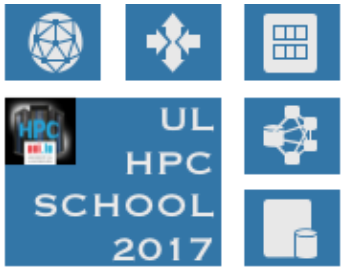
PS12: introduction to R

HPC School 2018

Aurélien Ginolhac

2018-06-13

What is R?



is shorthand for "GNU R":

- An **interactive** programming language derived from **S** (**J. Chambers**, Bell Lab, 1976)
- Appeared in 1993, created by **R. Ihaka** and **R. Gentleman**, University of Auckland, NZ
- Focus on data analysis and plotting
- is also shorthand for the ecosystem around this language
  - Book authors
  - Package developers
  - Ordinary useRs

Learning to use **R** will make you **more efficient** and **facilitate the use** of advanced data analysis tools

R practical session

Why use R?

- It's *free!* and **open-source**
- easy to install / maintain
- multi-platform (Windows, macOS, GNU/Linux)
- can process big files and analyse huge amounts of data (db tools)
- integrated data visualization tools, *even dynamic*
- fast, and even faster with *C++* integration via Rcpp.
- easy to get help
  - huge R community in the web
  - stackoverflow with a lot of tags like **r**, **ggplot2** etc.
  - rbloggers

