R for Hydrology

Capabilities and recent developments

Cas Neyens
27 February 2018

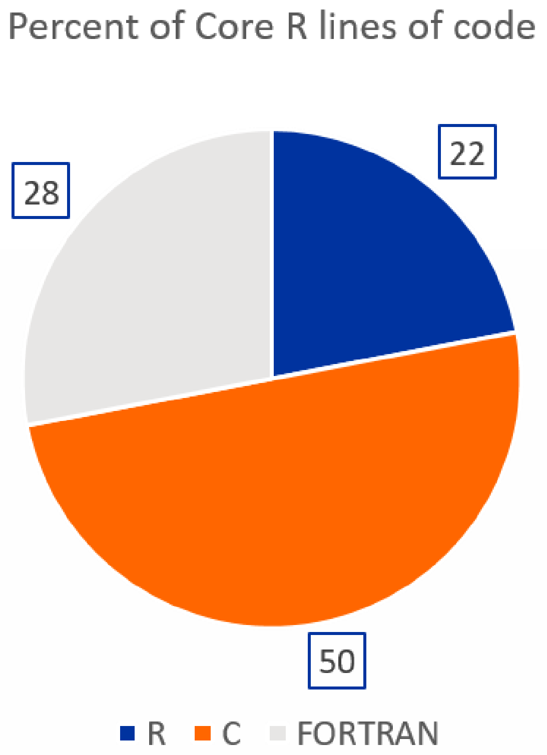
IUPWARE alumni event 2018
Cuenca, Ecuador

About R

- High-level, interpreted programming language
- Scripts
- Community based (i.e. packages, see CRAN)
- Free
- Great IDE: RStudio
- Source code: C, Fortran & R

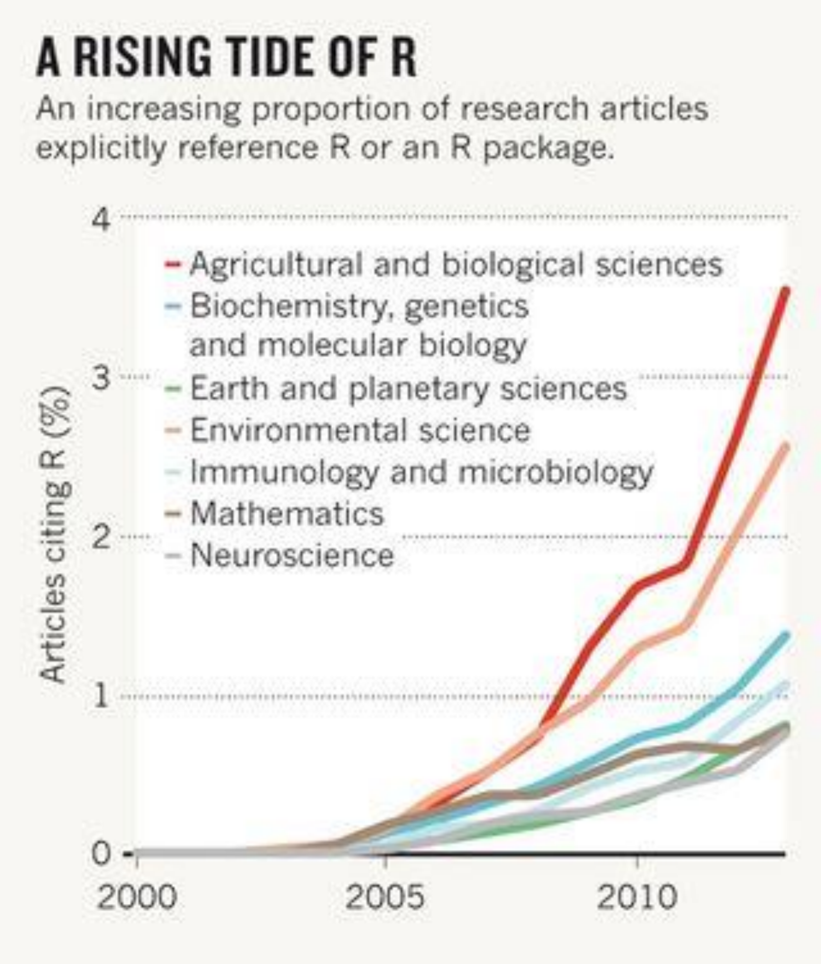
About R

- High-level, interpreted programming language
- Scripts
- Community based (i.e. packages, see CRAN)
- Free
- Great IDE: RStudio
- Source code: C, Fortran & R



Why R?

