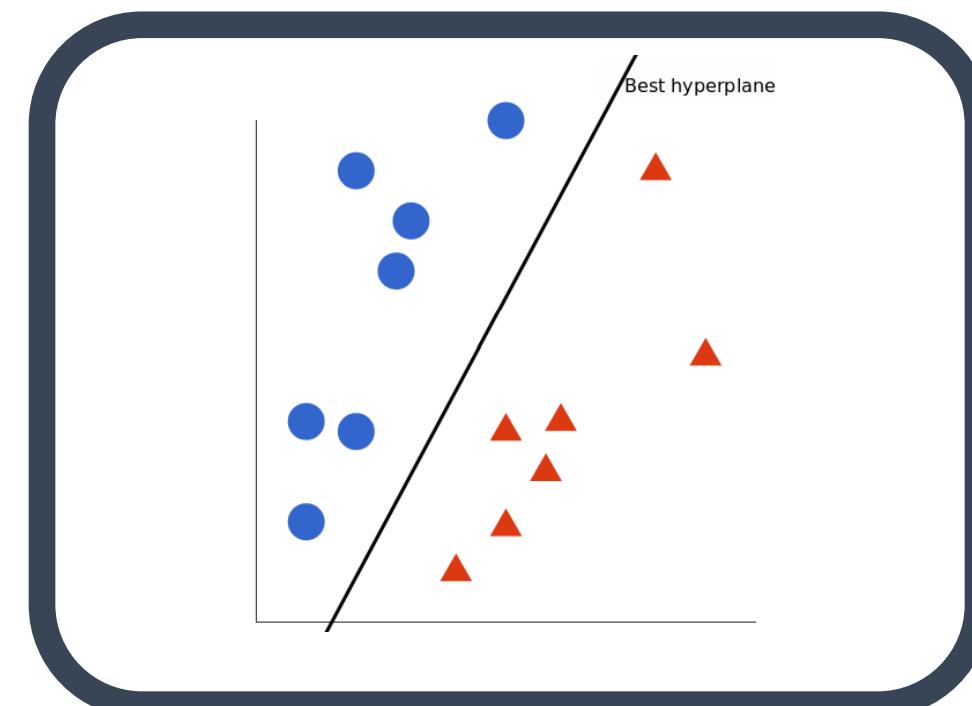
<https://www.blopig.com/blog/2017/04/a-very-basic-introduction-to-random-forests-using-r/>

kNN



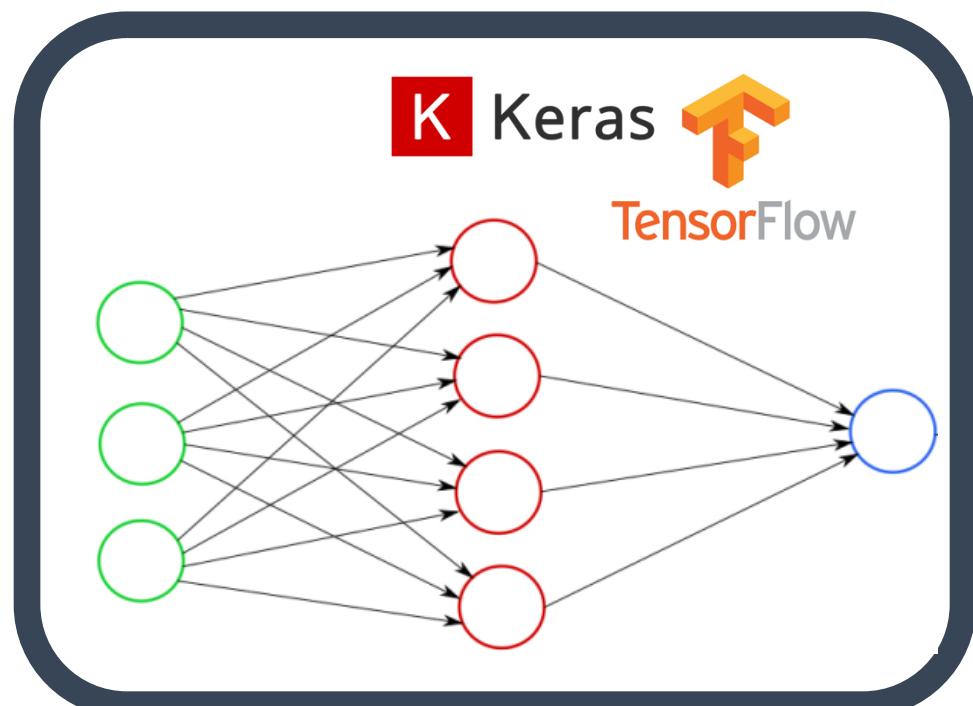
<https://www.edureka.co/blog/knn-algorithm-in-r/>