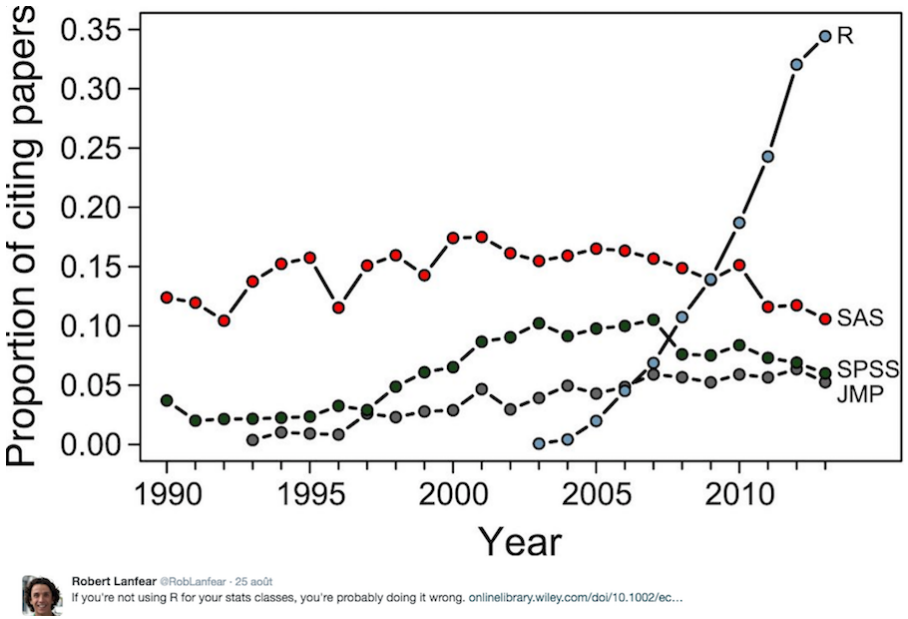
R practical session

Twitter R community

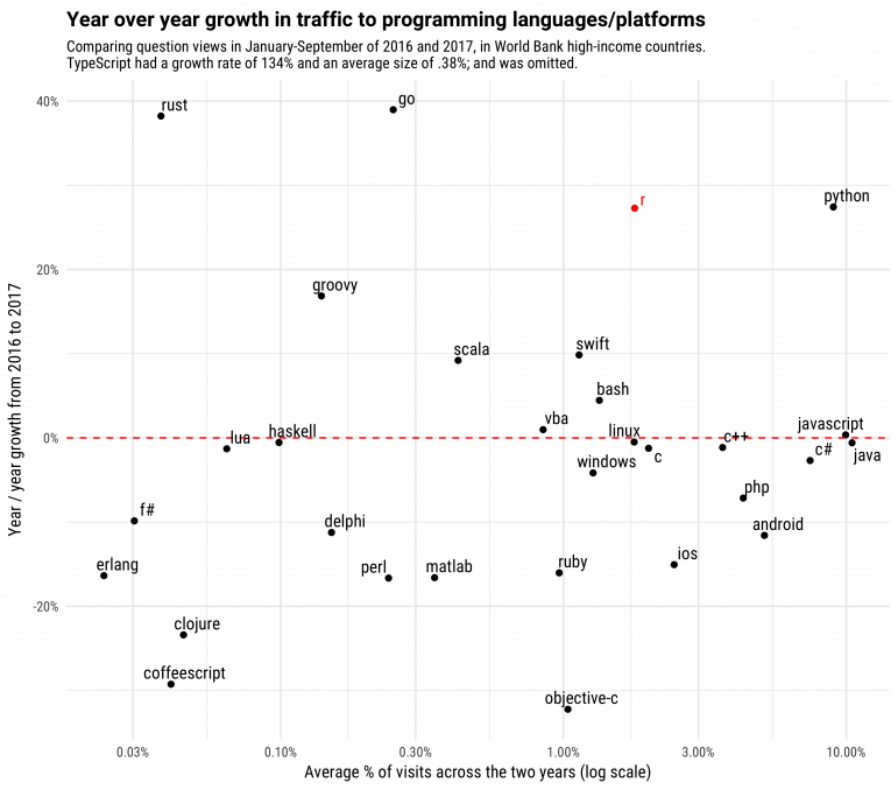
#rstats on twitter

R practical session

Constant trend



Source: Touchon & McCoy. *Ecosphere*. 2016



Source: D. Robinson, StackOverflow blog

R practical session

Packages

+12,000 in Feb 2018

CRAN

reliable: package is checked during submission process

MRAN for Windows users

bioconductor

dedicated to biology. status

typical install:

```
source("https://bioconductor.org/biocLite.R")
biocLite("limma")
```

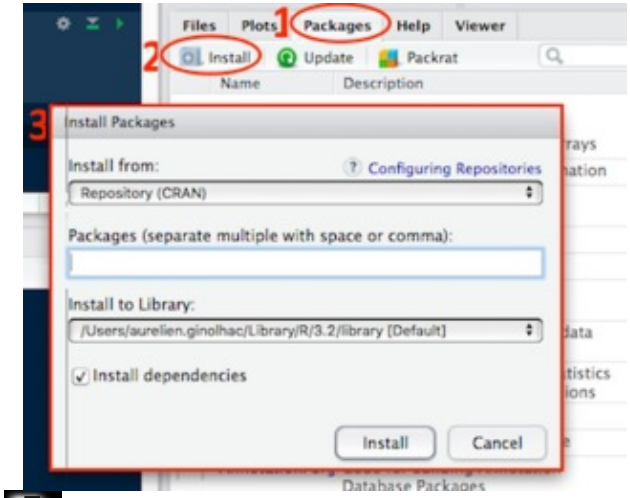
GitHub

easy install thanks to devtools. status

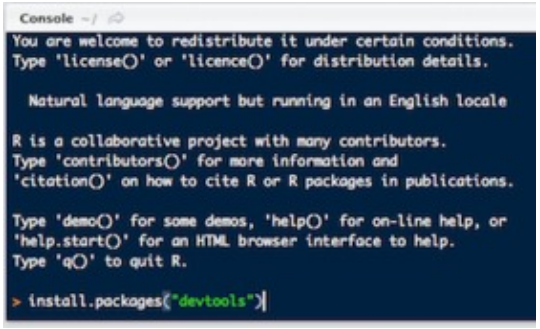
```
# install.packages("devtools")
devtools::install_github("tidyverse/readr")
```

could be a security issue

CRAN install from Rstudio



github install from Rstudio' console



more in the article from David Smith

R practical session

R is hard to learn

R base is complex, has a long history and many contributors

source: Robert A. Muenchen' blog

R practical session

Tidyverse

creator

We think the **tidyverse** is better, especially for beginners. It is

- recent (both an issue and an advantage)
- allows doing powerful things quickly
- unified
- consistent, one way to do things
- give strength to learn base R
- criticisms will come later (yes, many)

