

MBA
USP
ESALQ

Data Wrangling

Prof. Wilson Tarantin Jr.

Data Preparation in R

Hernandes Matias Junior 124.799.116-40

Data wrangling

- We will mainly use the dplyr.
 - Dplyr is a package contained in the tidyverse
 - It contains useful functions for the manipulation/preparation of databases
 - Material for reference:
 - <https://dplyr.tidyverse.org/>
 - <https://github.com/rstudio/cheatsheets/blob/master/data-transformation.pdf>
 - Wickham, H. & Grolemund, G. **R for Data Science**: <https://r4ds.had.co.nz/index.html>

Data wrangling

- **Pipe:** chaining of several functions in sequence
- **Rename:** change of variable names
- **Mutate:** change of variables content and creation of new variables
- **Filter:** selection of observations based on logical criteria
- **Select:** selection of variables
- **Summarise:** creation of tables with summary statistics (descriptive statistics)
- **Group by:** to group observations based on criteria
- **Join:** to join (*merge*) databases

Projects creation and Scripts R Markdown

Hernandes Matias Junior 124.799.116-40

R Markdown

- Introduction to R Markdown
- Basic formatting of the text
- Formulas insertion
- Chunks
- To generate outputs (HTML; PDF, DOC)
- Material for reference:
 - <https://rmarkdown.rstudio.com/index.html>

Data Science & Analytics Projects in the GitHub

Hernandes Matias Junior 124.799.116-40

Git

- Useful software for version control
- It records changes made in the files
- We will use it in conjunction with the Github
- To install the Git on the computer (<https://git-scm.com/downloads>)
 - It's just to pass all steps in the suggested configurations

Github

- Website used to keep the files
 - <https://github.com/>
- It's organized in repositories (folders) that can be shared, and they can be published.
 - It's useful to store and share portfolio of projects
- Computer files can be sent to Github (via Git)

Git and Github

- Add and Commit
 - Create a folder on the desktop of your computer.
 - In RStudio, create a new scrip and write # Versão 1
 - Save this file in the folder with the name Versão Exemplo.R
 - Inside the folder, click using the mouse right button and choose Git Bash Here
 - In Git, write **git init** (it initializes Git in the selected folder)
 - Write **git add "Versão Exemplo.R"** (it adds the file to the index)
 - Use **git commit -m "título"** to generate versions (it is the versions)

The commit name, example: "First Version"

Git and Github

- Push
 - In your Github, create a new repository and name it as you like.
 - Copy the link from the created repository
 - In Git, write **git remote add origin(folder link).....**
 - Finally, type **git push – u origin master** (it sends the file to the repository, remaining on the main branch)
 - It will request login into Github on the first time
- After update, it is possible to verify that the file is already in your Github!

Git and Github

- Versions creation and comparison
 - Open the file Example Version and write another line: # Versão 2
 - After saving, close it, and open the Git Bash Here in the folder pressing the mouse right button
 - Use the same procedures:
 - **git add “Versão Exemplo.R”**
 - **git commit -m “Segunda Versão”**
 - **git push -u origin master**
 - The new version is already available in Github and we can compare them!

Note that it was not necessary to inform the address

Git and Github

- Creating branches in the repository
 - In the previous command, we change the main branching of the repository
 - We could create new branches in Github
 - **git checkout -b “nome da nova branch”**
 - In Git, there is already the indication of the change from “master” to "new"
 - The same add and commit procedures
 - **git push -u origin “nome da nova branch”**

Git and Github

- Importing repositories (Clone and Pull)
 - It can be useful to save files on your computer that are in the Github
 - A way of “downloading” these files is through the clone function
 - Create a folder on your computer
 - Inside the folder, open the Git Bash Here pressing the mouse right button
 - In the Github, click on **code** and copy the link in the repository chosen
 - In Git, type **git clone(repository link).....**
 - After changes in Github, indicate **cd “repository” to** download again
 - Then, type **git pull** (the file was updated on the computer)

Git and Github

- Copying public repository (Fork)
 - It is possible to copy repositories that are published in the Github
 - Search for some theme of interest
 - Access the repository
 - On the top right corner, there is the **Fork** button
 - After clicking it, you can see the repository on your list (on your profile)

Git, Github and RStudio

- It is possible to integrate Git, Github and RStudio
- In RStudio, click on File → New Project → Version Control → Git
 - In “Repository URL” indicate the link of the repository in Github
- After creating a document (R Script, R Markdown), click on Git and make **commit** and then **push**
 - It is also possible create **pull** of the files of the repository that was indicated

Functions and Iterations with Purrr Package

Hernandes Matias Junior 124.799.116-40

Functions, Purrr

- **Creating functions in R**
- **Assign conditions (“IF”)**
- **Iterations with Purrr (map functions)**
- **Material for reference:**
 - Wickham, H. & Grolemund, G. **R for Data Science**: [Wtps://r4ds.had.coz/index.html](https://r4ds.had.co.nz/index.html)
 - <https://github.com/rstudio/cheatsheets/blob/master/purrr.pdf>