Intro to Tidyverse!

Janani Ravi

jravilab.github.io | @janani137 | janani@msu.edu

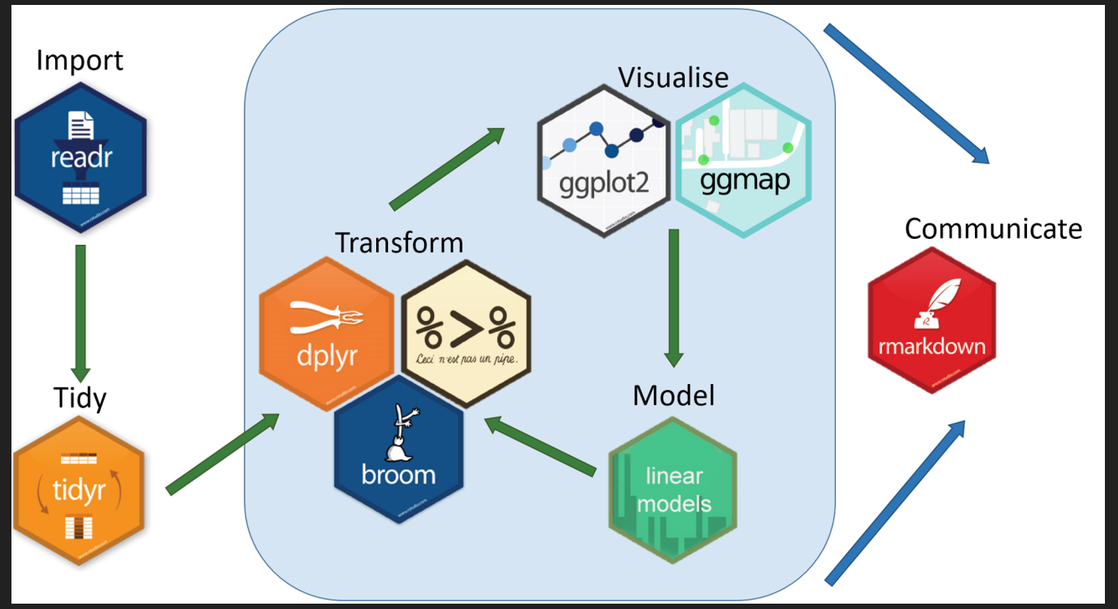
Intro

- **Familiar w/ R and ggplot? Introduce yourself to your neighbor**
  - Who you are | Name, affiliation
  - Do you have the same version of
    - R (3.6+), RStudio & Tidyverse?
    - NO? Installation time!

Need help? Raise your hand!  **OR ask on chat!** 

Today!