SVM



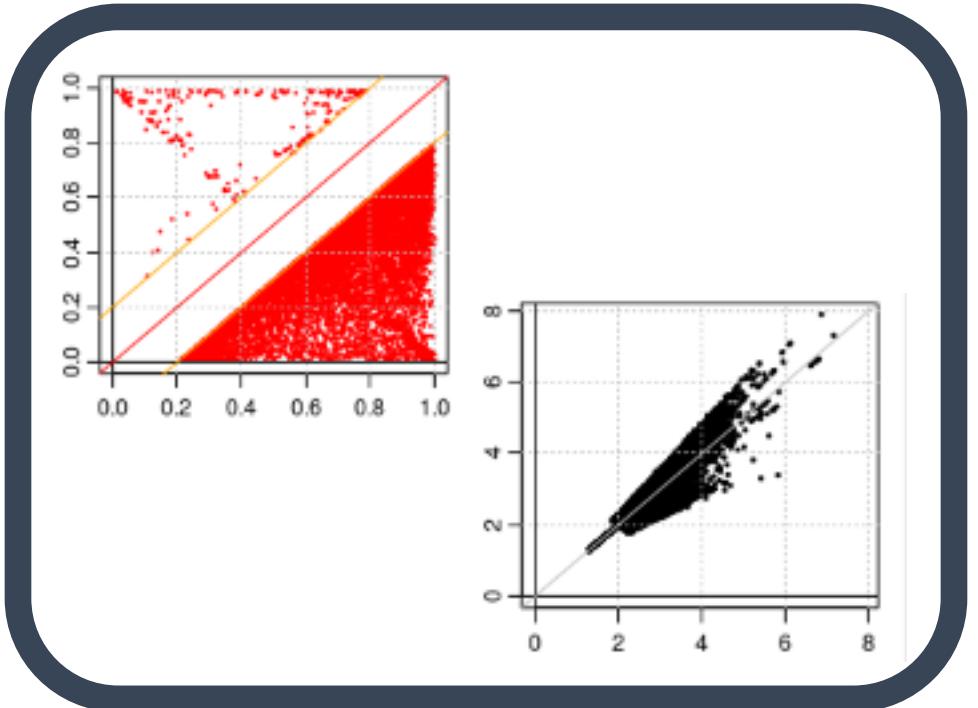
<https://cran.r-project.org/web/packages/e1071/vignettes/svmdoc.pdf>

Neural Networks



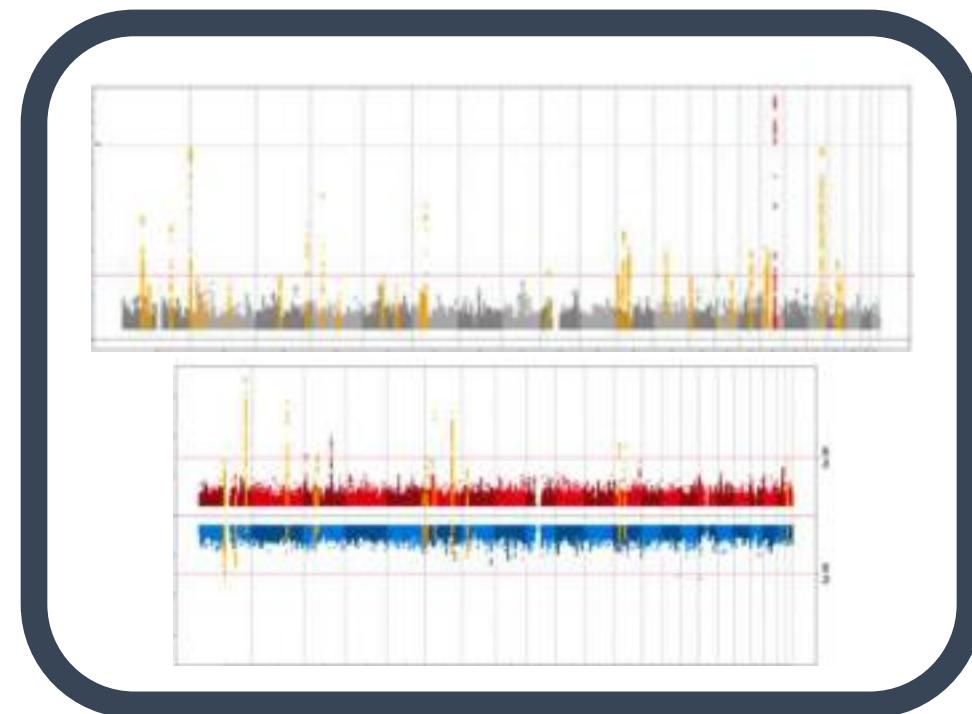
<https://keras.rstudio.com/>
[https://tensorflow.rstuio.com/](https://tensorflow.rstudio.com/)