- Focus on data analysis
- All data in a single environment
- Growing community
  - *Also in hydrology*
- “Should I use Python or R ?”
  - *Both are great*
- Reproducible research !!



Sylvia Tippmann/Source: Elsevier Scopus database

Reproducible research

"Most computational hydrology is not reproducible, so is it really science?"

Hutton et al., 2016

- Hydrological models rarely reproducible
- Can be improved by scripted analysis & input generation

Demo overview

1. Quick introduction to data types
2. Data: importing, cleaning and transforming
3. Time-series analysis
4. Using R as a GIS
5. Pre- and postprocessors of models
6. A little bit about GitHub

Data types

Data structures
Scalar
Vector
Arrays & matrices
List
Data frame (≈ tibble)

Data classes
Numeric
Character
Logical
Factor
User created

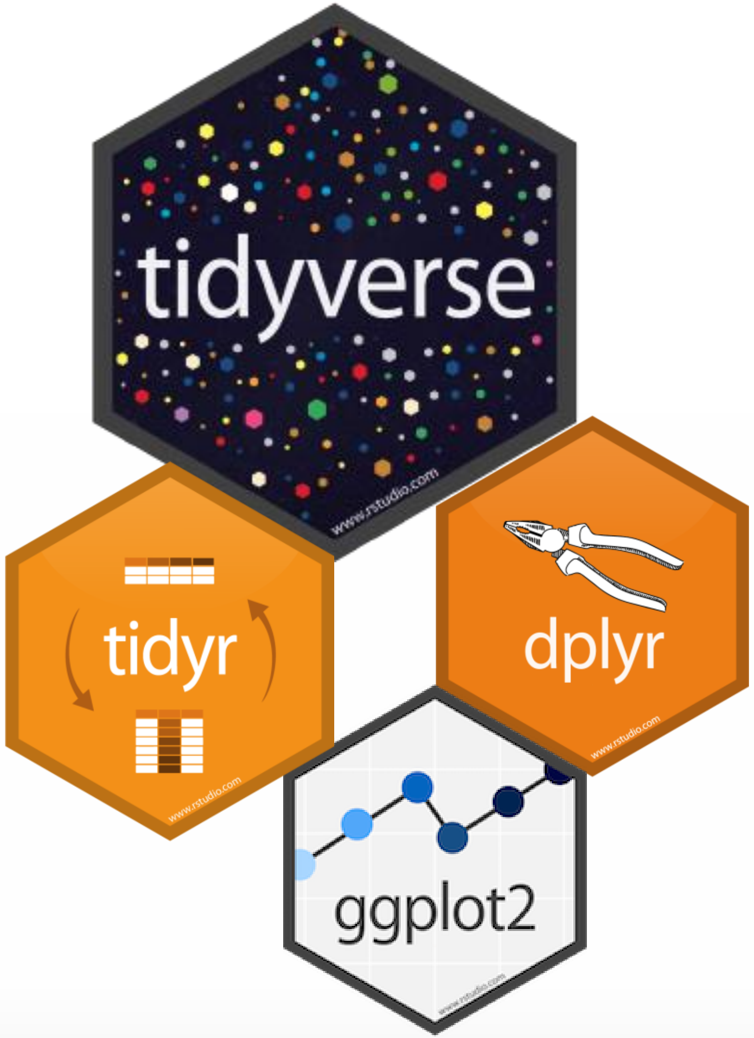
Data handling

- “Clean data”:
  1. Each variable is in a column
  2. Each observation is a row
  3. Each value is a cell

- Tidyverse

- → data in long format

- Data has an ‘id’



Data handling

Precipitation (mm)

Date	Dessel	Vorselaar	Overpelt	Tessenderlo
26/04/2011	0.00	0.00	0.00	0.00
27/04/2011	0.25	0.39	2.90	0.75
28/04/2011	1.52	1.43	0.99	1.30
29/04/2011	2.03	3.85	0.30	1.88

