Introduction to R and the tidyverse

– tools of the trade –

Paolo Crosetto

Tools

Tools of the trade

- **Book:** R for Data Science 2d edition (free)
- **R editor:** Rstudio (free)
- **Packages:** tidyverse (free)
- **Hosting** and **version control:** gitlab (free)
- **Data**: freely available data + eurostat + tidytuesday (free)

> bottom line, all tools here are `free`.

Why R?

Your R background

> **How many of you have already used …?**

- Excel
- SAS, SPSS
- Stata
- Base R
- Tidyverse R
- Python, Jupyter, Julia, C, C++…

Why R?

- it’s `free` (as in *free beer* and also as in *free speech*)
- it has a *huge* code and user `base`
- it is a programming language *built for doing statistics*
- you have to *understand what you are doing*
- it forces you to use `scripts` -> reproducible research
- Increasingly used in `industry` & `data science`

Why not Excel? [a short rant]

> *Excel is bad for* `reproducible` *work*

- click-based interface is a source of horrible errors
- Reinhardt & Rogoff see here for more
- Other famous Excel disasters
- You can run macros, but then it gets awkward anyway

Reproducible work: why

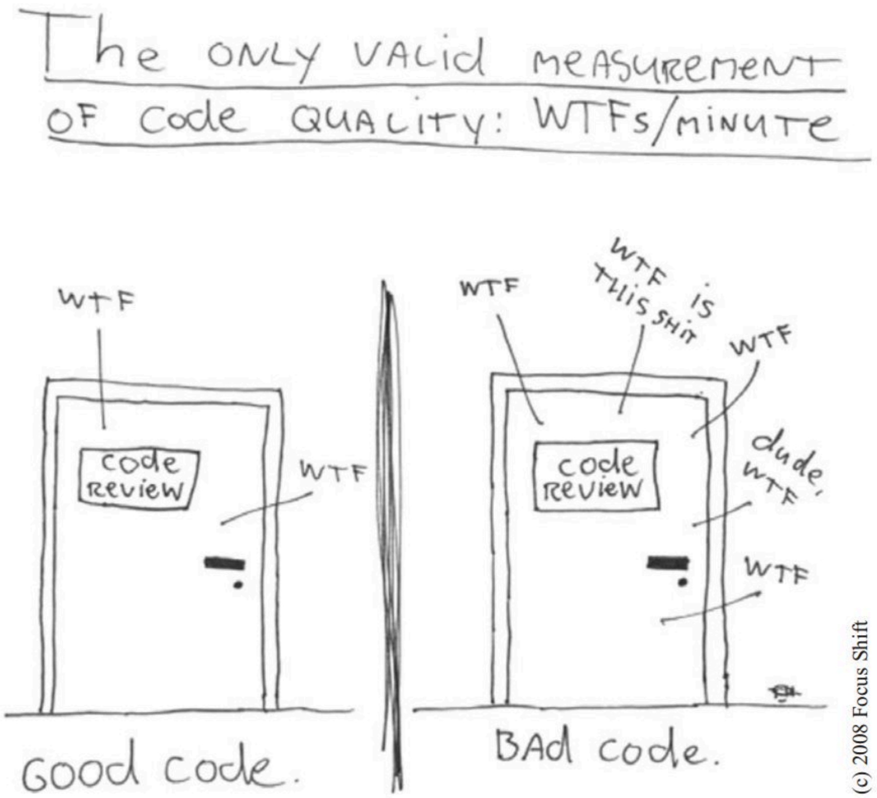
> *Whenever you do serious analysis, consider this*

- in any analysis, *hundreds* of `operations` on your data
- each can be a `mistake`, can be forgotten, or done twice
- 1 month later: **`will you remember`**?

- **NO**
- it’s hard **for you**. Imagine for **others**.

Reproducible work: why

> Take care of your future self: do `reproducible` work



Reproduced with the kind permission of Thom Holwerda.
http://www.osnews.com/story/19266/WTFs_m

Reproducible work: how

> *You need to keep track of what you do.*

- always use a **script-based** language
- experiment interactively, try and re-try…
- …but then lock what you did in a *script*
- **comment** extensively

Install time!