GWAS - QC & Data Harmonization



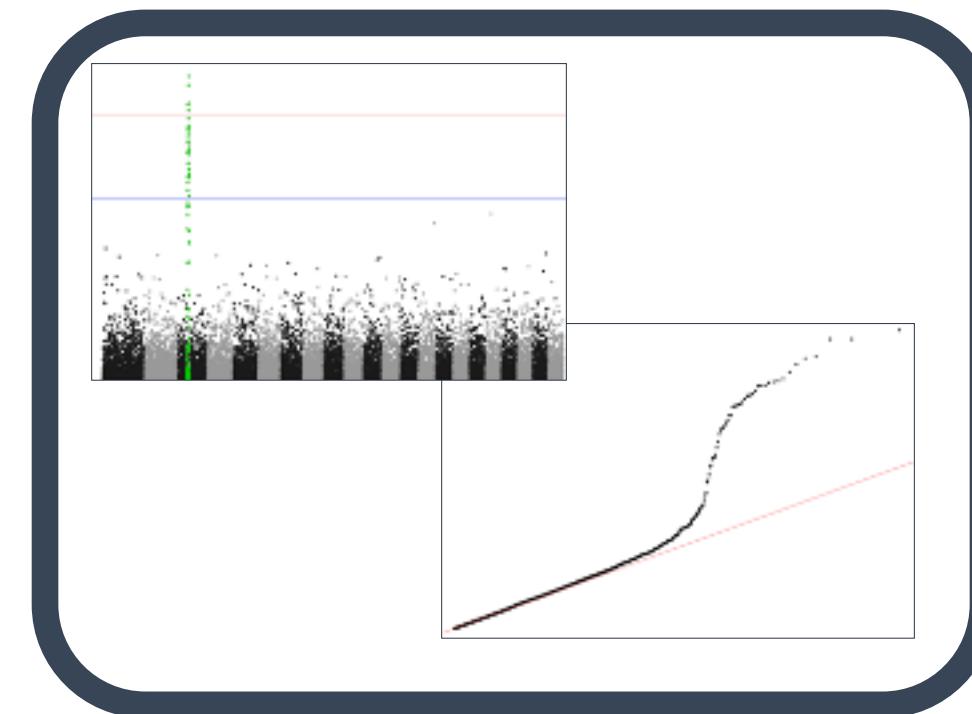
EasyQC: <https://www.uni-regensburg.de/medizin/epidemiologie-praeventivmedizin/genetische-epidemiologie/software/>