Wide format

☹

Data handling

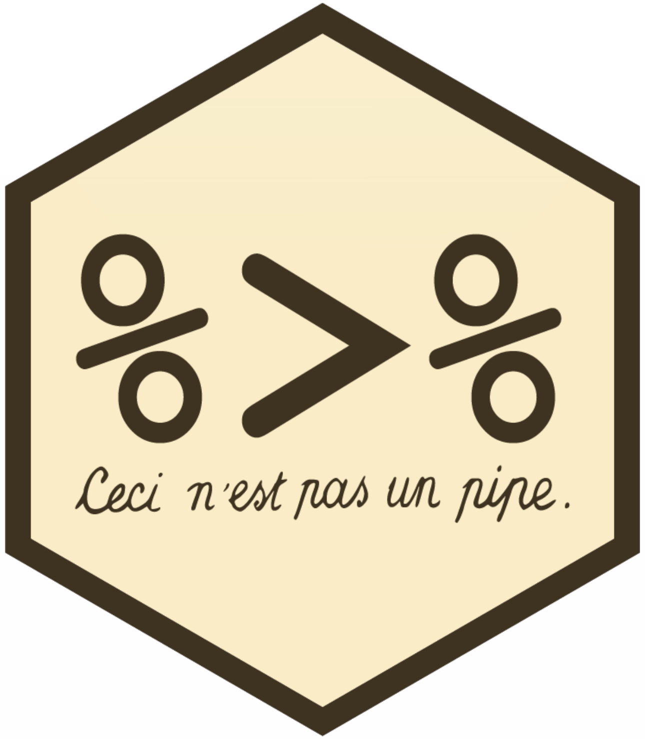
Date	Location	Precipitation (mm)
26/04/2011	Dessel	0.00
27/04/2011	Dessel	0.25
28/04/2011	Dessel	1.52
29/04/2011	Dessel	2.03
26/04/2011	Vorselaar	0.00
27/04/2011	Vorselaar	0.39
28/04/2011	Vorselaar	1.43
29/04/2011	Vorselaar	3.85
26/04/2011	Overpelt	0.00
27/04/2011	Overpelt	2.90
28/04/2011	Overpelt	0.99
29/04/2011	Overpelt	0.30
26/04/2011	Tessenderlo	0.00
27/04/2011	Tessenderlo	0.75
28/04/2011	Tessenderlo	1.30
29/04/2011	Tessenderlo	1.88