Install checklist

- R (https://pbil.univ-lyon1.fr/CRAN/)
- Rstudio https://posit.co/downloads/)
- Package `tidyverse`:
    - open Rstudio
    - `install.packages(“tidyverse”)`
- git (https://git-scm.com/downloads)
- `gitlab` UGA (https://gricad-gitlab.univ-grenoble-alpes.fr/)

Setting up the course

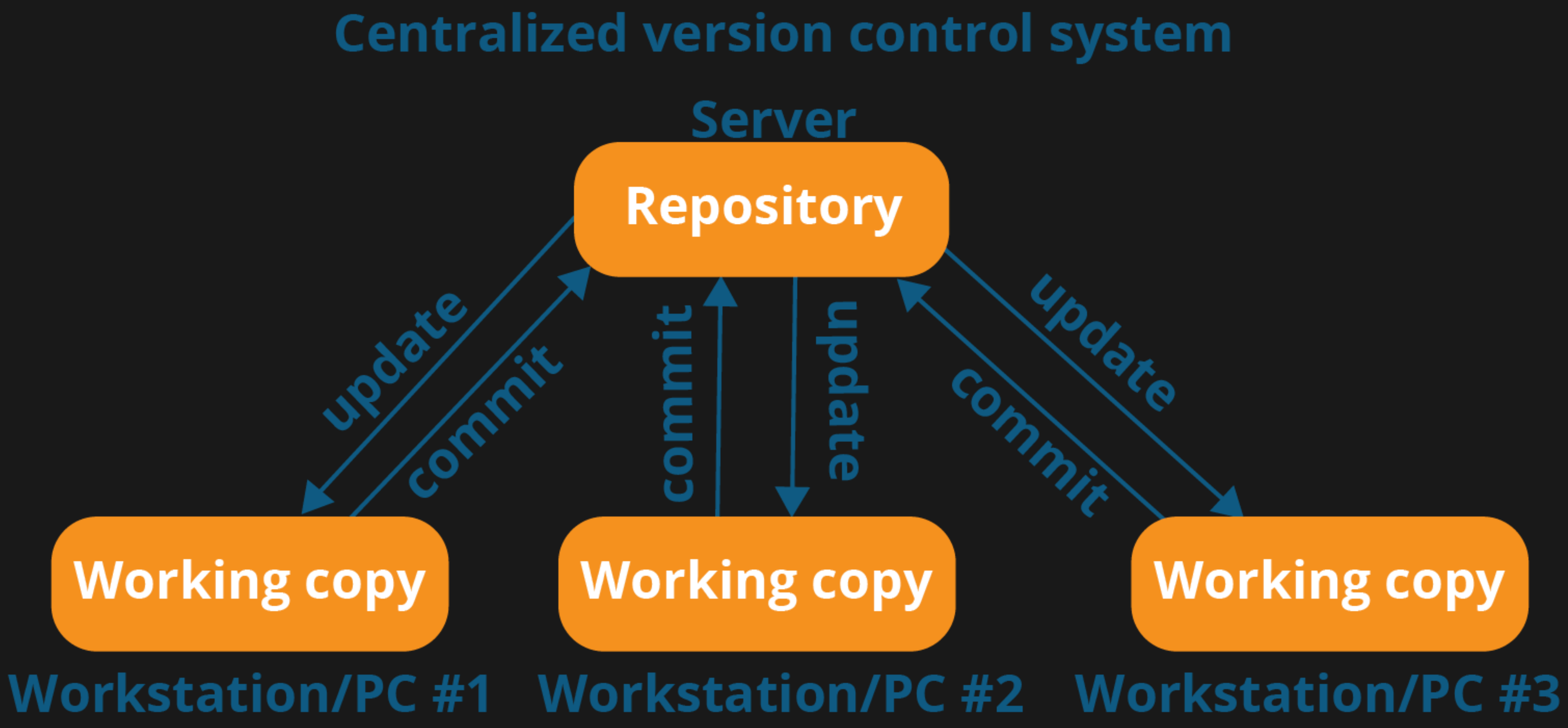
- I will invite you to the `gitlab`: **accept**
- Open Rstudio
- File >> New Project >> Version Control >> Git
- Type in the URL: https://gricad-gitlab.univ-grenoble-alpes.fr/crosettp/intro_to_r_2024_25.git
- Select a directory where you want to store your project
- `All set`!