Hadley Wickham

Hadley, Chief Scientist at **Rstudio**

- coined the *tidyverse* at userR meeting in 2016
- developed and maintains most of the core *tidyverse* packages

R practical session

RStudio

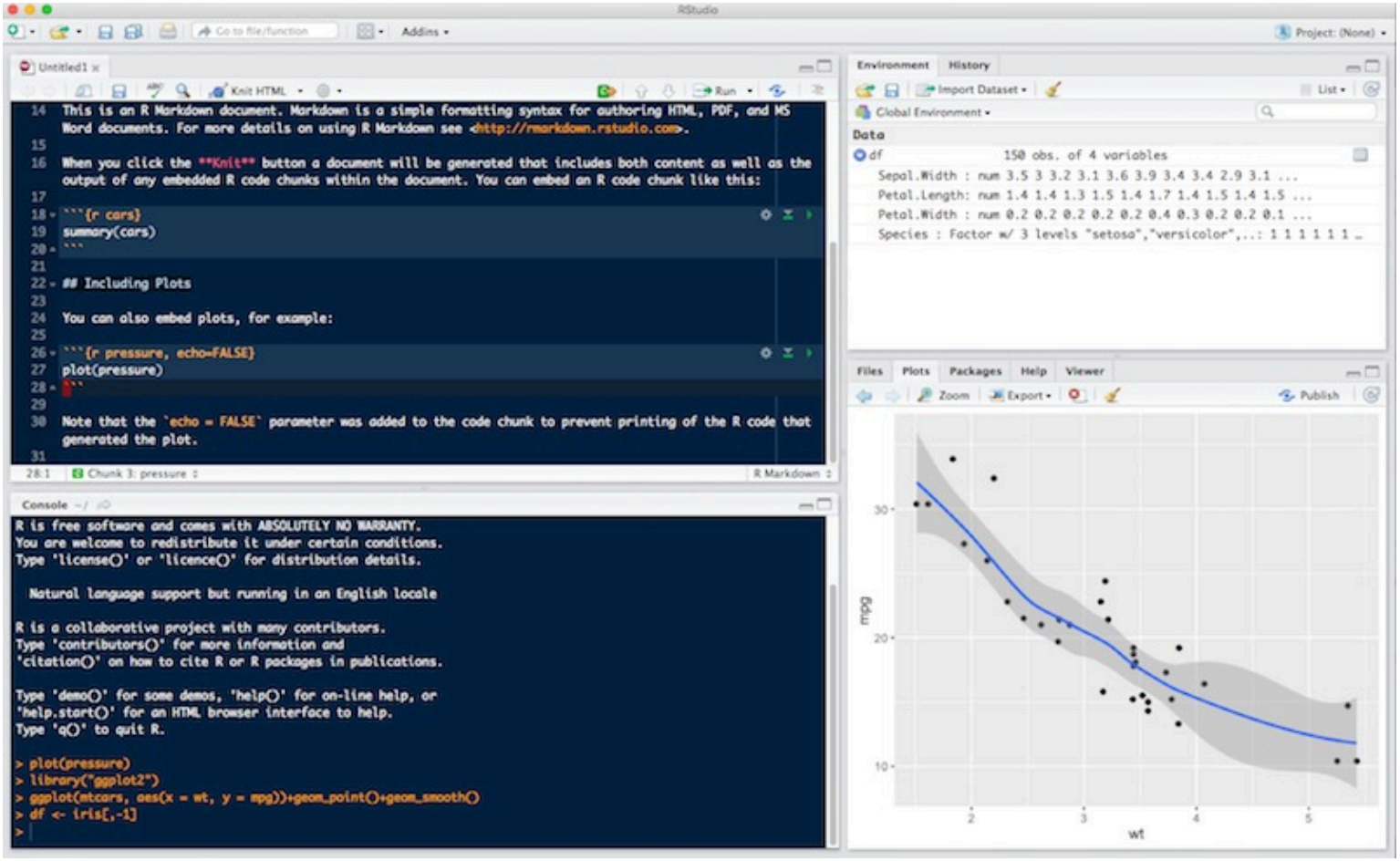
Rstudio

makes working with R easier

R practical session

Rstudio

The 4 panels layout



R practical session

Four panels

scripting

- could be your main window
- should be a **Rmarkdown** doc
- tabs are great

Environment

- Environment, display loaded objects and their `str()`
- History is useless IMO
- nice `git` integration
- database **connections** interface

Console

- could be hidden with **inline** outputs
- embed a nice `terminal` tab
- `Rmarkdown` logs

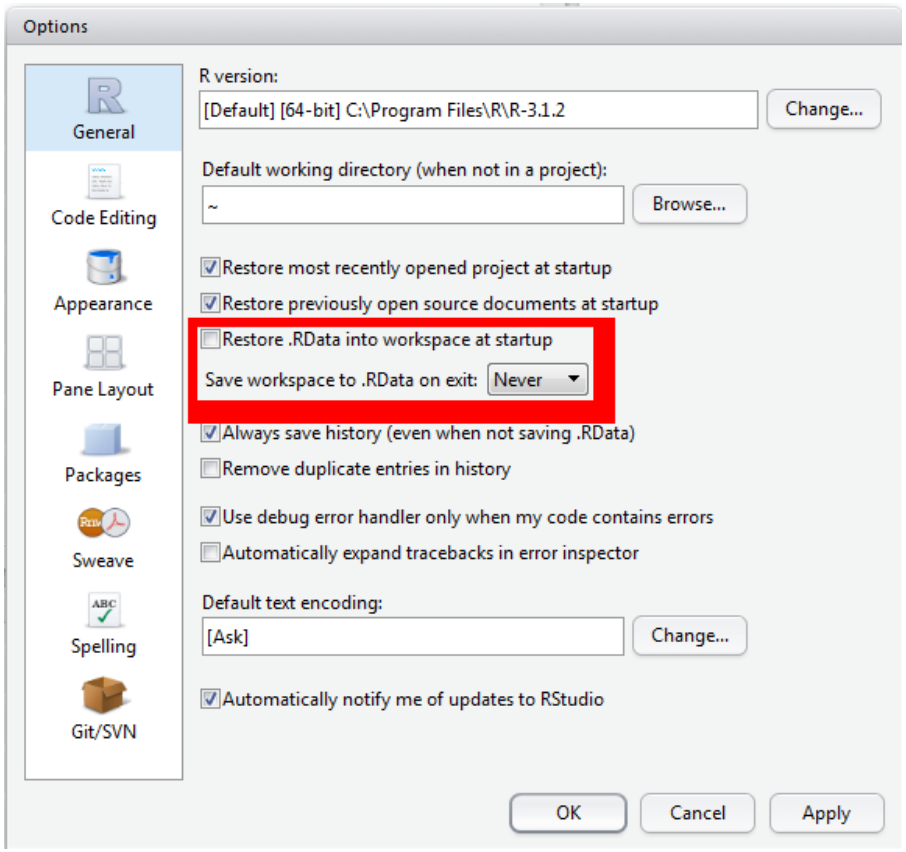
Files / Plots / Help

- necessary package management tab
- unnecessary plots tabs with **inline** outputs
- help tab

R practical session

For reproducibility

options to activate / deactivate



`rm(list = ls())` is not recommended

- does **not** make a fresh R session
  - `library()` calls remain
  - working directory not set!
  - modified functions, evil `==` <- `!=`
- knitting `Rmarkdown` files solves it

source: Jenny Bryan article

R practical session

Code diagnostics

highly recommended

using `Global Options -> Code -> Diagnostics` editing pane:

source: Kevin Ushey' article

- check argument calls

```
1  add_numbers <- function(x, y) {
2    x + y
3  }
4
5  add_numbers(1)
6
```

argument 'y' is missing, with no default

- missing arguments

```
1  list(
2    first = 1
3    second = 2
```

expected ',' after expression

- variable definitions

```
1  hw <- HoltWinters(ldeaths)
2  p <- predict(HW, n.ahead = 36, level = 0.95)
3
4
```

no symbol named 'HW' in scope; did you mean 'hw'?