Long fomat

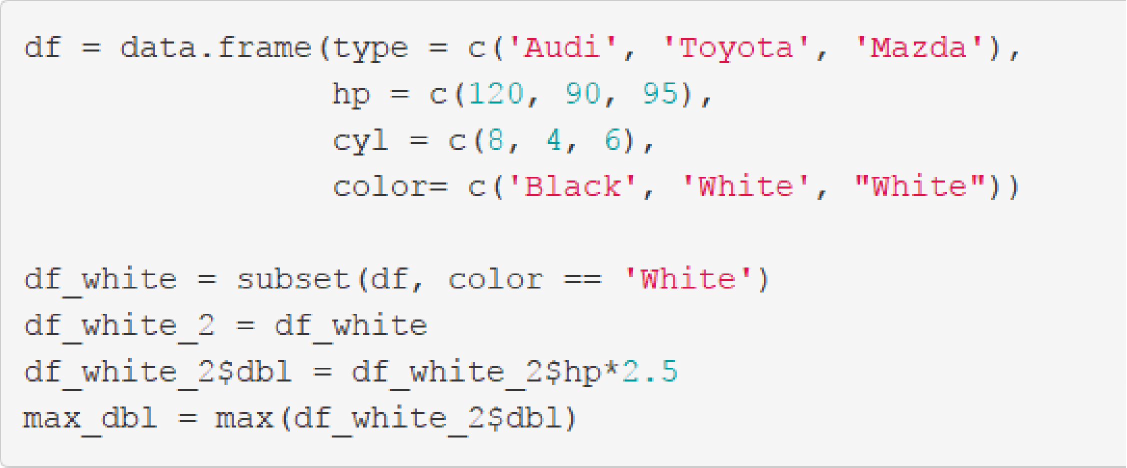
☺

Data handling

`magrittr` package

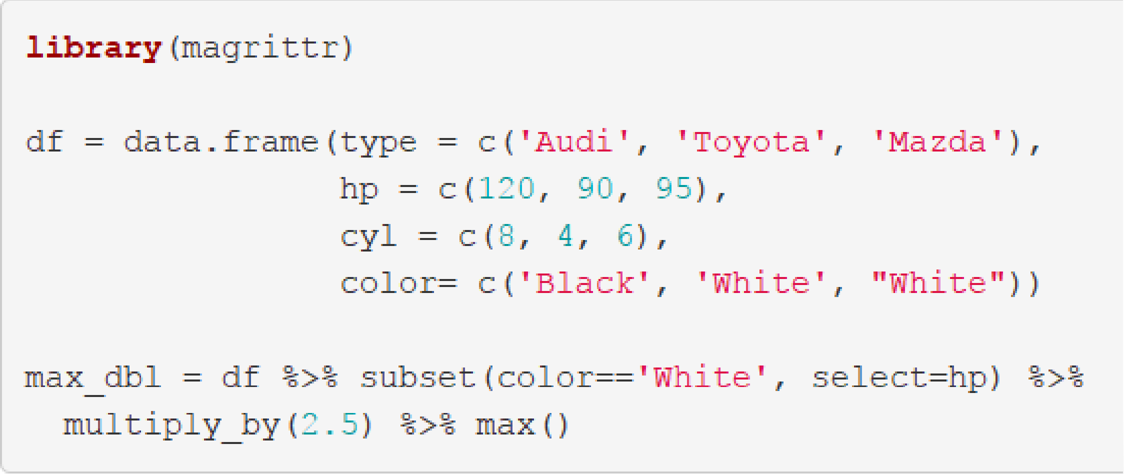


Previously



```
df = data.frame(type = c('Audi', 'Toyota', 'Mazda'),
                hp = c(120, 90, 95),
                cyl = c(8, 4, 6),
                color= c('Black', 'White', "White"))

df_white = subset(df, color == 'White')
df_white_2 = df_white
df_white_2$dbl = df_white_2$hp*2.5
max_dbl = max(df_white_2$dbl)
```

Now

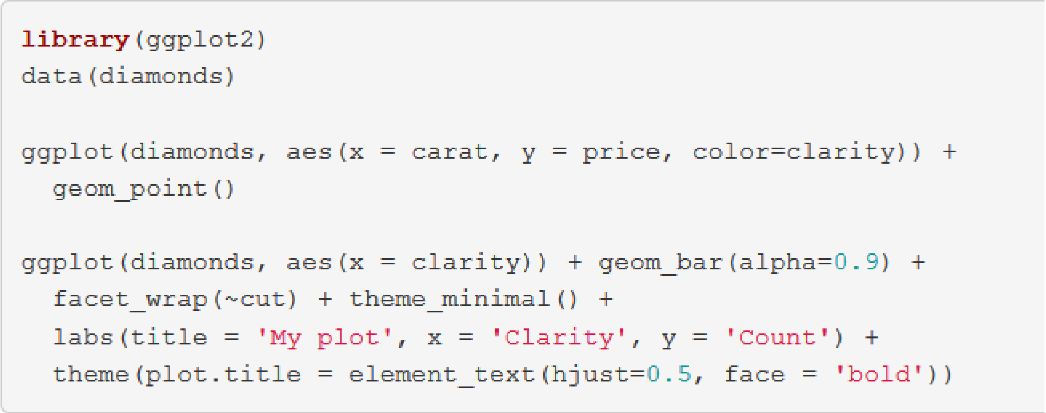


```
library(magrittr)

df = data.frame(type = c('Audi', 'Toyota', 'Mazda'),
                hp = c(120, 90, 95),
                cyl = c(8, 4, 6),
                color= c('Black', 'White', "White"))

max_dbl = df %>% subset(color=='White', select=hp) %>%
  multiply_by(2.5) %>% max()
```