- **Workshop**
  - Intro to Tidyverse



Welcome to *tidyverse*

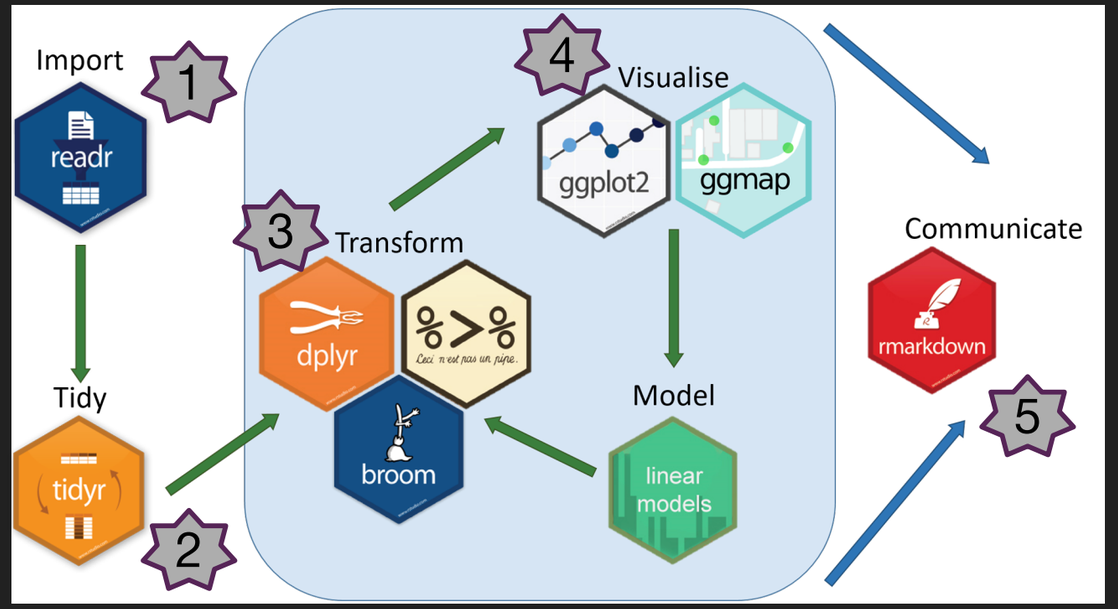
P1: Getting Started w/ readr

P2: Reshaping data w/ tidyr

P3: Data wrangling w/ dplyr

P4: DataViz w/ ggplot

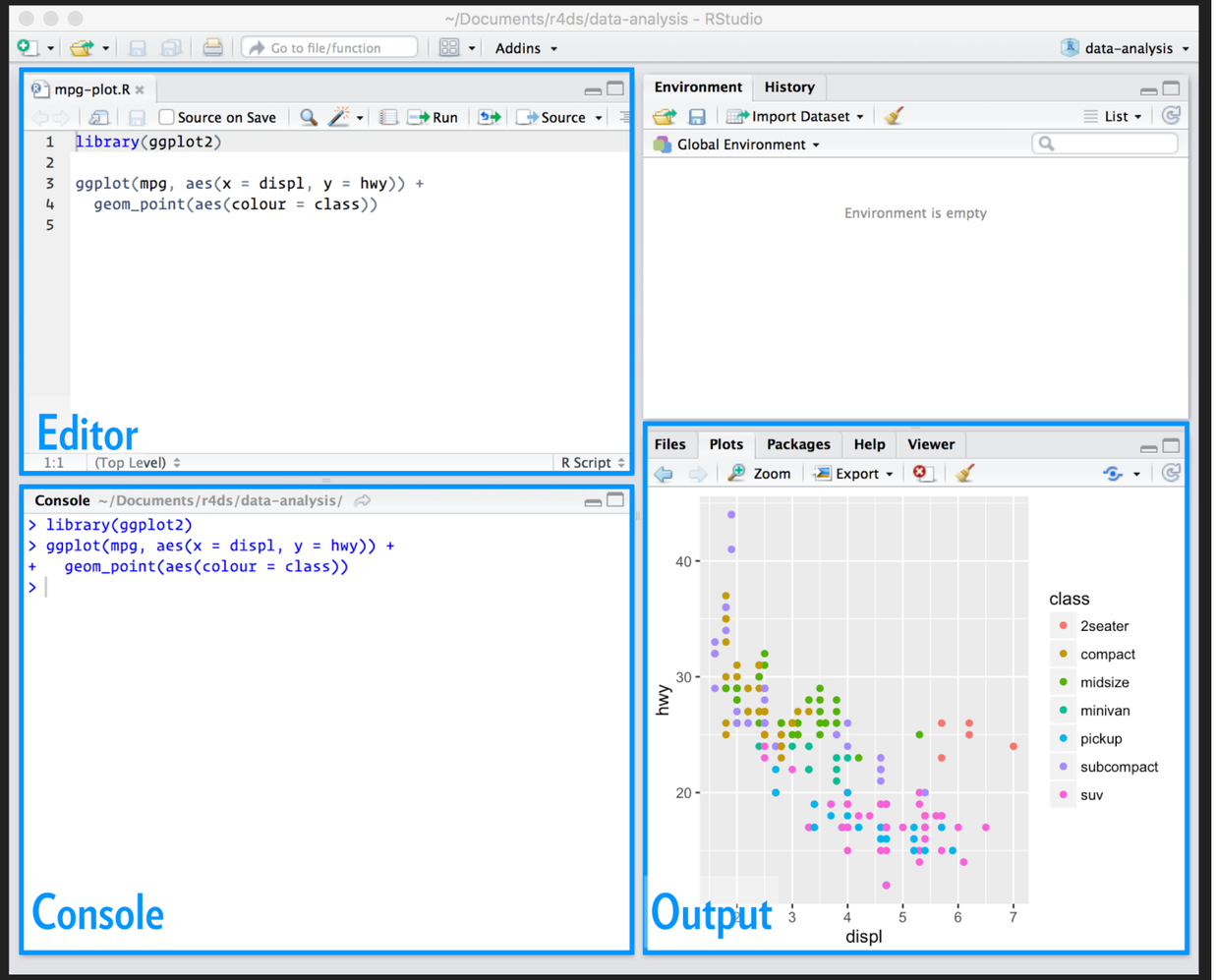
P5: Wrap-up w/ RMarkdown



Part 1: Getting Started: Environment

- ✓ Installing RStudio, R
- ✓ Installing tidyverse

```
> library(tidyverse)
── Attaching packages ─────────────────────── tidyverse 1.2.1 ──
✔ ggplot2 3.0.0     ✔ purrr   0.2.5
✔ tibble  1.4.2     ✔ dplyr   0.7.6
✔ tidyr   0.8.1     ✔ stringr 1.3.1
✔ readr   1.1.1     ✔ forcats 0.3.0
── Conflicts ────────────────────────── tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
```