GWAS Data Management & Plots



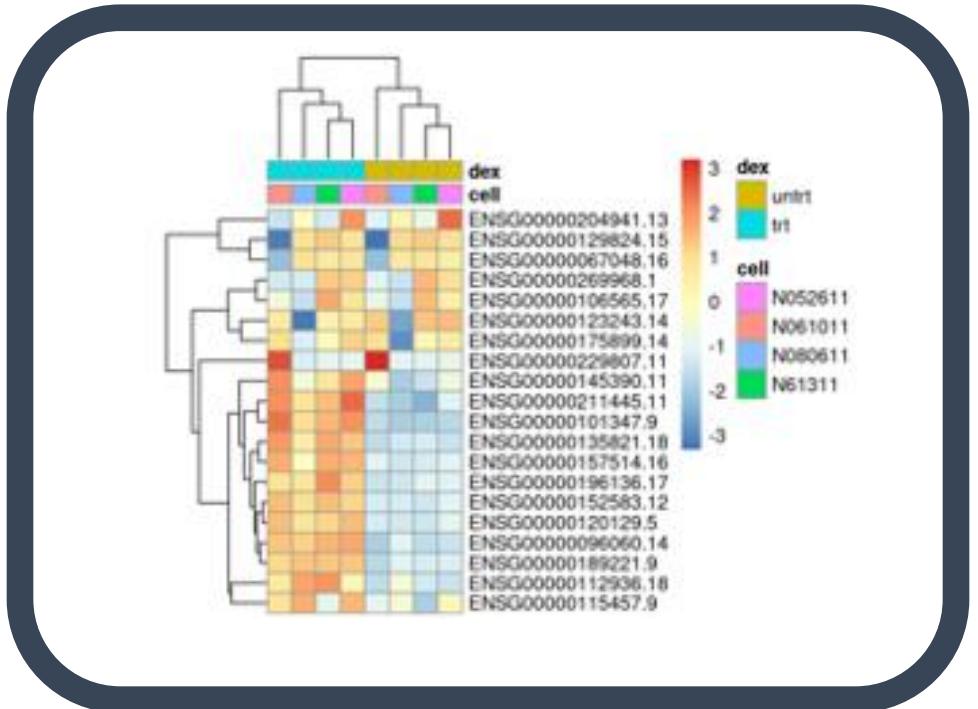
EasyStrata: <https://www.uni-regensburg.de/medizin/epidemiologie-praeventivmedizin/genetische-epidemiologie/software/>

More Plotting...



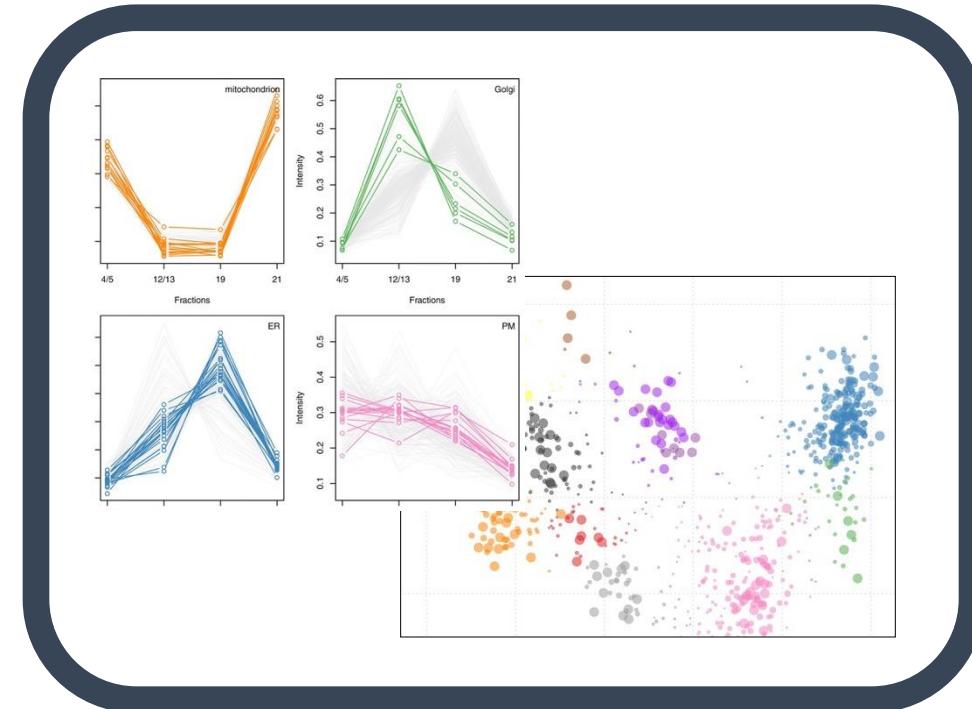
Manhattan and QQ plots:
<https://cran.r-project.org/web/packages/qqman/vignettes/qqman.html>