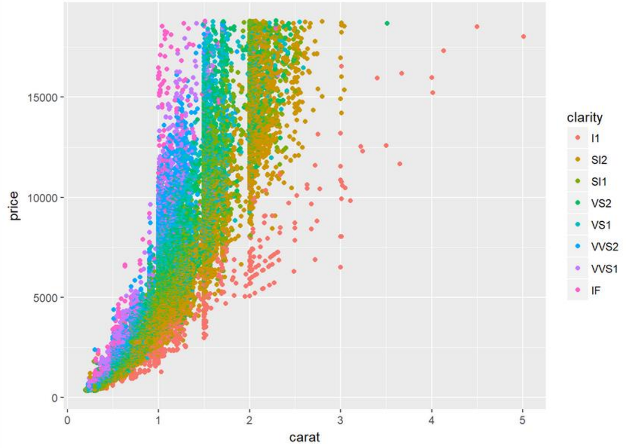
Data plotting

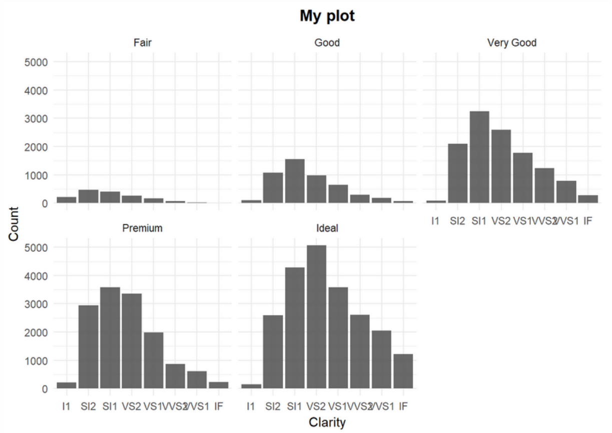
`ggplot2` package

- Versatile
- Easy to plot complex data



```r
library(ggplot2)
data(diamonds)

ggplot(diamonds, aes(x = carat, y = price, color=clarity)) +
  geom_point()

ggplot(diamonds, aes(x = clarity)) + geom_bar(alpha=0.9) +
  facet_wrap(~cut) + theme_minimal() +
  labs(title = 'My plot', x = 'Clarity', y = 'Count') +
  theme(plot.title = element_text(hjust=0.5, face = 'bold'))
```





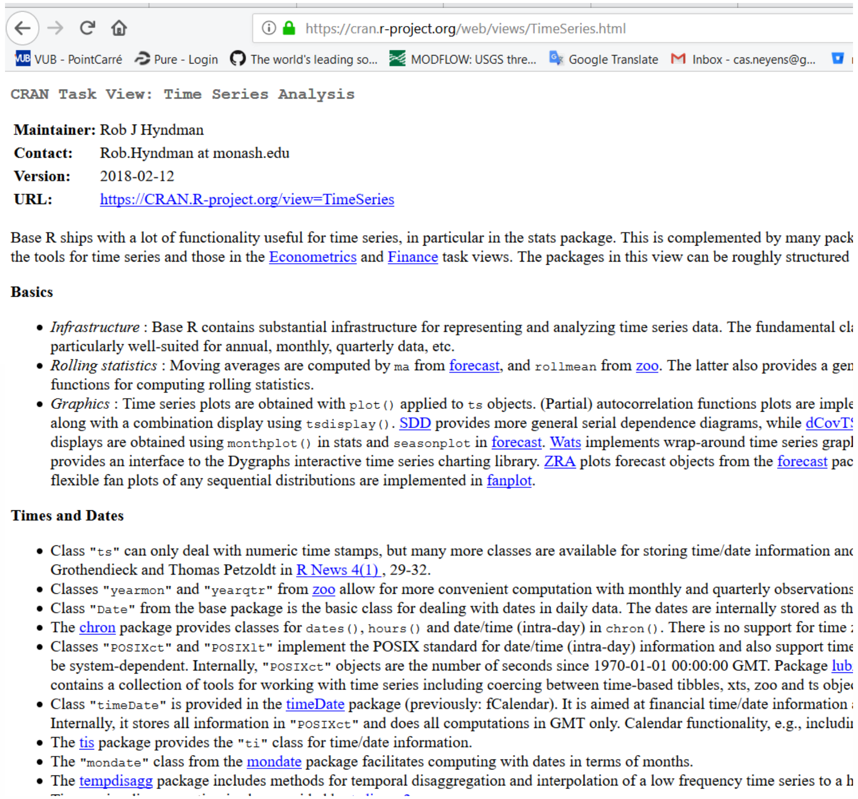
Time series analysis

- R is very adapt at handling time series

- Various packages and formats available
  - `xts` & `ts` traditionally very popular