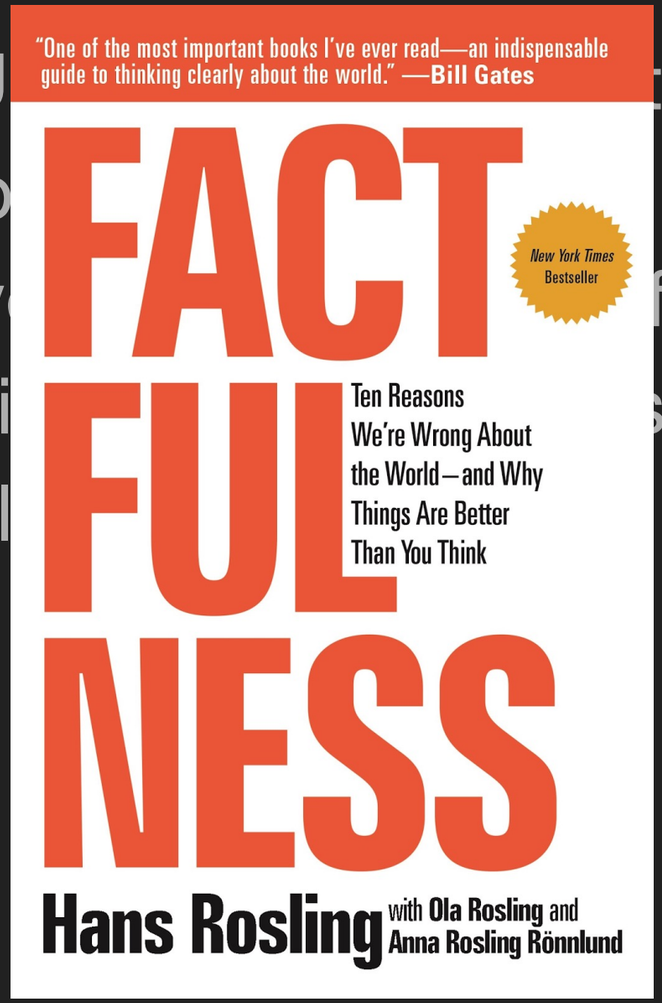
Getting Started: Data, your data

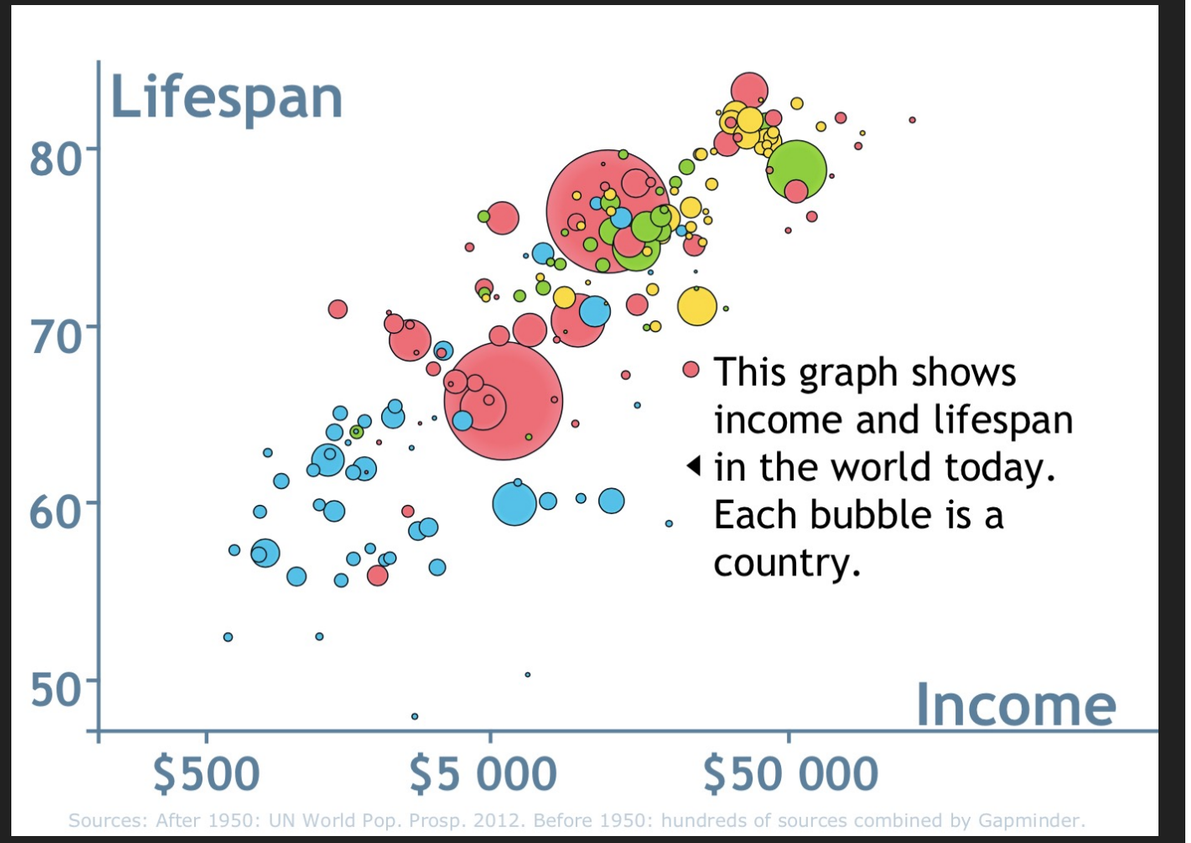
1. Import your data

```
library(tidyverse)
read_csv(file="my_data.csv",
                    col_names=T)        # comma-separated values
read_delim(file="my_data.txt", col_names=T,
                       delim="//")     # any delimitter
# Other useful packages
# readxl by Jenny Bryan
read_excel(path="path/to/excel.xls",
                       sheet=1,
                       range="A1:D50",
                       col_names=T)
```

