



Getting Started with R

Enrico Toffalini

PSICOSTAT

Why becoming an R user?

- **Mainstream in academia** for data science, increasingly used in business. *Job market advantage!*
- **Free & open-source**: wherever you go, R will be with you at no costs (unlike *MPLUS, MATLAB, SPSS*, etc.)
- **Real programming language**: difficult at the beginning, but: 1) gives you lots of flexibility; 2) has transfer on other programming languages (e.g., *Python*).
- **Vast community support** thanks to a large and active community (also, *chatGPT, Lucrez-IA*, etc., know it pretty well!).
- **Huge ecosystem**, >20,000 packages on CRAN, more from other sources (e.g., GitHub), to do amazing stuff with statistical data analysis, machine learning, data visualization, developing webapps [*shiny*], writing reports and even entire books [*bookdown, rmarkdown*]), can integrate with *quarto, github*.
- Facilitates **reproducible research** by sharing code and workflows.

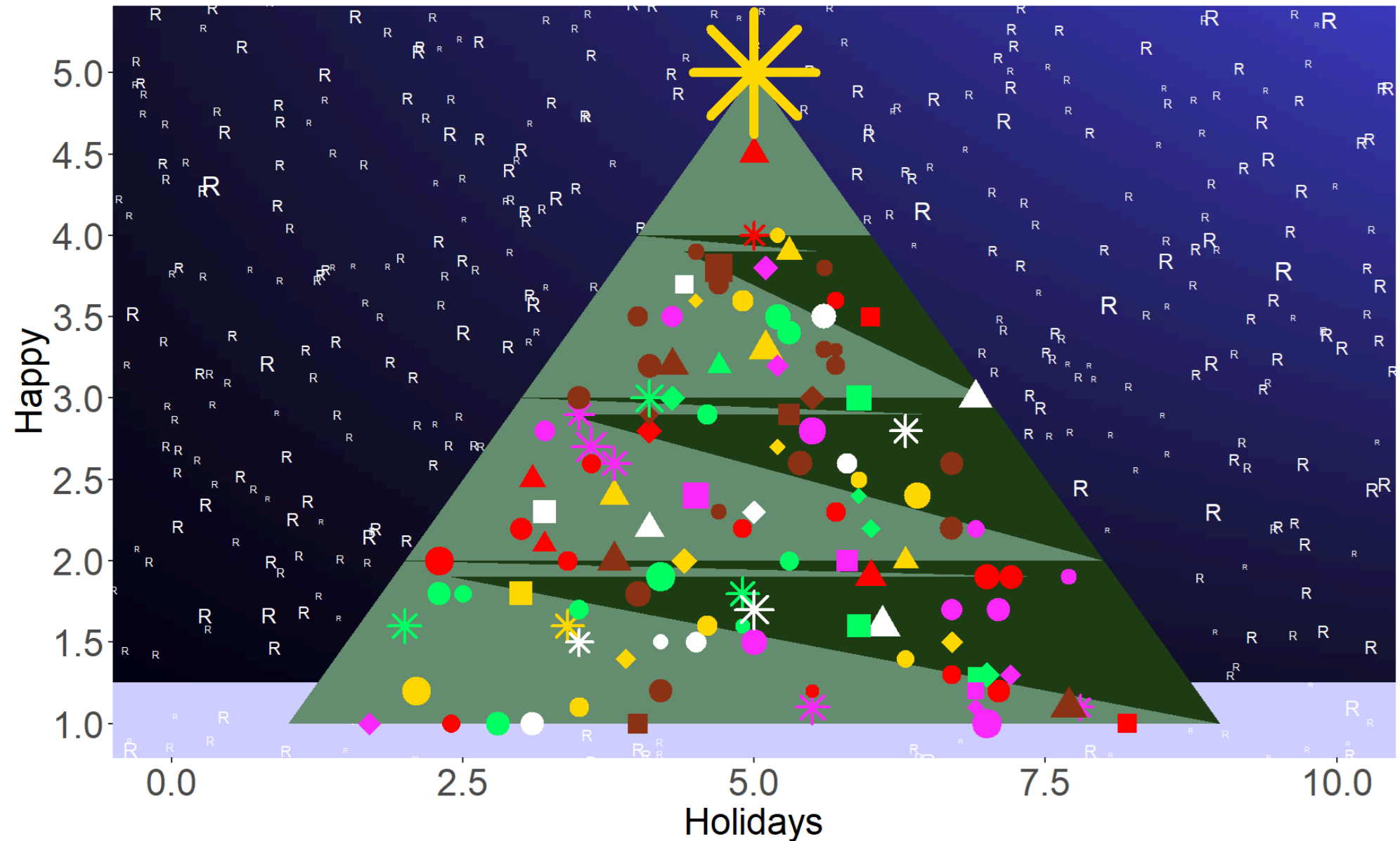
What you may expect to learn in this course:

- Executing fundamental operations and using basic functions;
- Working with essential data types and structures;
- Gaining some proficiency in managing and manipulating data with vectors and dataframes;
- Understanding some fundamental concepts of programming.

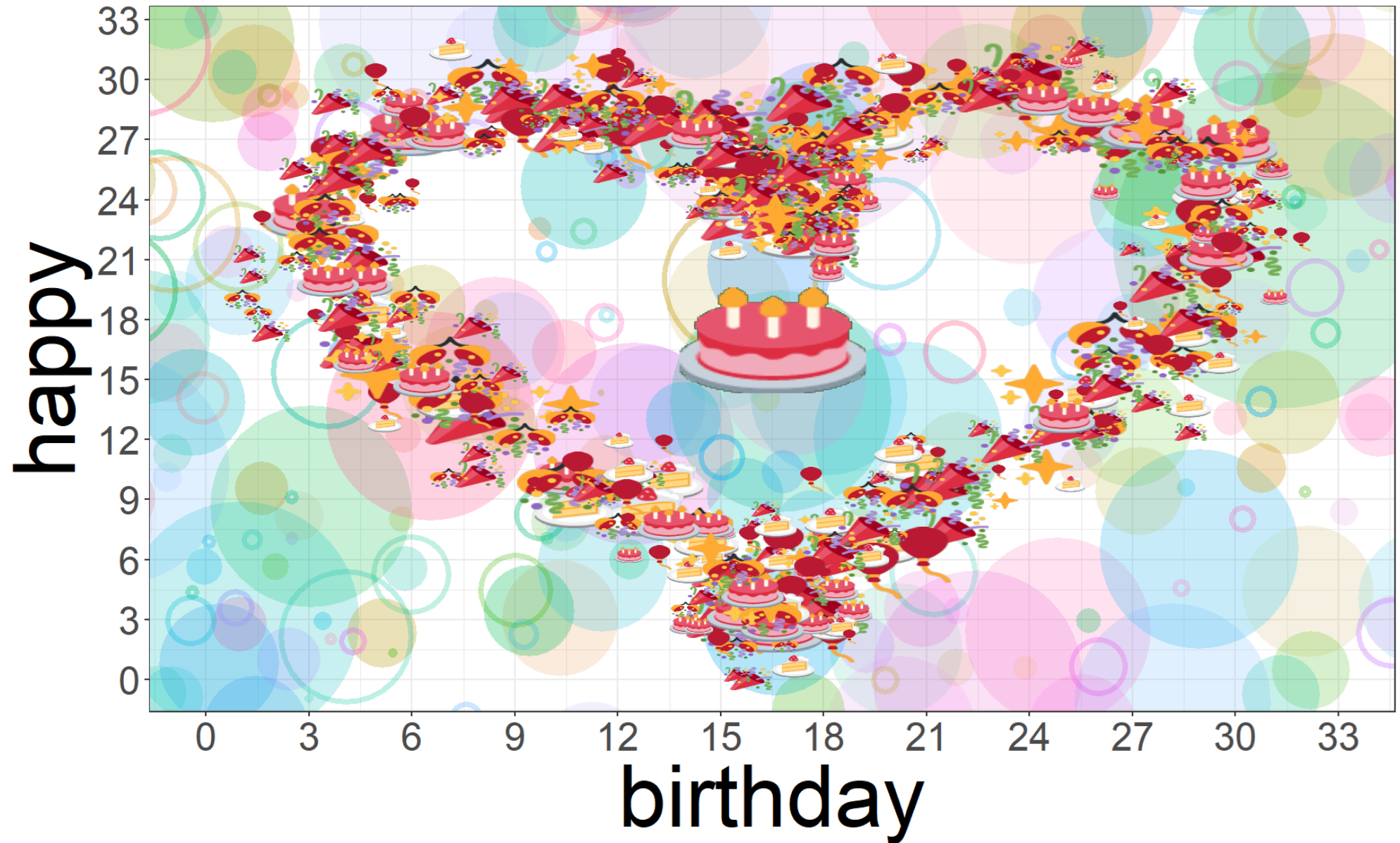
Over the next *couple of years*, following this PhD program, you may have the opportunity to learn to use R to perform at least some fundamentals about:

- Core statistical inference methods;
- (Generalized) linear (mixed-effects) modeling;
- Data visualization using *ggplot2*;
- Power analysis via data simulation ;
- Structural Equation Modeling (SEM);
- Conducting Meta-Analysis.

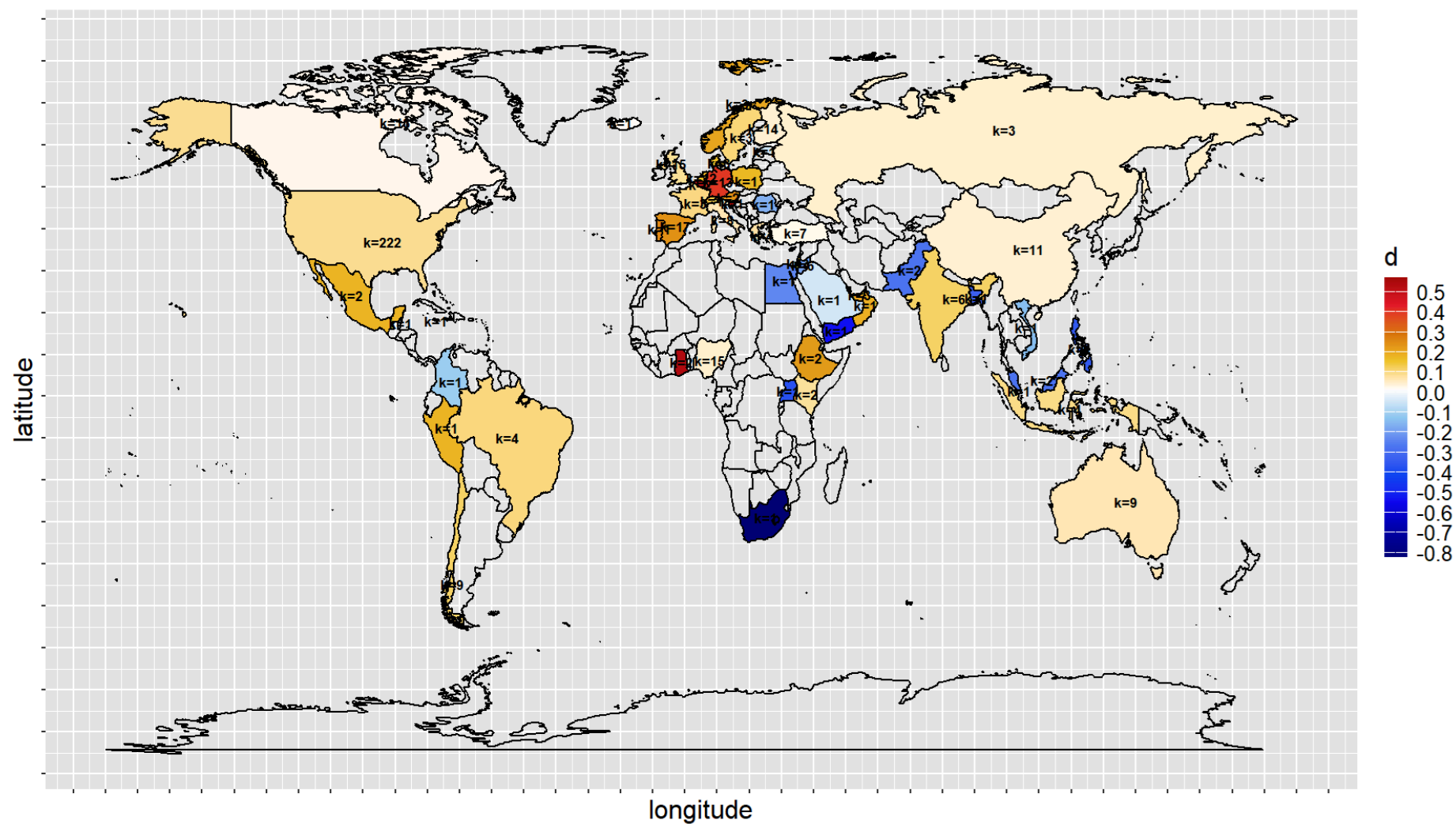
you may even create greeting cards



you may even create greeting cards

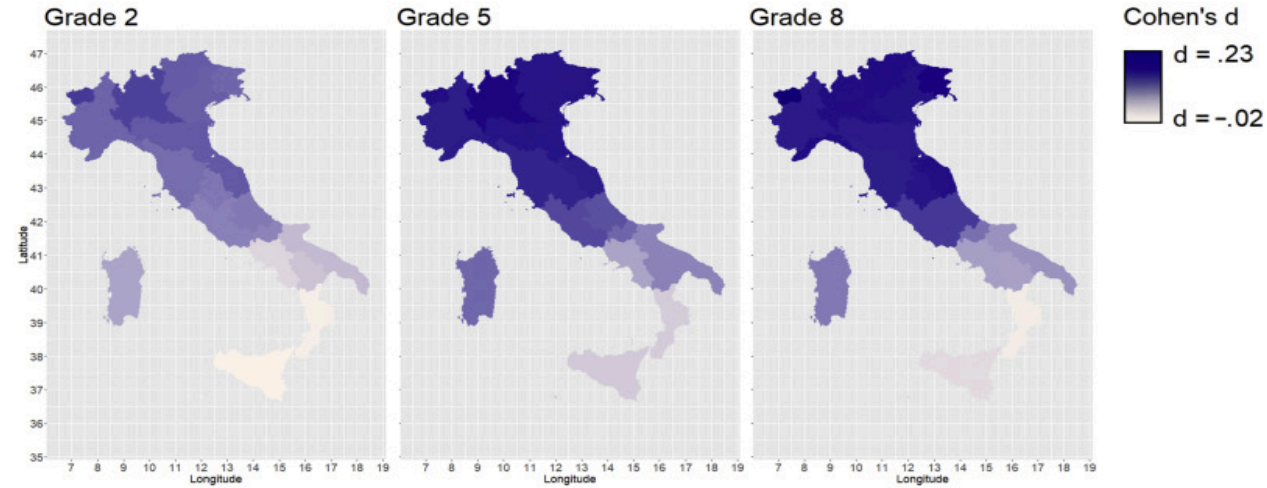


or like fancy infographics

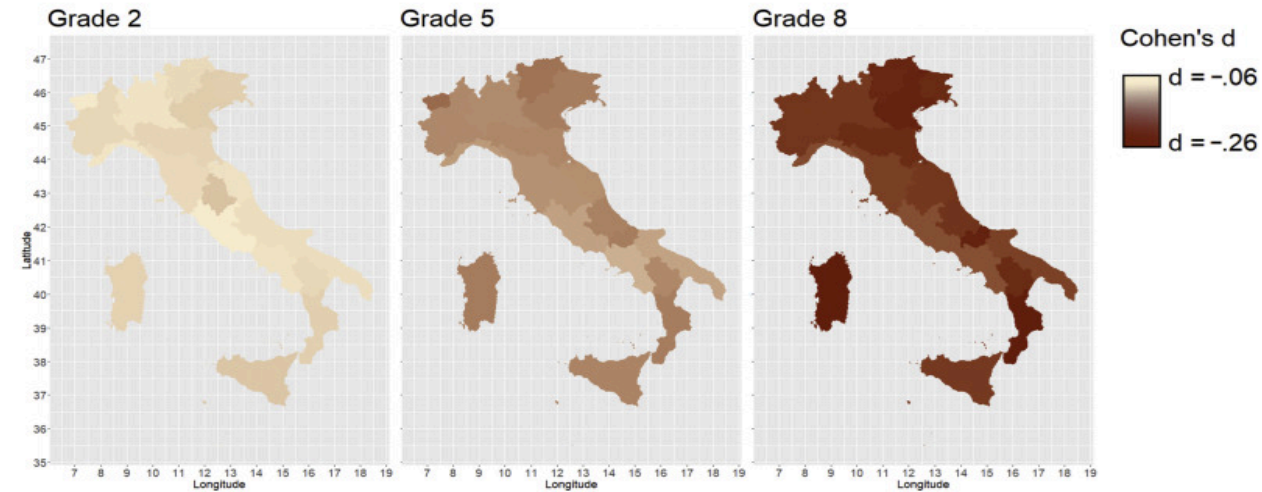


or like fancy infographics

A.