Gene Expression Analysis



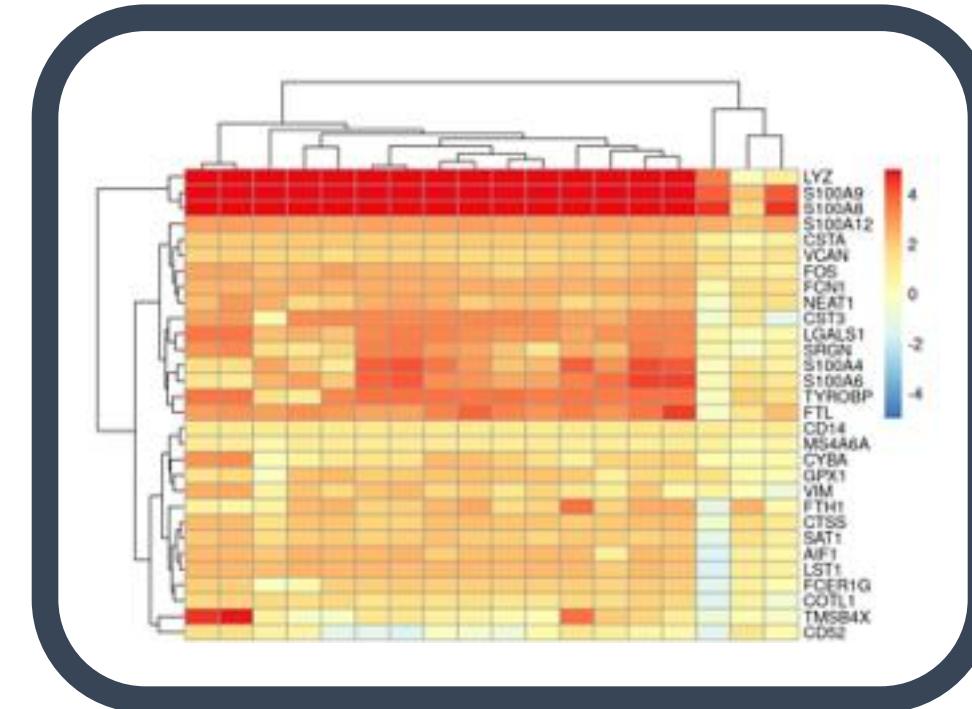
DESeq2, limma, EdgeR, etc.:
http://www.bioconductor.org/packages/release/BiocViews.html#_RNASeq

Proteomics Analysis



RforProteomics:
http://www.bioconductor.org/packages/release/BiocViews.html#__Proteomics_RforProteomics.html