Getting Started: Today's Dataset

2. U ts e.g.,
Gap
Unv f
stati sed
worl





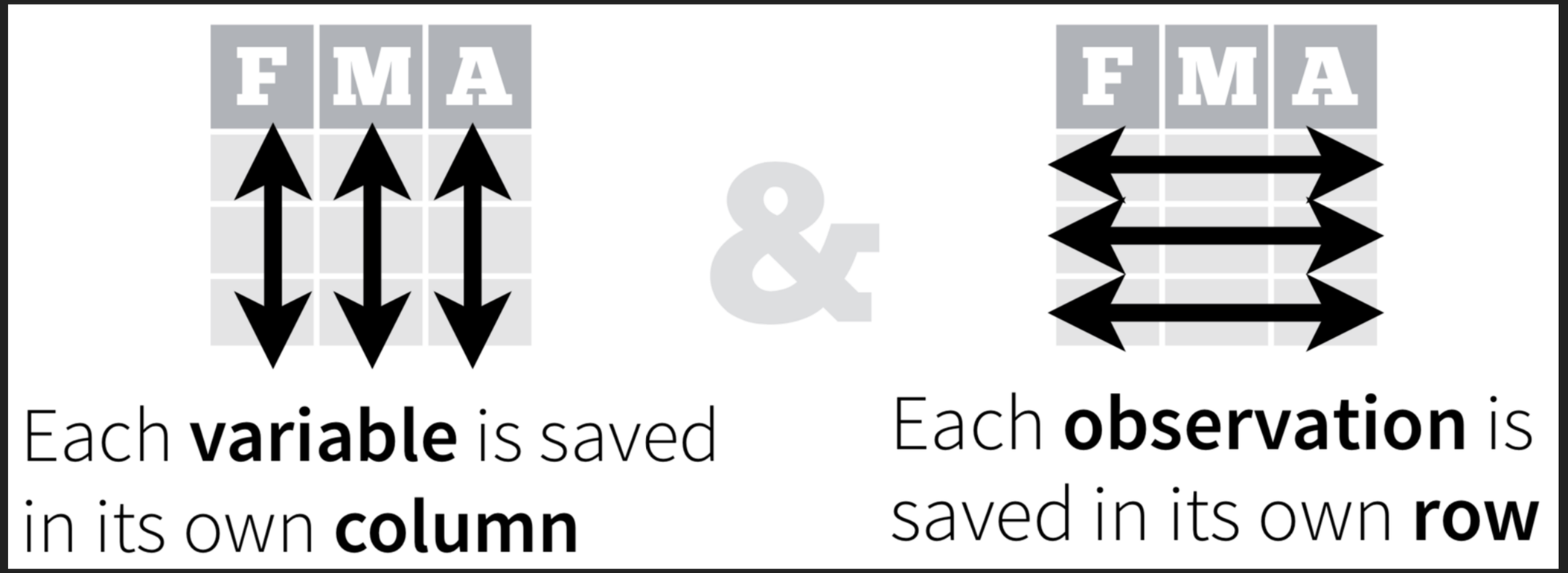
https://www.gapminder.org/

Getting Started: Knowing your data

```
# gapminder::gapminder
str(gapminder)        # Structure of the dataframe
gapminder             # Data is in a cleaend up 'tibble' format by default
head(gapminder)       # Shows the top few observations (rows) of your data frame
glimpse(gapminder)    # Info-dense summary of the data
View(gapminder)       # View data in a visual GUI-based spreadsheet-like format
```