- unused variables & style recommendations

R practical session

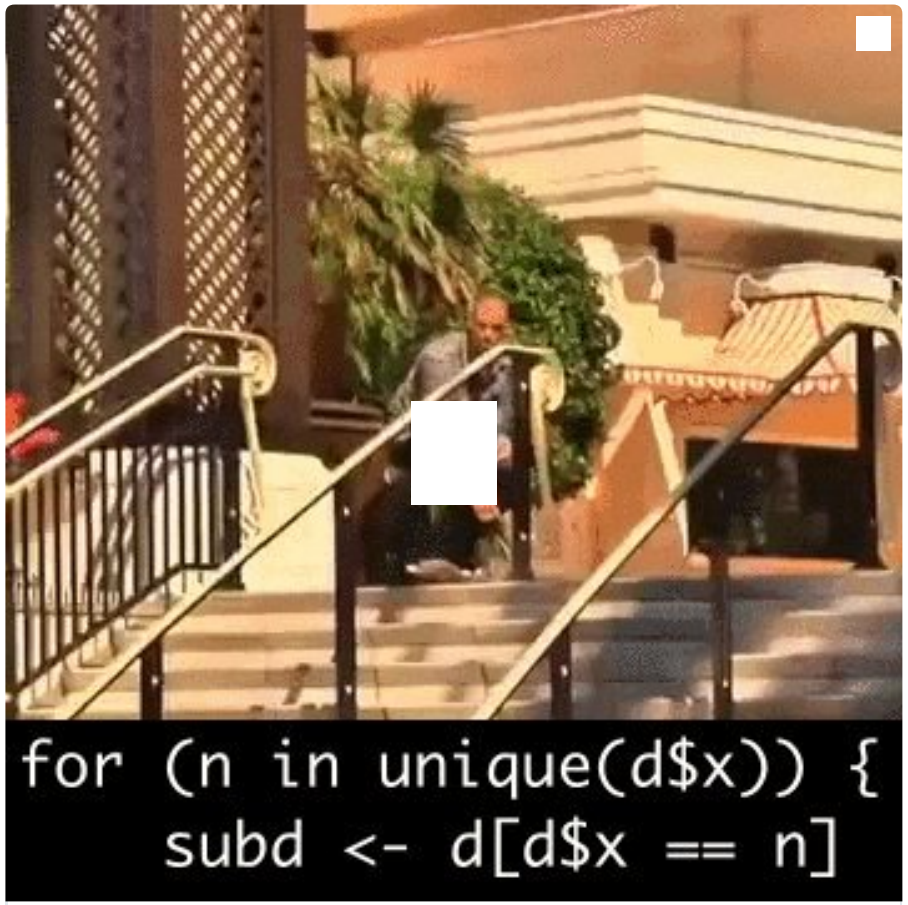
Data types and structures

R base

Necessary R base

We could let **base** down, but the **tidyverse** is wrapping around it

Some functions need to be known. And in R, everything is a function.



David Robinson
@drob


R practical session

Getting started

Let's get ready to use **R** and **RStudio**

Do the following

- Open up RStudio
- Maximize the RStudio window
- Click the Console pane, at the prompt (>) type in `3 + 2` and hit enter

```
> 3 + 2
```

R practical session

4 main types

mode( )

Type	Example
numeric	integer (2), double (2.34)
character (strings)	“tidyverse!”
boolean	TRUE / FALSE
complex	2+0i