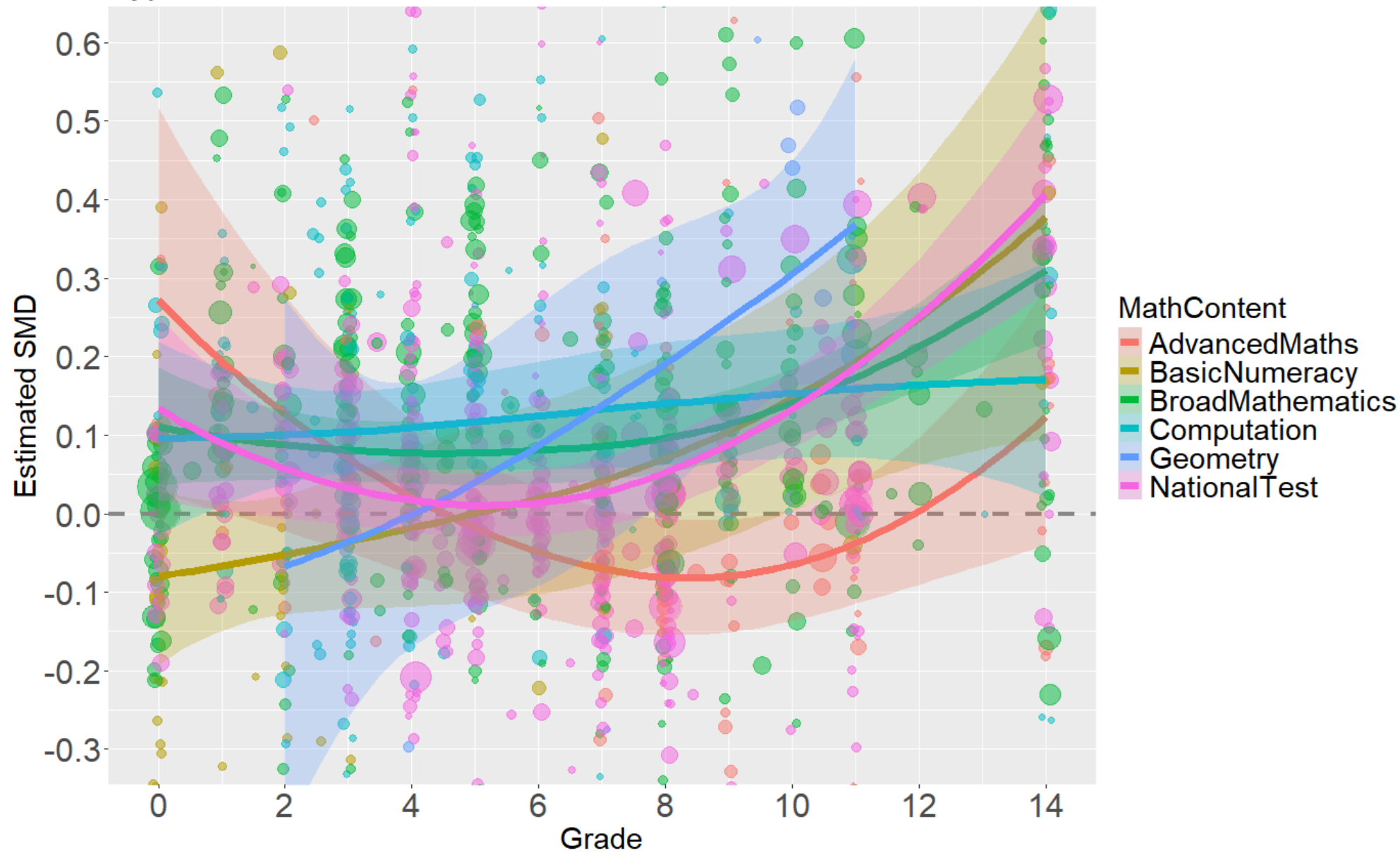


B.