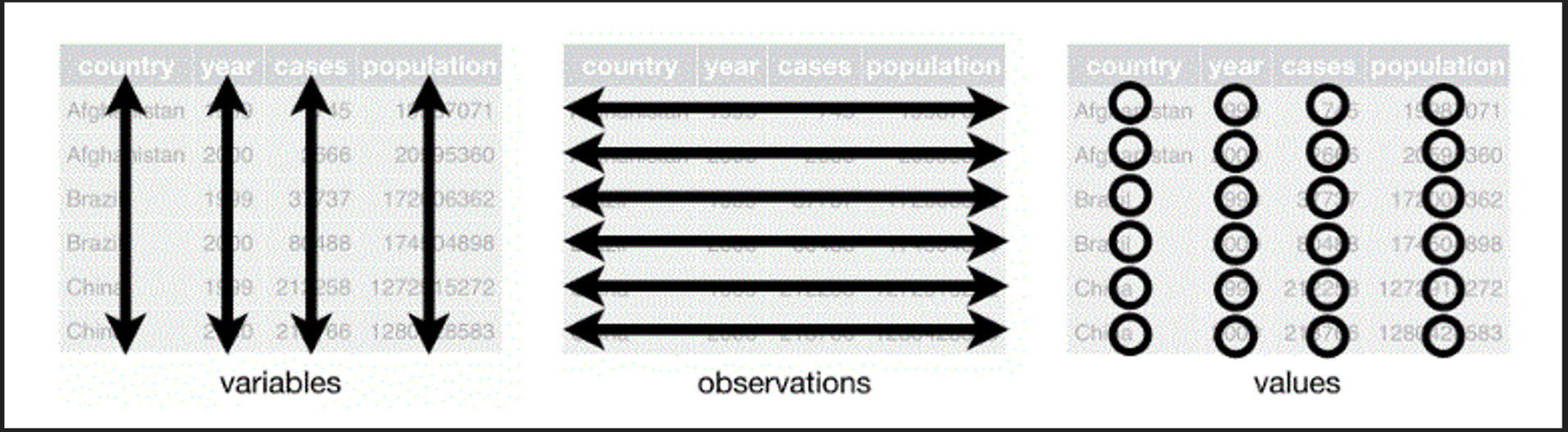
Back to RStudio

Part 2: Reshaping data w/ *tidyr*



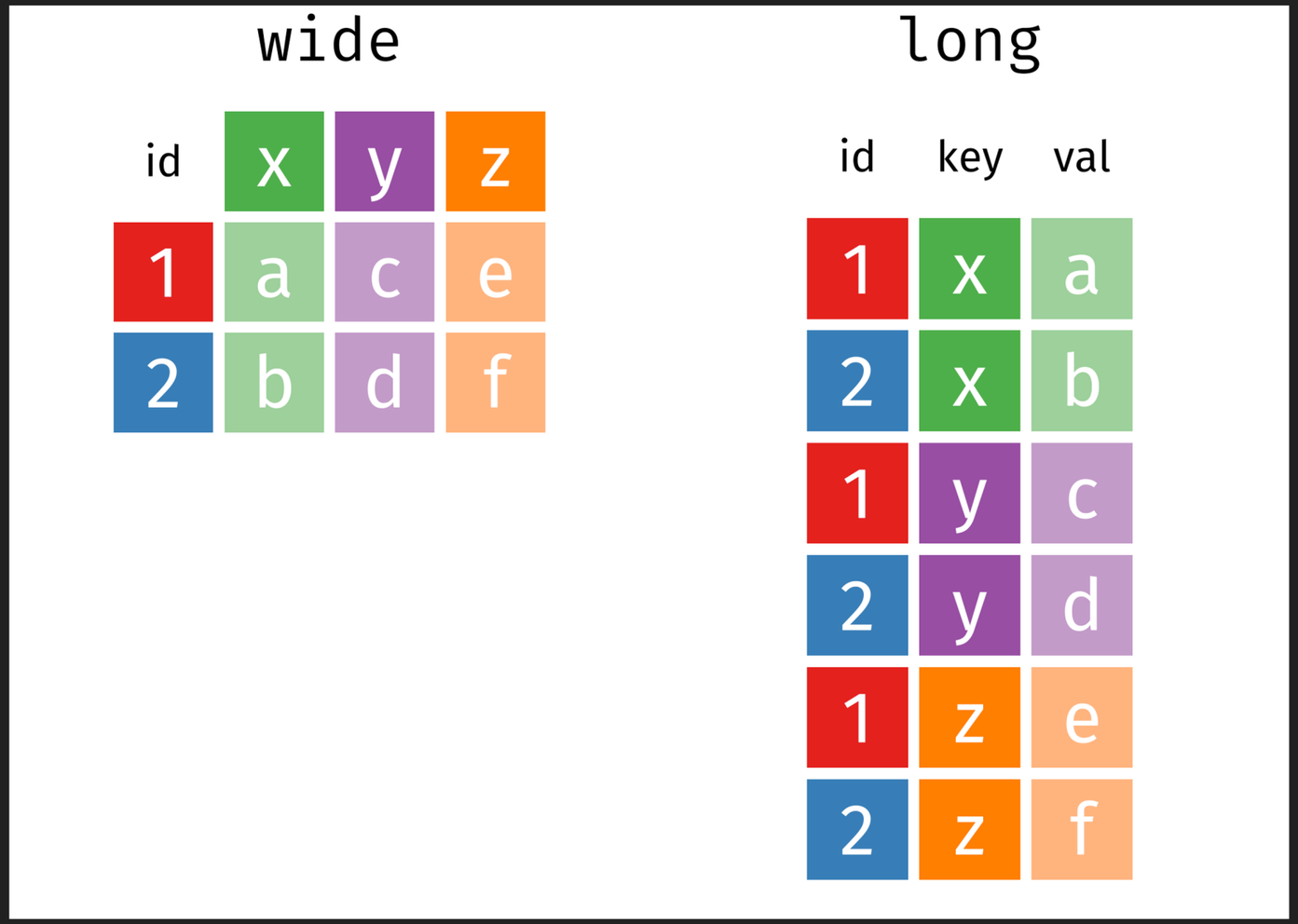
Tidy Data … In a ‘Tidy’ dataset, …

Part 2: What is tidy data?



- Each variable in the data set is placed in its own column.
- Each observation is placed in its own row.
- Each value is placed in its own cell.