Single-Cell RNASeq



<https://cran.r-project.org/web/packages/e1071/vignettes/svmdoc.pdf>

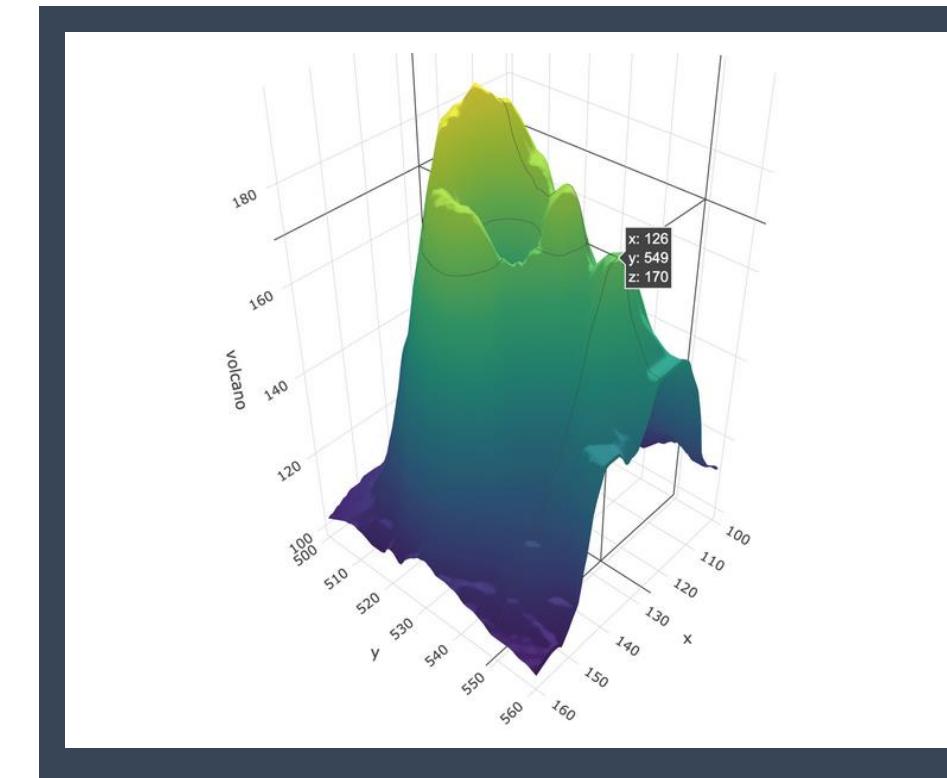
— TEASER Omics Data

<http://www.bioconductor.org/packages/release/BiocViews.html>

COOL STUFF IN R

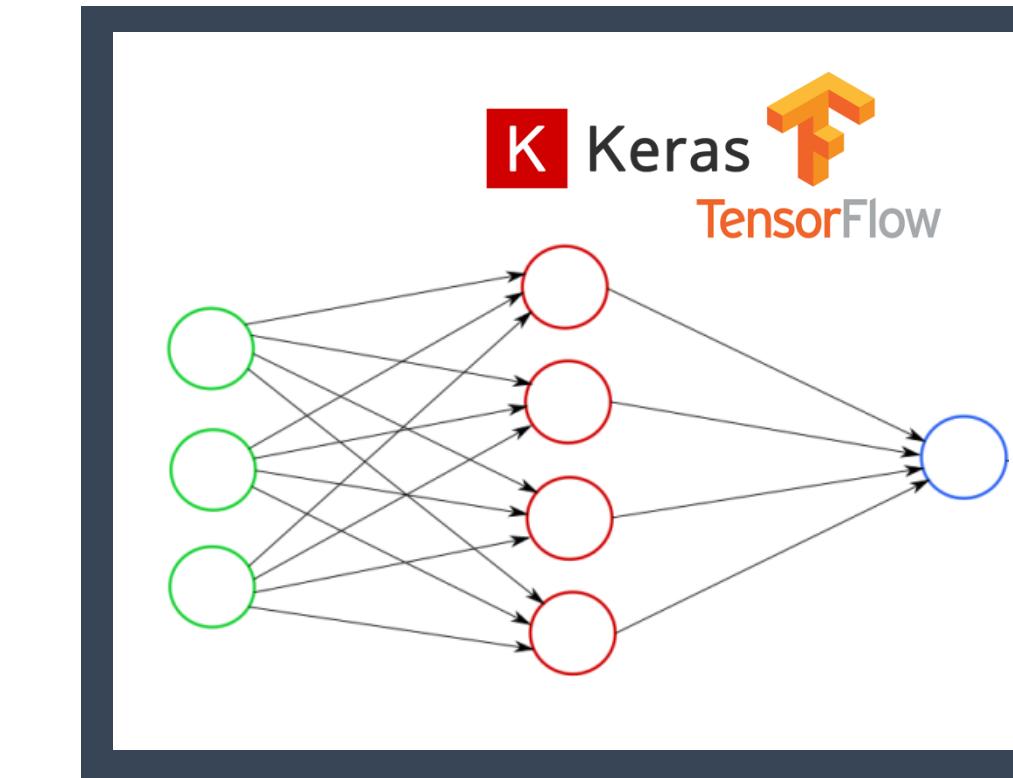
— FROM EXCEL TO R

PLOTTING IN 3D



<https://plotly-r.com/d-charts.html>

DEEP LEARNING



<https://keras.rstudio.com/>
<https://tensorflow.rstudio.com/>

BAYESIAN STATISTICS



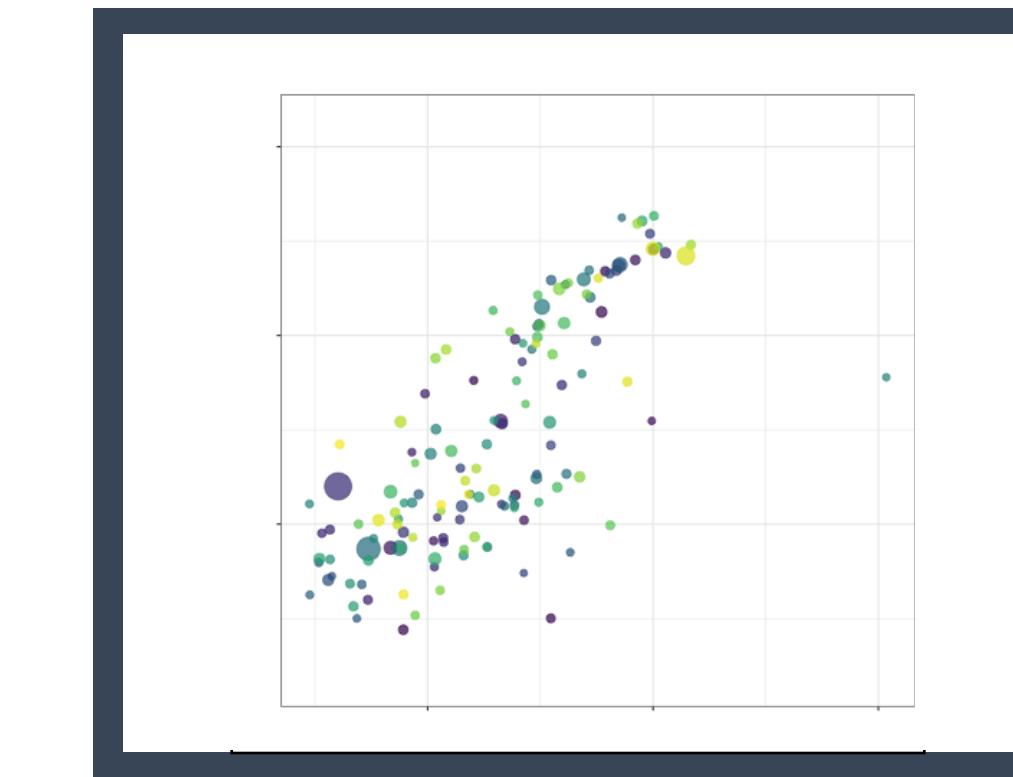