- Non-uniformity in data classes & structures

- Mostly designed for finance/economics forecasting



CRAN Task View: Time Series Analysis

Maintainer: Rob J Hyndman
Contact: Rob.Hyndman at monash.edu
Version: 2018-02-12
URL: https://CRAN.R-project.org/view=TimeSeries

Base R ships with a lot of functionality useful for time series, in particular in the stats package. This is complemented by many pack
the tools for time series and those in the Econometrics and Finance task views. The packages in this view can be roughly structured

Basics

- *Infrastructure* : Base R contains substantial infrastructure for representing and analyzing time series data. The fundamental cla
particularly well-suited for annual, monthly, quarterly data, etc.
- *Rolling statistics* : Moving averages are computed by `ma` from forecast, and `rollmean` from zoo. The latter also provides a gen
functions for computing rolling statistics.
- *Graphics* : Time series plots are obtained with `plot()` applied to `ts` objects. (Partial) autocorrelation functions plots are imple
along with a combination display using `tsdisplay()`. SDD provides more general serial dependence diagrams, while dCovTS
displays are obtained using `monthplot()` in stats and `seasonplot` in forecast. Wats implements wrap-around time series grapl
provides an interface to the Dygraphs interactive time series charting library. ZRA plots forecast objects from the forecast pac
flexible fan plots of any sequential distributions are implemented in fanplot.

Times and Dates

- Class `"ts"` can only deal with numeric time stamps, but many more classes are available for storing time/date information anc
Grothendieck and Thomas Petzoldt in R News 4(1) , 29-32.
- Classes `"yearmon"` and `"yearqtr"` from zoo allow for more convenient computation with monthly and quarterly observations
- Class `"Date"` from the base package is the basic class for dealing with dates in daily data. The dates are internally stored as th
- The chron package provides classes for `dates()`, `hours()` and date/time (intra-day) in `chron()`. There is no support for time
- Classes `"POSIXct"` and `"POSIXlt"` implement the POSIX standard for date/time (intra-day) information and also support time
be system-dependent. Internally, `"POSIXct"` objects are the number of seconds since 1970-01-01 00:00:00 GMT. Package lub
contains a collection of tools for working with time series including coercing between time-based tibbles, xts, zoo and ts objec
- Class `"timeDate"` is provided in the timeDate package (previously: fCalendar). It is aimed at financial time/date information
Internally, it stores all information in `"POSIXct"` and does all computations in GMT only. Calendar functionality, e.g., includi
- The tis package provides the `"ti"` class for time/date information.
- The `"mondate"` class from the mondate package facilitates computing with dates in terms of months.
- The tempdisagg package includes methods for temporal disaggregation and interpolation of a low frequency time series to a h