Part 2: In a ‘Tidy’ dataset, …

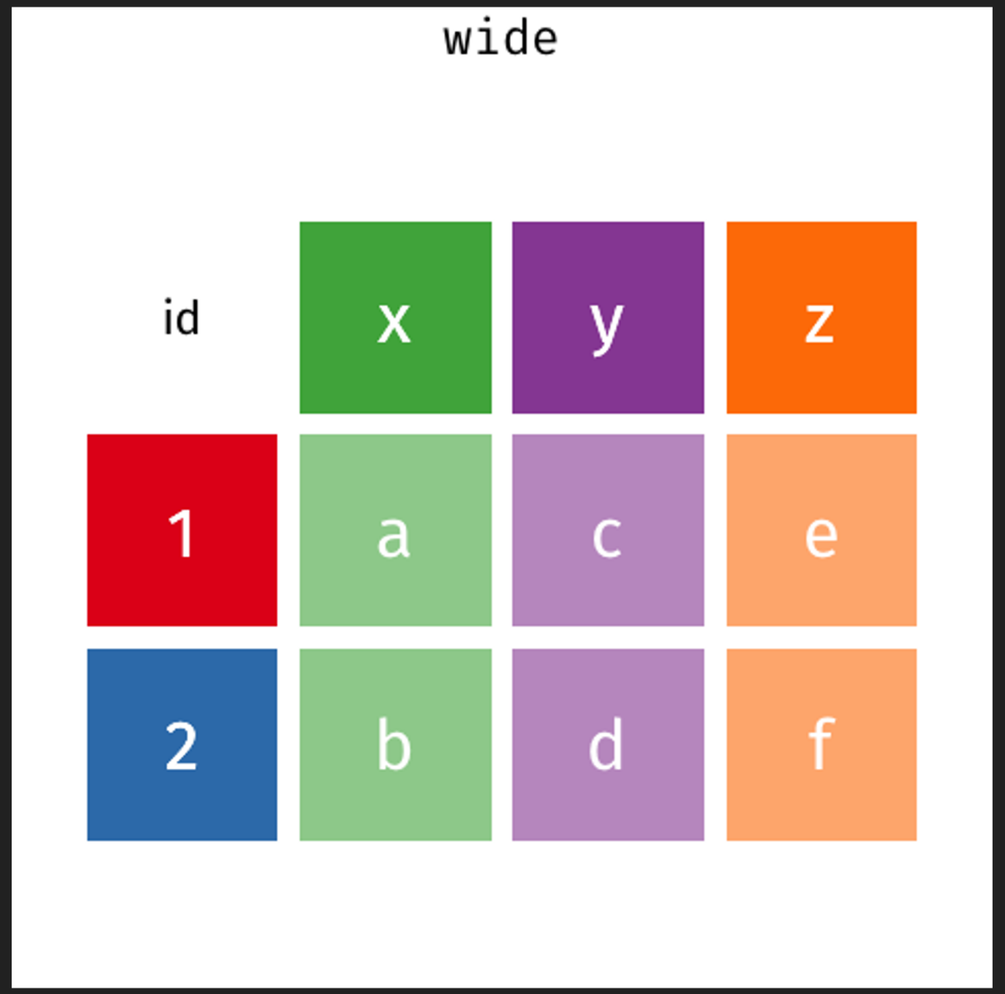


Part 2: Reshaping data w/ *tidyr*

```
pivot_longer()    # Gather COLUMNS -> ROWS
pivot_wider()     # Spread ROWS -> COLUMNS
separate()        # Separate 1 COLUMN -> many COLUMNS
unite()           # Unite several COLUMNS -> 1 COLUMN
```

Part 2: Reshaping data w/ *tidyr*

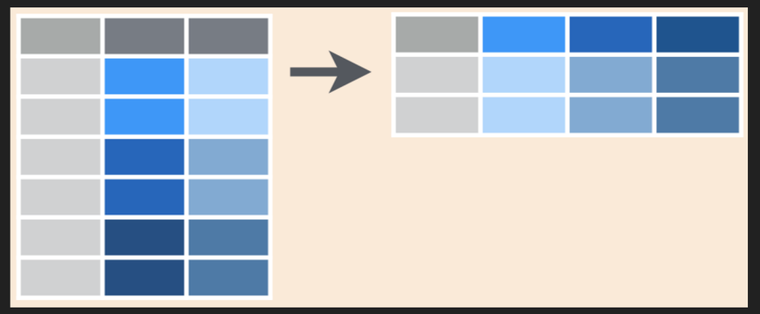
Tidy data



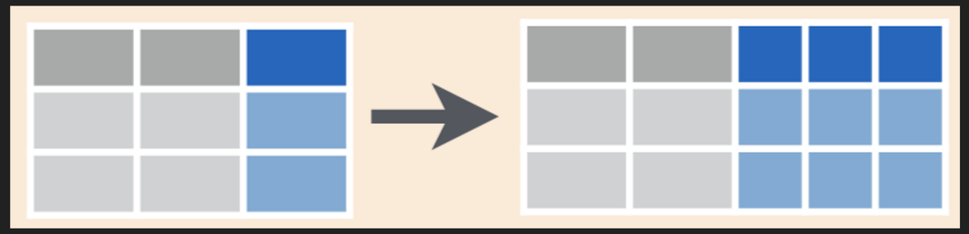
gadenbuie/tidyexplain

Part 2: Reshaping data w/ *tidyr*

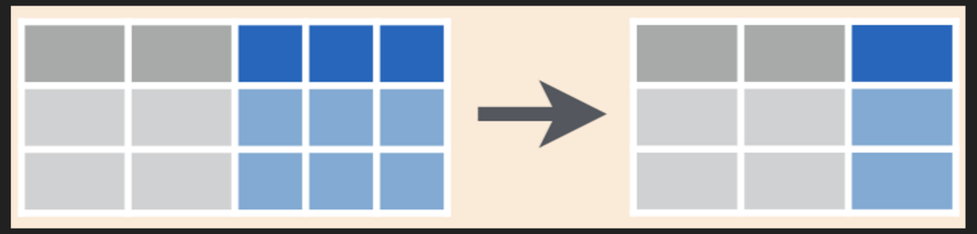
pivot_longer



pivot_wider



separate



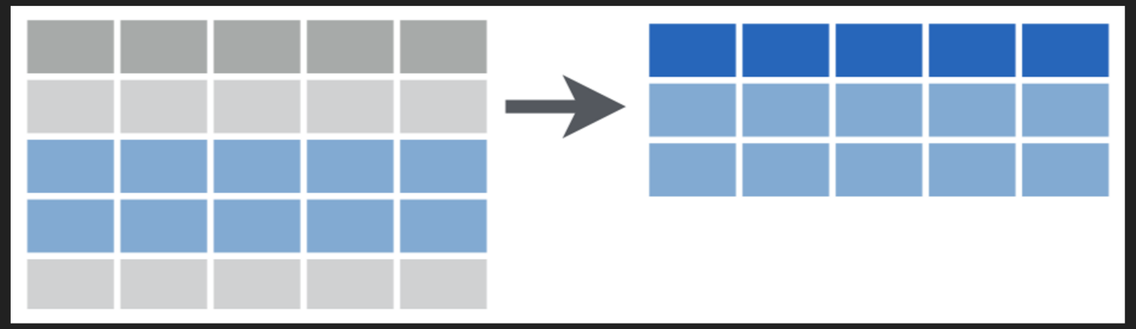
unite

Back to RStudio

Part 3: Wrangling data w/ *dplyr*

```
filter()    # PICK observations by their values | ROWS
select()    # PICK variables by their names | COLUMNS
mutate()    # CREATE new variables w/ functions of existing variables | COLUMNS
transmute() # COMPUTE 1 or more COLUMNS but drop original columns
arrange()   # REORDER the ROWS
summarize() # COLLAPSE many values to a single SUMMARY
group_by()  # GROUP data into rows with the same value of variable (COLUMN)
```

