
```
library(tidyverse)
```

Data Wrangling in R

Ben Best ben@ecoquants.com

MBON Pole to Pole Brazil Workshop

2018-08-07

slides: bit.ly/r-wrangle-for-p2p

Motivation

- MBON Pole to Pole: develop a “Community of Practice”
  - Best practices
  - Common tools
- Manipulate (ie “wrangle”) data to:
  - Check quality
  - Analyze
  - Visualize
  - Publish
    - eg OBIS DarwinCore

slides: bit.ly/r-wrangle-for-p2p

How many of you have used these?

1. R
   scientific programming language
2. RStudio
   integrated development environment (IDE)
3. dplyr
   R package for grammar of data manipulation
4. rmarkdown
   authoring framework for data science
5. git
   version control system
6. Github
   web hosting service for git to collaborate

slides: bit.ly/r-wrangle-for-p2p

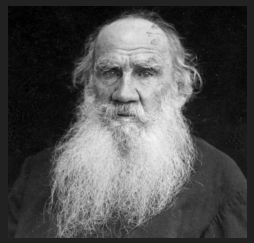
For Big-Data Scientists, ‘Janitor Work’ Is Key Hurdle to Insights

nytimes.com/2014/08/18/technology/for-big-data-scientists-hurdle-to-insights-is-janitor-work.html

“Data scientists, according to interviews and expert estimates, spend from 50 to 80 percent of their time mired in the mundane labor of collecting and preparing data, before it can be explored for useful information.”

— NY Times (2014)

“Happy families are all alike; every unhappy family is unhappy in its own way.”
— Leo Tolstoy



“Tidy datasets are all alike, but every messy dataset is messy in its own way.”
— Hadley Wickham

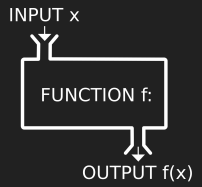
Tidy Manifesto

cran.r-project.org/web/packages/tidyverse/vignettes/manifesto.html

1. Reuse existing data structures

2. Compose simple functions with the pipe

3. Embrace functional programming

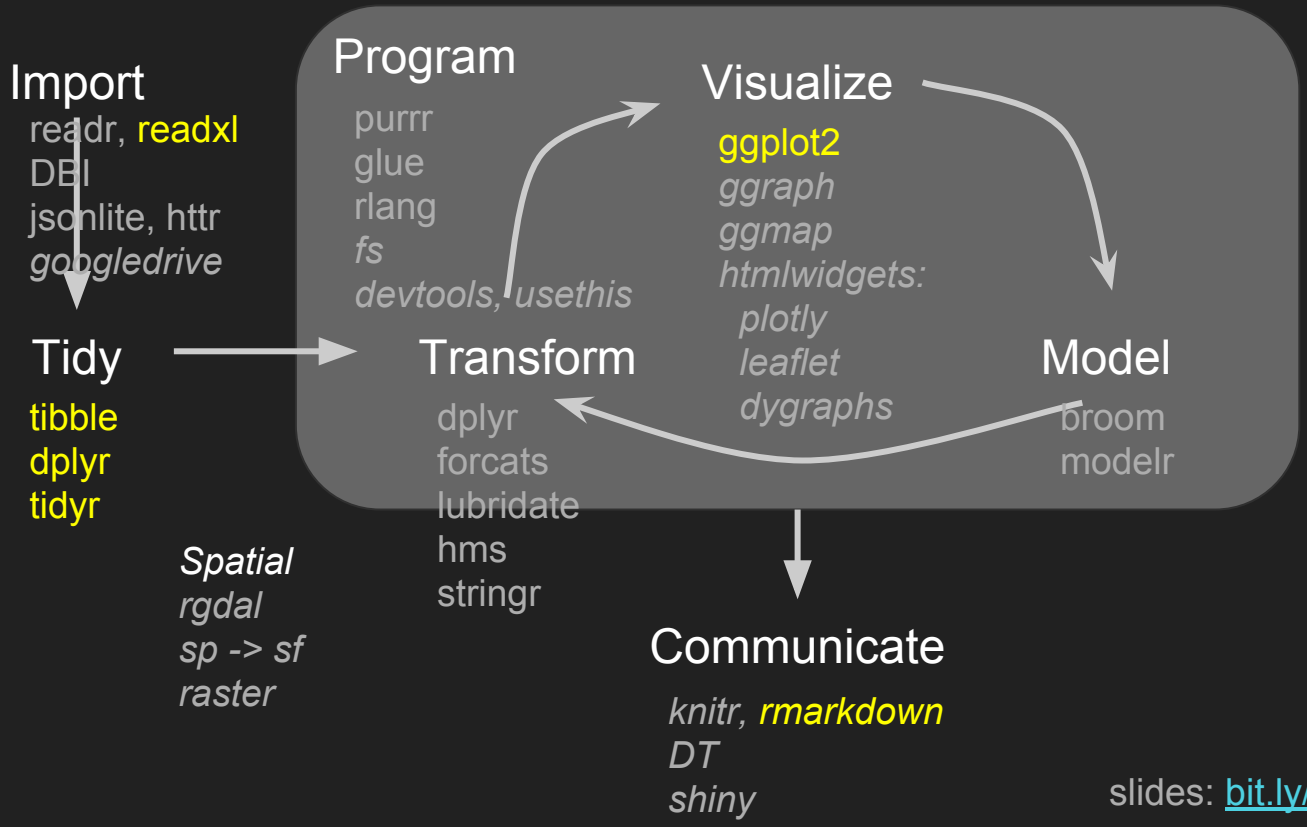


4. Design for humans

”

blog.codinghorror.com/falling-into-the-pit-of-success

Tidyverse process & packages *(unofficial in italics)*



slides: bit.ly/r-wrangle-for-p2p

Which package do I use and how?