git & gitlab

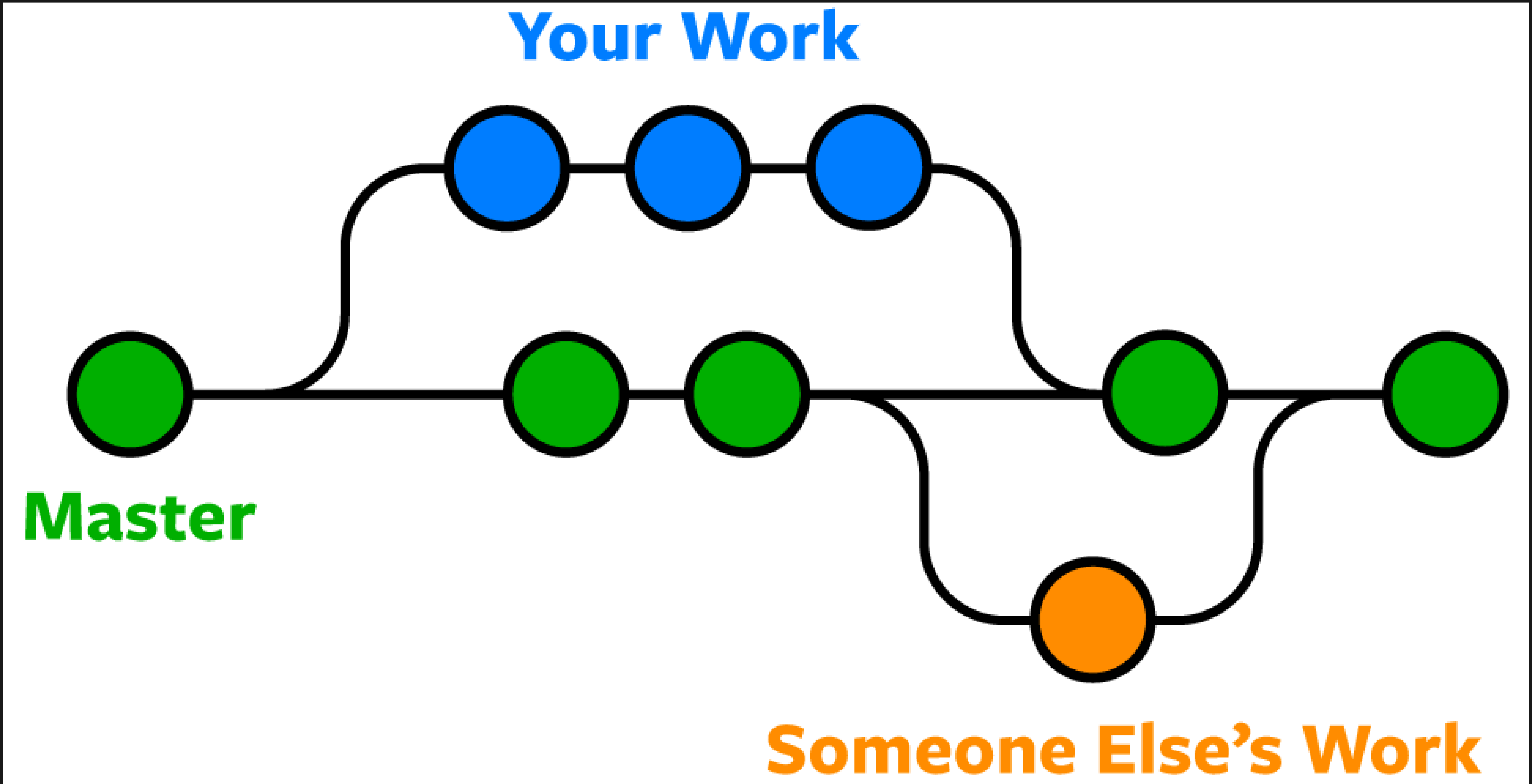
Reproducible work: online repository

- like word **comments** but for coders
- like **dropbox** but for coders
- keeping **track** of all code changes
- allowing **cooperation**
- forcing **clarity**