Part 3: Wrangling data w/ *dplyr*



filter

select

Part 3: Wrangling data w/ *dplyr*

summarise

group_by

Part 3: Wrangling data w/ *dplyr*

Mutating joins

a

x1	x2
A	1
B	2
C	3

+

b

x1	x3
A	T
B	F
D	T

=

left_join()

x1	x2	x3
A	1	T
B	2	F
C	3	NA

right_join()

x1	x3	x2
A	T	1
B	F	2
D	T	NA

inner_join()

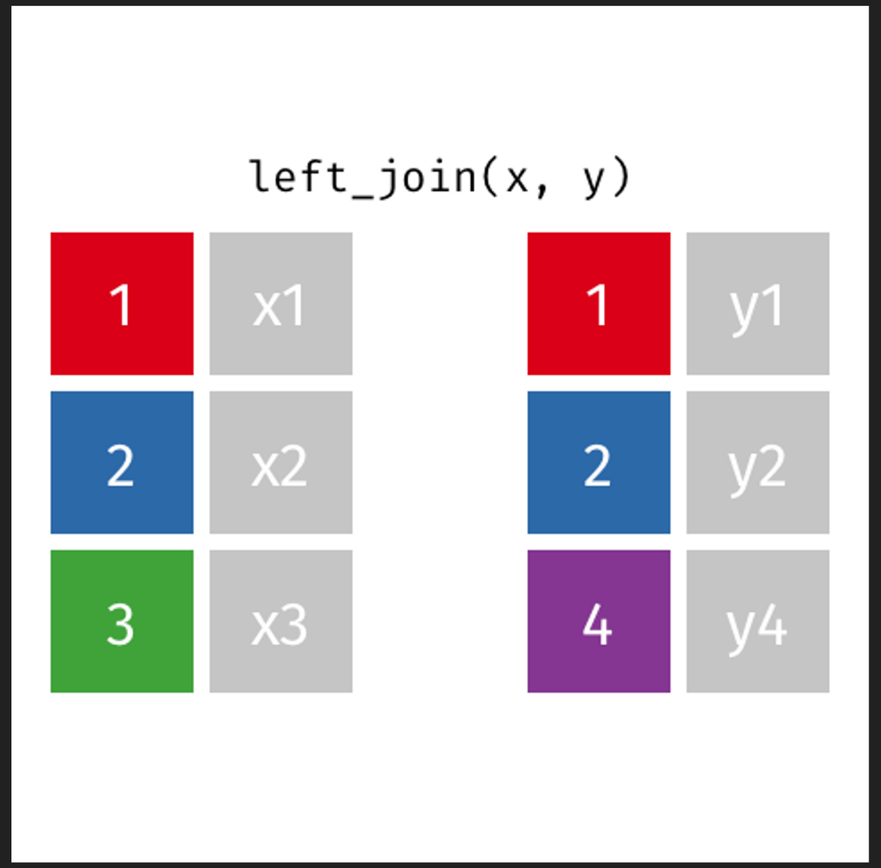
x1	x2	x3
A	1	T
B	2	F

outer_join()