R practical session

Structures

Vectors

`c()` is the function for **concatenate**

R practical session

Data frames are special lists

`data.frame`

same as list **but** where all objects *must* have the **same** length

Example, 3 elements of same size

```
data.frame(
  f = factor(c("AA", "AA", "BB")),
  v = c(43, 5.6, 2.90),
  s = rep(4, 3))
```

```
  f    v s
1 AA 43.0 4
2 AA  5.6 4
3 BB  2.9 4
```

Example, missing one element in `v`

```
data.frame(
  f = factor(c("AA", "AA", "BB")),
  v = c(43, 5.6),
  s = rep(4, 3))
Error in data.frame(f = factor(c("AA", "AA", "BB")),
v = c(43, 5.6), s = rep(4, : arguments imply
differing number of rows: 3, 2
```

R practical session

Concatenate atomic elements

i.e build a vector

collection of simple *things*

- *things* are the smallest elements: **atomic**
- **must** be of same **mode**: automatic coercion
- indexed, from **1** to `length(vector)`
- created with the `c()` function

```
c(2, TRUE, "a string")
[1] "2"        "TRUE"     "a string"
```

assignment operator, create object

operator is `<-`, associate a *name* to an object

```
my_vec <- c(3, 4, 1:3)
my_vec
[1] 3 4 1 2 3
```

Tip

Rstudio has the built-in shortcut Alt + - for `<-`

R practical session

hierarchy

source: **H. Wickham** - *R* for data science, licence CC

in console

```
is.vector(c("a", "c"))
[1] TRUE
mode(c("a", "c"))
[1] "character"
is.vector(list(a = 1))
[1] TRUE
is.atomic(list(a = 1))
[1] FALSE
is.data.frame(list(a = 1))
[1] FALSE
```

R practical session

Vectors

subsetting

important

Unlike **python** or **Perl**, vectors use **1-based** index!!

R practical session

Vectorized operation

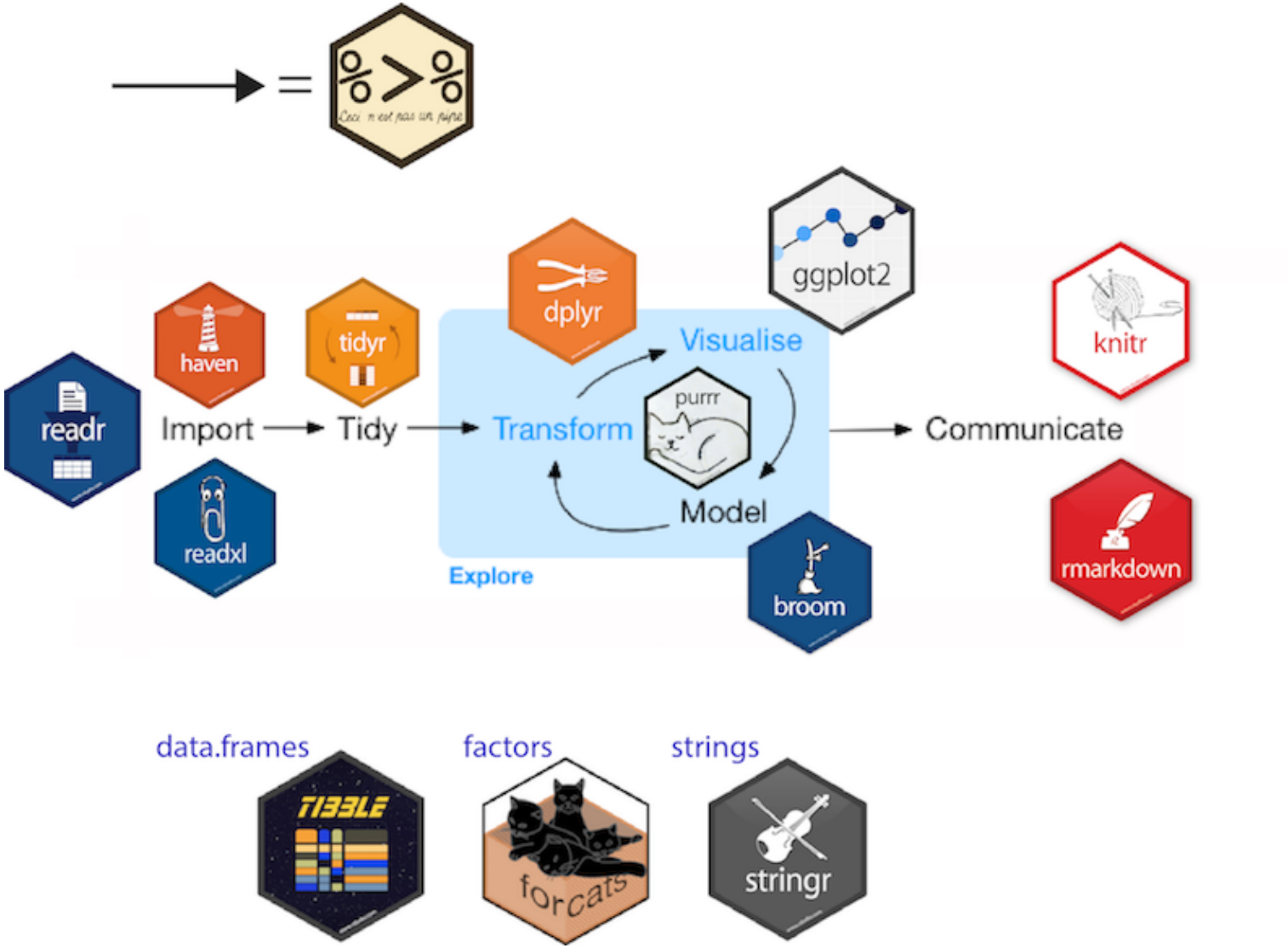
one of the best R feature

```
my_vec <- 10:18
my_vec
[1] 10 11 12 13 14 15 16 17 18
my_vec + 2
[1] 12 13 14 15 16 17 18 19 20
```