What is git /1


Centralized version control system
Server
Repository
update
commit
commit
update
update
commit
Working copy
Working copy
Working copy
Workstation/PC #1
Workstation/PC #2
Workstation/PC #3

What is git /2



Git basics

1. `pull` the remote repo to your local machine.
2. work locally, make changes…
3. see? git pane updates
4. check changes: diff
5. happy with the changes? `commit` them (using a nice message)
6. done a bunch of changes? `push` them to the remote repo
7. repeat

Gitlab exercise

- Navigate to your student folder.
- You’ll see a `deleteme.txt` file
- Open it
- Follow instructions

RStudio

RStudio `tour`

- panes
- resources
- console
- history
- plots
- packages
- help
- editor

Rstudio exercise

1. investigate what does the function `summary` do (`?`)
2. find package info for package `babynames`
3. not found? install `babynames` (`install.packages`)
4. try to `summary babynames`
5. import the `babynames` dataset to your environment
6. explore the objects in your environment
7. `View()` filter, arrange, reorder the dataset visually

R basics

First, let’s start a `notes` file

- in the Files pane
  - navigate to `your` **student folder**
  - Click on New Blank File >> R script
  - name it “`notes_L1.R`”; click to open it
- it now opens in the Editor pane
- in the first line, type “`library(tidyverse)`” & save.
- `commit`, `pull`, `push`

> I also start a `notes_L1.R` file in `/Paolos_Notes`

Some R basic commands

- concatenate to create vectors: `c()`
- assign: `<-`
- turn vectors into matrices: `rbind(), cbind()`
- matrix extraction: `[ ]`
- data frames (tibbles): `mpg`
- variable extraction: `$`
- variable types and how to change them: `is.*`; `as.*`
- create strings: `paste`

Exercises

> **1.** *Create a vector of 4 numbers, call it “vec1”. Then another, call it “vec2”.*
>
> *Create a column matrix of these two.*

> **2.** *assign the data frame* `mpg` *to an object “**df**”*
>
> *extract the columns* `hwy` *and* `cty`*, turn them into a character*
>
> *create a single text variable of city and highway consumption*

Quarto

Markdown

> a simple markup language to generate simple text file with minimal syntax

- renders to `HTML` by default
- (but also to `PDF` or to different types of slides)
- (and to **word**, and to **powerpoint**)
- **minimal** syntax
- cheat sheet (Rstudio -> Help -> Cheatsheets -> R Markdown reference guide)

Exercise

> Create your first (and last!) Markdown file

- in the Files pane, navigate to **your** folder
- New Blank File >> Text file
- name it `my_first.md`
- open it in editor; write stuff
- try rendering in different format
- `commit`, `pull`, `push`

Rmarkdown

> allows to insert R code into markdown documents

- code *blends* into text files
- code is **executed** each time the document is compiled
- all dependencies installed by default on Rstudio

Quarto

- A complete modern publishing tools in R
- Tutorial: File -> New File -> Quarto document…
- Embed R code
- Export to a variety of formats

Exercise

> Create your first (of many!) Quarto file

- in the Files pane, navigate to your folder
- New Blank File >> Quarto doc…
- name it `my_first_quarto.qmd`
- open it in editor
- we’ll play around together
- try rendering in different format
- `commit`, `pull`, `push`

Exercise

1. Open `my_first_quarto.qmd`
2. add a code chunk: `R` compute the square root of a number
3. `Render`
4. `commit`, `pull`, `push`

Quarto in this course

- These slides made using `quarto`
- Your final exam statistical essay will use `quarto`
- you can easily deploy `.qmd` to the web

Online resources

Cheat sheets

> you might want to have a list of functions at hand for your exercises

- Base R cheat sheet: here
- Rstudio cheat sheet: here
- data manipulation cheat sheet: here
- ggplot2 cheat sheet: here
- Eurostat cheat sheet: here

Books and other online resources

- Tidyverse website: package documentation and examples
- R for Data Science: book covers all of the course and more
- Happy git with R: for a `git` and `Rstudio` guide

Getting info

- **inline help**
  - `?command`: returns command documentation
  - `??topic`: returns search results for the topic
- **vignettes**
  - `HTML` tutorials, with code and results
- **StackExchange** here
  - code Q&A website. Not `R`-only.
  - use sensible keywords
  - learn how to ask a question