CRAN Task Views

cran.r-project.org/web/views

Cheat Sheets

rstudio.com/resources/cheatsheets

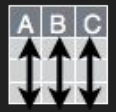
- RStudio IDE
- R Markdown
- Data Import (tidyr)
- Data Transformation (dplyr)

Data Import cheat sheet

Tidy data organizes tabular data (as long) for use across R packages:

- Each **observation** is in its own **row**

- Each **variable** is in its own **column**



`gather(``data, key, value`**`)`**

wide -> long

country	1999	2000
A	0.7K	2K
B	37K	80K
C	212K	213K

→

country	year	cases
A	1999	0.7K
B	1999	37K
C	1999	212K
A	2000	2K
B	2000	80K
C	2000	213K

key value

`spread(``data, key, value`**`)`**

long -> wide

country	year	type	count
A	1999	cases	0.7K
A	1999	pop	19M
A	2000	cases	2K
A	2000	pop	20M
B	1999	cases	37K
B	1999	pop	172M
B	2000	cases	80K
B	2000	pop	174M
C	1999	cases	212K
C	1999	pop	1T
C	2000	cases	213K
C	2000	pop	1T

key value

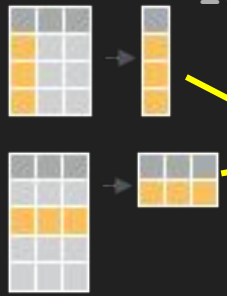
→

country	year	cases	pop
A	1999	0.7K	19M
A	2000	2K	20M
B	1999	37K	172M
B	2000	80K	174M
C	1999	212K	1T
C	2000	213K	1T

Data Transformation cheat sheet

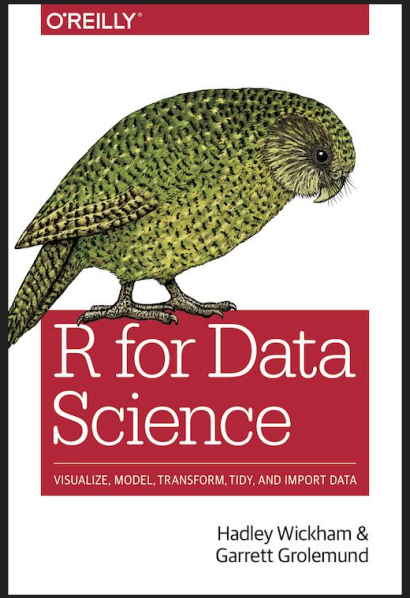
dplyr is a grammar of data manipulation, providing a consistent set of verbs that help you solve the most common data manipulation challenges. (works w/ db's)

pipes, aka "then" operator **%>%** are [elegant]



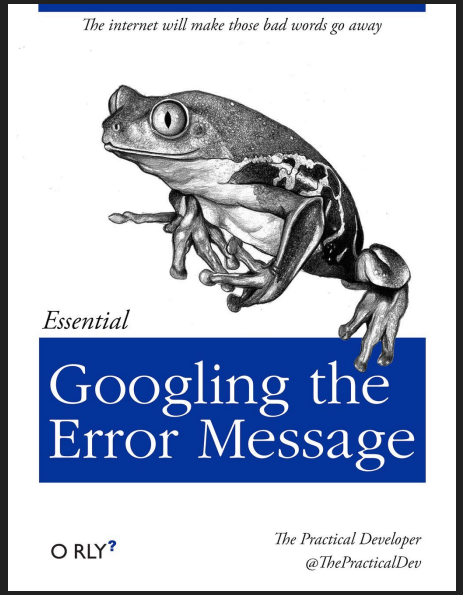
```
read_csv('data/surveys.csv') %>%      # read data
  mutate(yr = year(obs_date)) %>%     # add new column
  select(species_id, yr) %>%          # limit columns
  filter(species_id  == 'NL') %>%     # limit rows
  group_by(yr) %>%                    # group by values in column
  summarise(n_obs = n()) %>%          #   then summarise by group
  arrange(desc(n_obs)) %>%            # sort in descending order
  write_csv('data/surveys_clean.csv') # write out csv
```

Resources

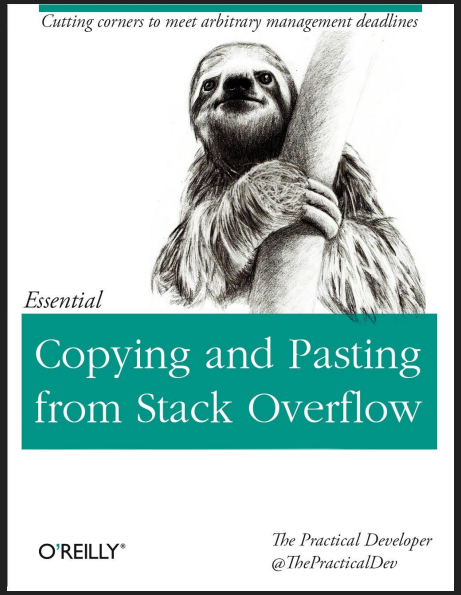


[r4ds.had.co.nz](r4ds.had.co.nz)

the "bible"



google.com



stackoverflow.com

slides: bit.ly/r-wrangle-for-p2p

rstudio::conf 2017 at Harry Potter World, FL

Demo(s)

github.com/bbest/p2p-demo (TODO together)

Examples:

- bbest.github.io/ioos-bio-tidyr
- marinebon.github.io/info-intertidal
- marinebon.github.io/sbc-datasets
- bbest.github.io/bbest/p2p-demo-1

slides: bit.ly/r-wrangle-for-p2p