R practical session

Tidyverse

packages in processes



R practical session

Tidyverse criticism

jobs

Yeedle
@yeedle

Realized today: #tidyverse R and base #rstats have little in common. Beware when looking for job which requires knowledge of R.

01:42 - 3 mars 2017

3 Voir les autres Tweets de Yeedle

Personal complains

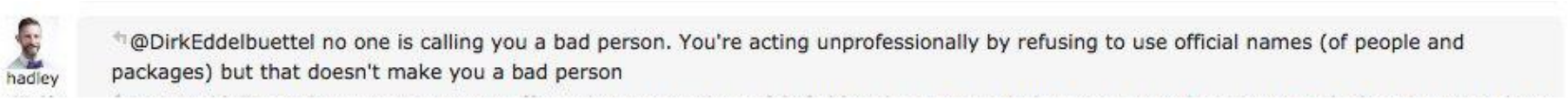
- still young so change quickly and drastically
- Backward compatibility is not always maintained.
- `tibbles` are nice but a lot of non-tidyverse packages require `matrices`. `rownames` still an issue.

No need for opposition base / tidyverse

Learning the *tidyverse* does not prevent to learn *R base*, it helps to get things done early in the process

R practical session

Community complains



source: SO, R chat room, 29 Nov 2017

4 days workshop at the doctoral school@Uni last Feb 2018, probably again March 2019

R practical session

Practical Session

Objectives

You will learn to:

- install and run R and Rstudio on your machine
- use R on the clusters
- download a file and process it
- create a simple *ggplot* remotely
- summarise a dataset using different packages and benchmark them
- demonstrate why packages are so much better than R base
- perform single machine parallelisation on `gaia`
- perform cluster parallelisation on `gaia`

R practical session

Acknowledgements

R practical session

Wrap up

You learned to:

- Introduction
  - R
  - Rstudio
  - tidyverse rationale
- data types
  - main categories
  - coerce
- data structures
  - main categories
  - sub-setting
  - vectorization

- Jospeh Emeras who wrote earlier version of this session
- Eric Koncina, slides prepared with his iosp R package
- Eric Koncina & Roland Krause for their content in the R workshop
- *HPC team*

R practical session