Dealing with dates & times

`lubridate` package

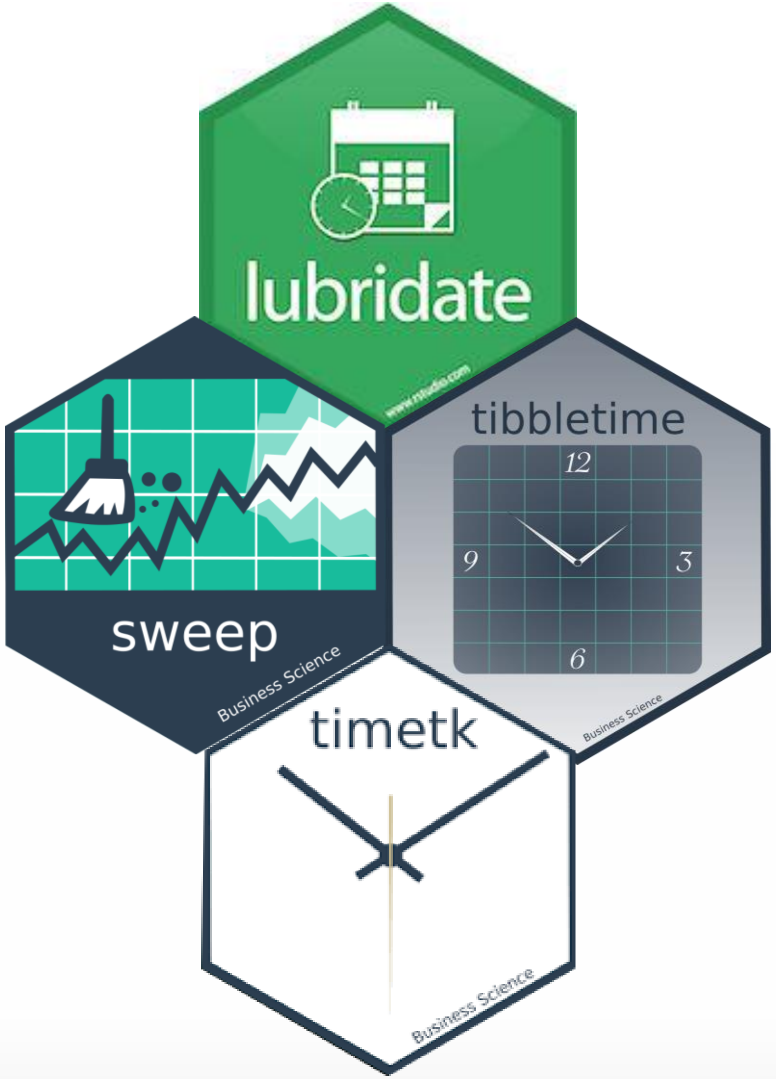
- Date
- Time
- Date-time

`tibbletime` package

- Data manipulation

`timetk` & `sweep`

- Coerce between classes
- Tidy forecasting data

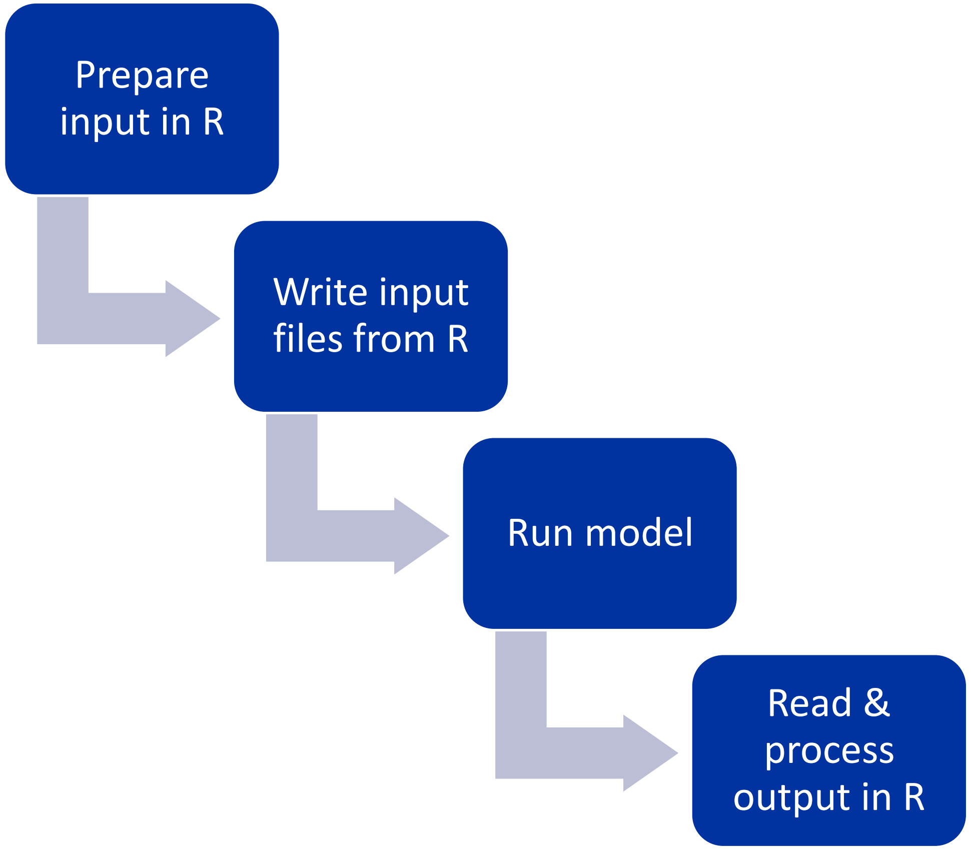


Using R as a GIS

- Spatial data: data with a spatial component (i.e. coordinate)
- Vectors (`sf`) & rasters (`raster`)
- Perform data wrangling & spatial analysis in the same environment
- Interactive plotting & html widget capabilities (`leaflet`)

Pre & postprocessers of models

- Preparing input & processing output in R
    - Handling data
- Calibration in R
    - More versatility
- Still in experimental phase
    - Non-uniformity



Prepare input in R

Write input files from R

Run model

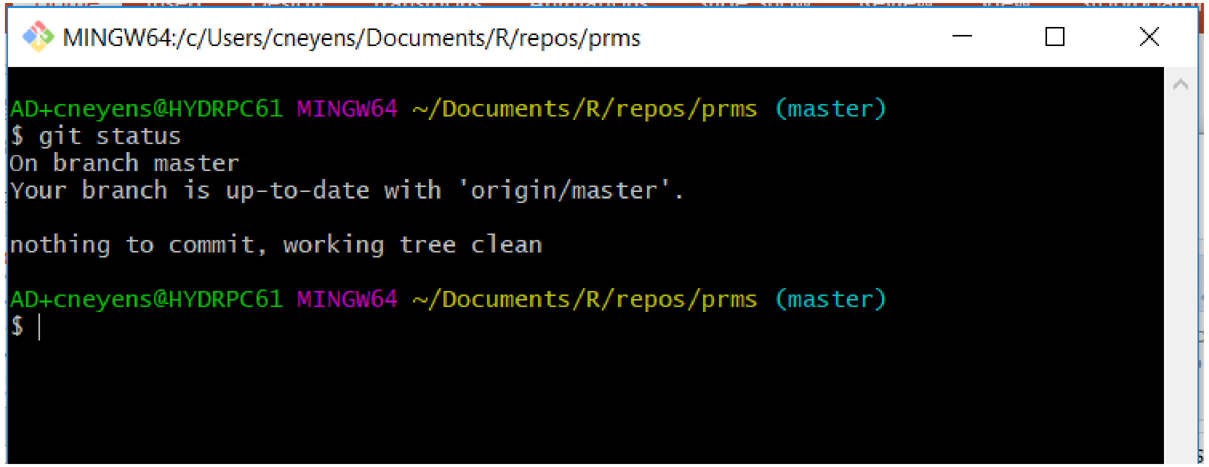
Read & process output in R

Pre & postprocessors of models

- Supported models (that I use):
  - PhreeqC (limited)
  - Hydrus (limited)
  - PRMS (limited; focusses on calibration in R)
  - MODFLOW (in development)
  - MT3DMS (limited; in development)
- A lot of room for improvement/packages
- Example: `RMODFLOW` (Rogiers, 2016)
  - Vignettes

GitHub

- `git` = version control software
  - Source code management



```
AD+cneyens@HYDRPC61 MINGW64 ~/Documents/R/repos/prms (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.

nothing to commit, working tree clean

AD+cneyens@HYDRPC61 MINGW64 ~/Documents/R/repos/prms (master)
$ |
```

- GitHub: online repository hosting
  - Free
  - Public repo’s
  - Allows issue reporting, pull requests, documentation, ….
  - Social interactions

GitHub

- HYDR GitHub (https://github.com/VUB-HYDR):
    - Codes: WetSpa, WetSpass, …
    - Tools: useful scripts (e.g. slugtest analysis)
    - Presentations & vignettes
    - Data analysis of publications
- In development
- Contact: cas.neyens@vub.be

Questions ?

cas.neyens@vub.be

VUB VRIJE UNIVERSITEIT BRUSSEL

KU LEUVEN

IUPWARE

Interuniversity Programme in Water Resources Engineering