<https://mc-stan.org/users/interfaces/rstan>

WEBPAGE WITH R SHINY



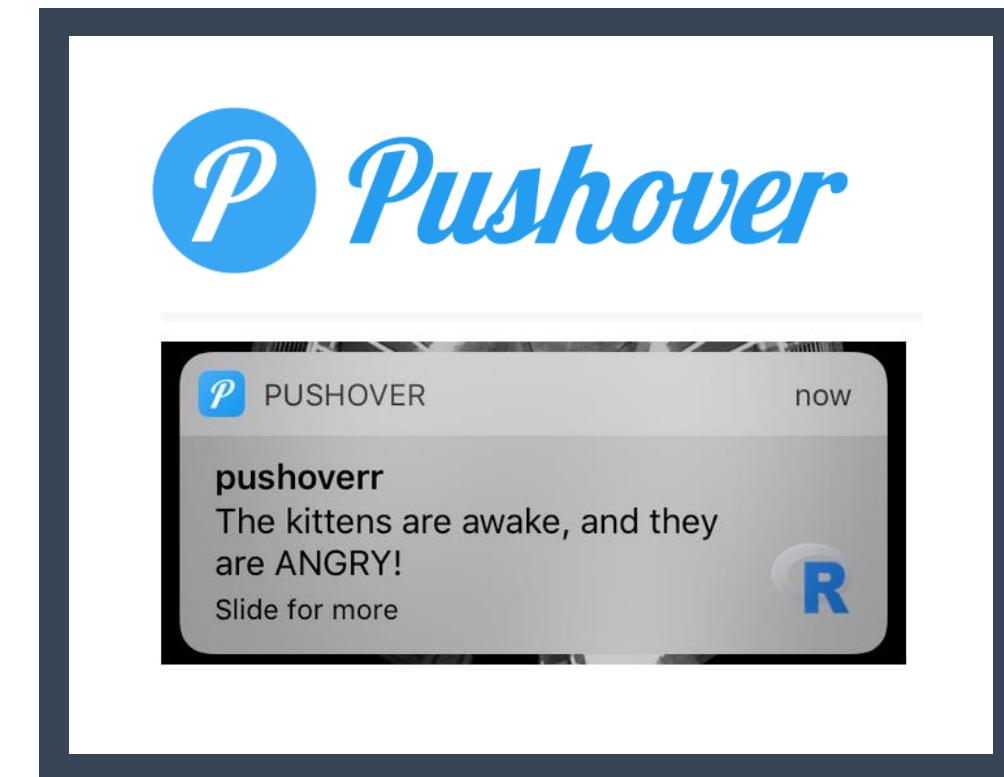
<https://shiny.rstudio.com/>

INTERACTIVE PLOTS



<https://gganimate.com/articles/gganimate.html>

MAIL AND MESSAGES



<https://github.com/briandconnelly/pushoverr>

THANK YOU FOR LISTENING



This keynote presentation was created by Thilde Terkelsen,
Data Scientist, Center for Health Data Science, SUND, KU.
For internal use at KU only, do not distribute commercially.