perform classical data analysis

Type of content x Grade interaction



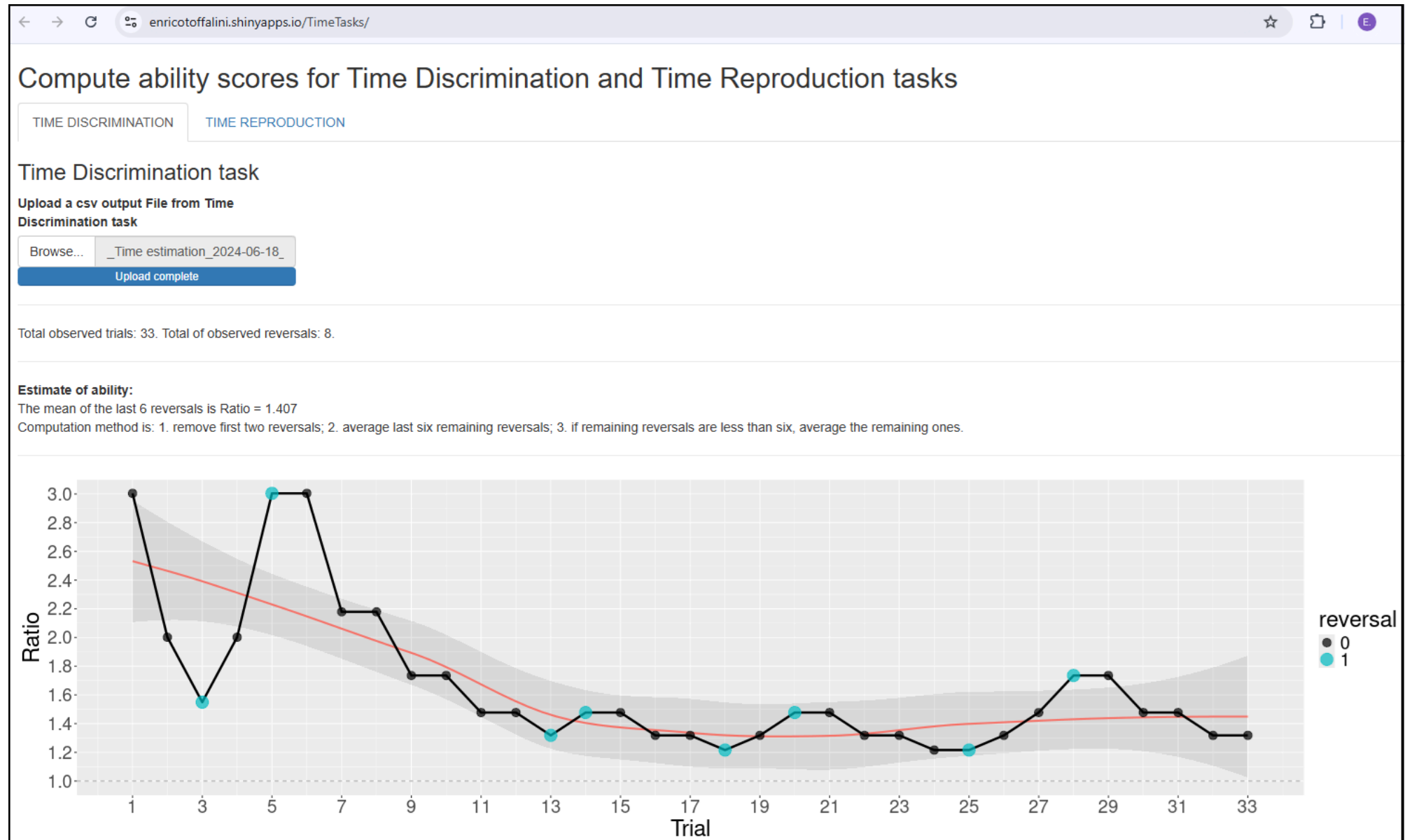
you may create interactive webapps with Shiny

see [Shiny gallery](#)

here's a couple of recent real examples from **Psicostat** members:

- this [game-like shiny app](#) developed for the science4all event in Padova; see [here](#) some explanation in Italian
- practical [ad-hoc shiny app](#) for scoring experimental data collected by students

you may create interactive webapps with Shiny



or entire websites and books

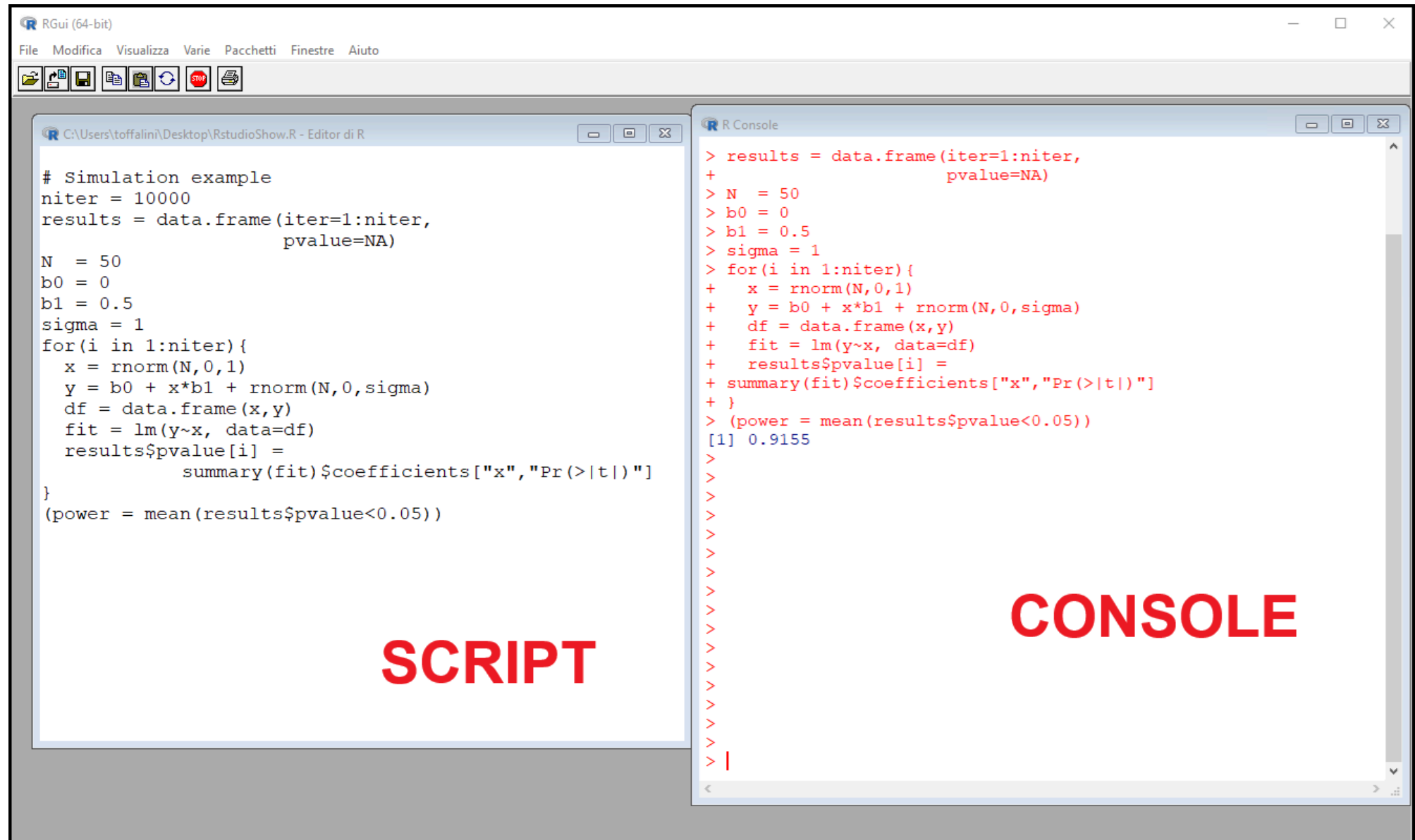
- this **entire course** is a website in its own right
- the **course textbook** is a book/website
- also see **this wonderful book** by Daniël Lakens explaining Statistical Inference

of course, these resources integrate other tools such as GitHub and Quarto, but they can be created within the R ecosystem

install R and Rstudio

first of all, for getting started, follow the instructions in *Chapter 1* of *Introduction2R* to ensure that both R and RStudio are installed

R Console (just base R)



R Studio (full IDE)

The screenshot displays the R Studio IDE interface with four main panes:

- SCRIPT**: The top-left pane shows R code for a simulation example. The code defines parameters (N=50, b0=0, b1=0.5, sigma=1, niter=10000) and performs a linear regression simulation using `lm()` and `summary()` to calculate the power of the test.
- CONSOLE**: The top-right pane shows the output of the R code, including the calculated power (0.918) and the results of the simulation.
- ENVIRONMENT**: The bottom-left pane shows the current environment with variables `df` (50 obs. of 2 variables), `fit` (List of 12), and `results` (10000 obs. of 2 variables). The `Values` section shows the values of `b0` (0), `b1` (0.5), `i` (10000L), `N` (50), `niter` (10000), `power` (0.9105), and `df` (1).
- FILE EXPLORER, ETC.**: The bottom-right pane shows the file explorer with a list of files in the current directory: `faviconpsicostat2.png` (79.8 KB, Oct 31, 2024, 10:25 AM), `psicostatLogo.png` (264 KB, Mar 4, 2024, 2:05 PM), and `RstudioShow.R` (378 B, Oct 31, 2024, 5:08 PM).

The R Studio interface includes a menu bar (File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help) and a toolbar with icons for running code, saving files, and other IDE functions. The status bar at the bottom shows the current file path and the time/date.

Some R packages that you will or may need in the future (1/3)

| Package | Used for what | Example(s) of functions |
|-----------------------------------|---|--|
| <code>base</code> (base R) | Basic functions | <code>sum</code> , <code>mean</code> , <code>sqrt</code> , <code>abs</code> , <code>c</code> , <code>data.frame</code> , <code>summary</code> , <code>scale</code> , <code>plot</code> , <code>+</code> , <code>-</code> |
| <code>stats</code> (base R) | Basic statistical calculations and functions | <code>sd</code> , <code>cor</code> , <code>cor.test</code> , <code>t.test</code> , <code>lm</code> , <code>glm</code> , <code>AIC</code> , <code>rnorm</code> , <code>rbinom</code> |
| <code>graphics</code> (base R) | Basic statistical calculations and functions | <code>boxplot</code> , <code>hist</code> , <code>barplot</code> |
| <code>effectsize</code> | Compute different effect sizes | <code>cohens_d</code> , <code>hedges_g</code> , <code>cohens_f</code> , <code>d_to_r</code> |

Some R packages that you will or may need in the future (2/3)

| Package | Used for what | Example(s) of functions |
|--------------------------|---|--|
| <code>lme4</code> | Fitting (generalized) (non-)linear mixed-effects models | <code>lmer</code> , <code>glmer</code> , <code>ranef</code> |
| <code>performance</code> | Useful tools for models | <code>check_collinearity</code> , <code>r2_nagelkerke</code> , <code>icc</code> |
| <code>effects</code> | Display effects for various statistical models | <code>allEffects</code> |
| <code>emmeans</code> | Estimate marginal means for various models | <code>emmeans</code> |
| <code>ggplot2</code> | Create beautiful plots using The Grammar of Graphics | <code>ggplot</code> , <code>geom_*</code> |

Some R packages that you will or may need in the future (3/3)

| Package | Used for what | Example(s) of functions |
|-----------------------|---|---|
| <code>lavaan</code> | Structural equation modeling (SEM) | <code>cfa</code> , <code>sem</code> |
| <code>semTools</code> | Useful tools for SEMs | <code>reliability</code> |
| <code>metafor</code> | Perform meta-analysis | <code>rma</code> , <code>rma.mv</code> , <code>forest</code> , <code>funnel</code> , <code>regtest</code> |
| <code>brms</code> | Fitting practically any Bayesian model via MCMC with STAN | <code>brm</code> , <code>prior</code> |
| <code>blavaan</code> | Fitting Bayesian SEMs | <code>bcfa</code> , <code>bsem</code> |

Let's Test the Environment!

Let's run a few commands in RStudio to familiarize with its console and see if the installation works properly

```
rnorm(10) # draw 10 random values from a Standard Normal distribution
```

```
[1]  0.80615400 -0.58547491  1.10525657 -0.40379478 -0.98313925  0.06539615  
[7]  0.81912540  1.60196277  2.44925450 -0.44644926
```

```
?rnorm # open the help tab for the "rnorm" function
```

```
round( rnorm(10, mean=100, sd=15) ) # draw 10 values from IQ distribution
```

```
[1]  92 103 140  99  86  86  99  73  97  83
```

```
install.packages("psych") # install a package from CRAN
```

```
library(psych) # load the newly installed package
```

```
fisherz(rho=0.9) # use it to transform a correlation into a Fisher's z
```

```
[1] 1.472219
```