x1	x2	x3
A	1	T
B	2	F
C	3	NA
D	NA	T

Part 3: Wrangling data w/ *dplyr*

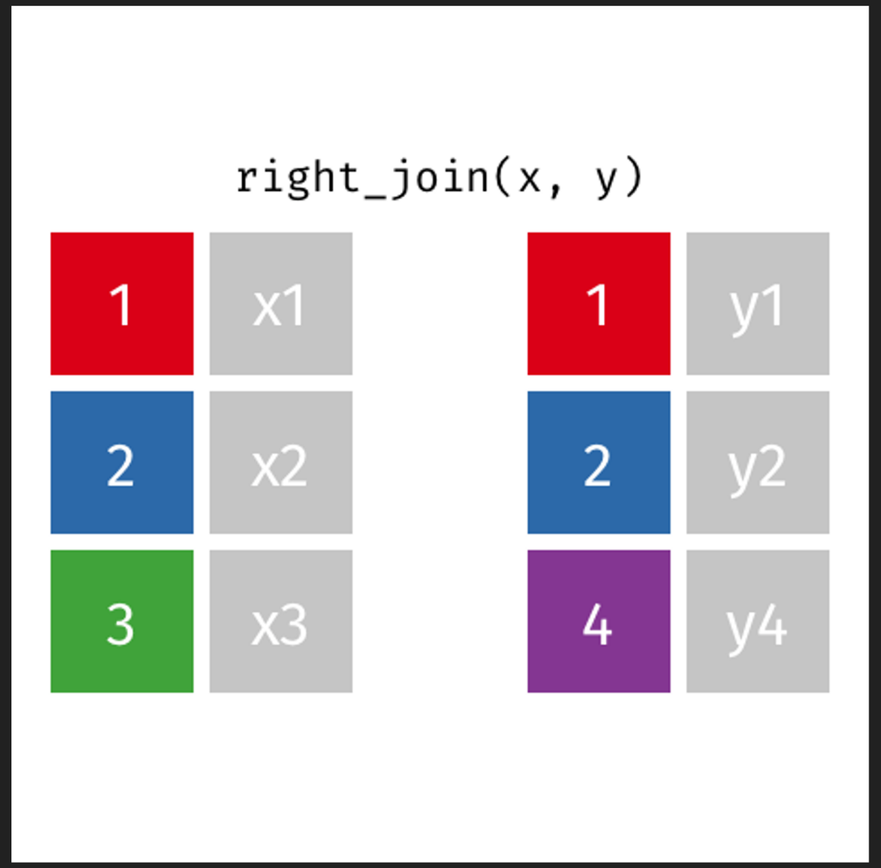
Mutating joins



gadenbuie/tidyexplain

Part 3: Wrangling data w/ *dplyr*

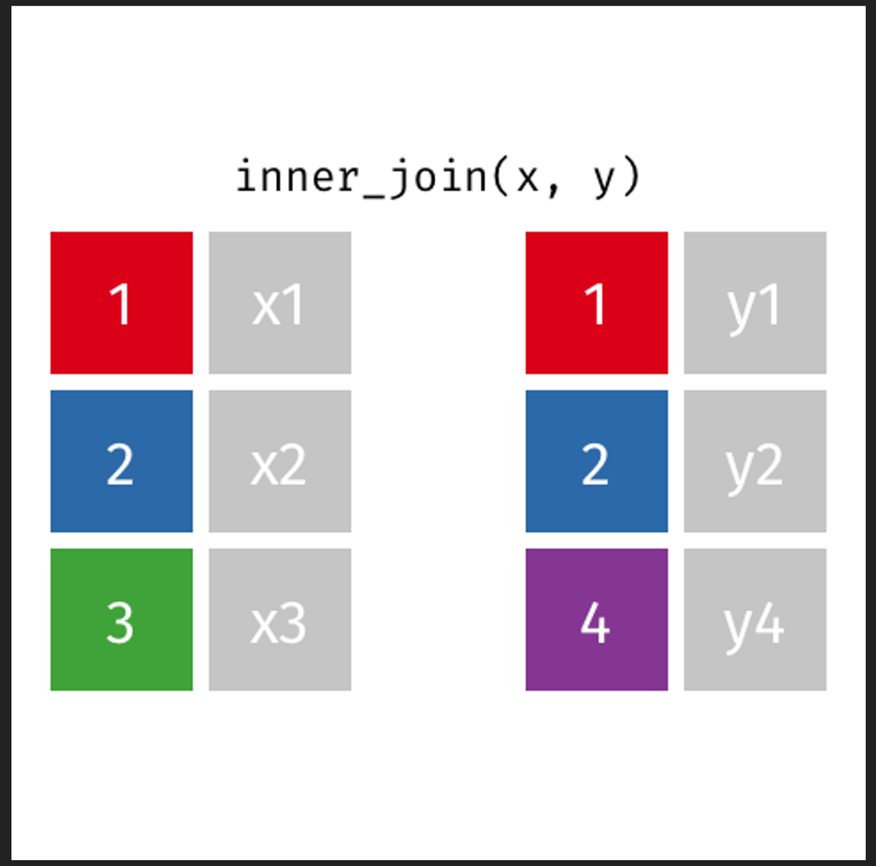
Mutating joins



gadenbuie/tidyexplain

Part 3: Wrangling data w/ *dplyr*

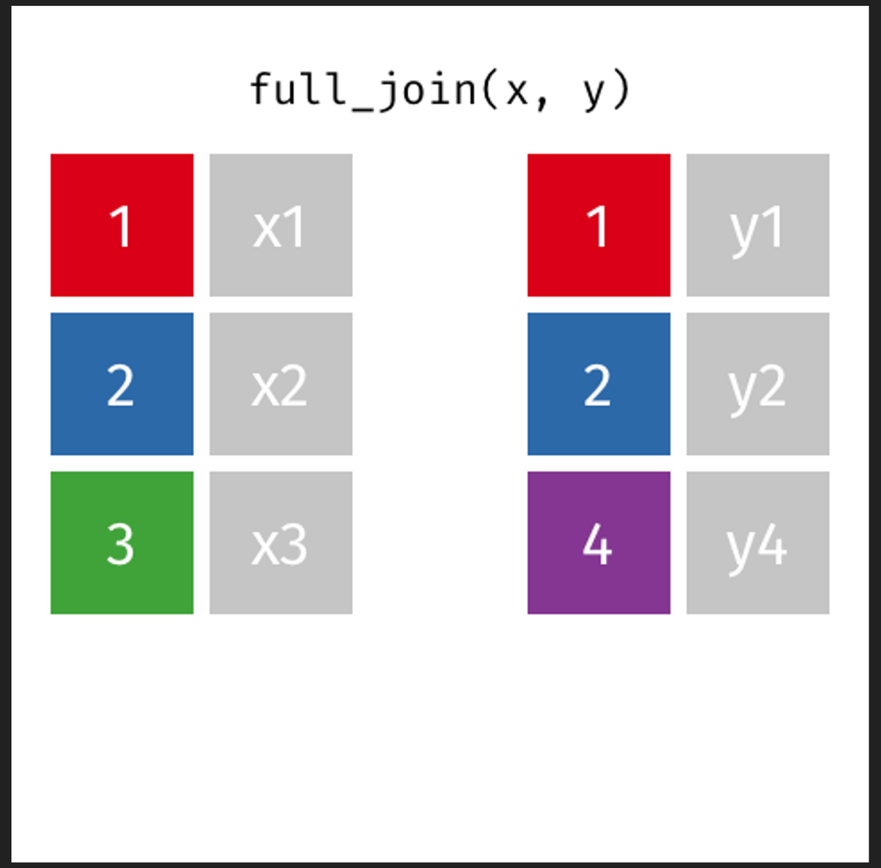
Mutating joins



gadenbuie/tidyexplain

Part 3: Wrangling data w/ *dplyr*

Mutating joins



gadenbuie/tidyexplain

Part 3: Wrangling data w/ *dplyr*

Set operations

y

x1	x2
A	1
B	2
C	3

+

z

x1	x2
B	2
C	3
D	4

=

intersect()

x1	x2
B	2
C	3

union()

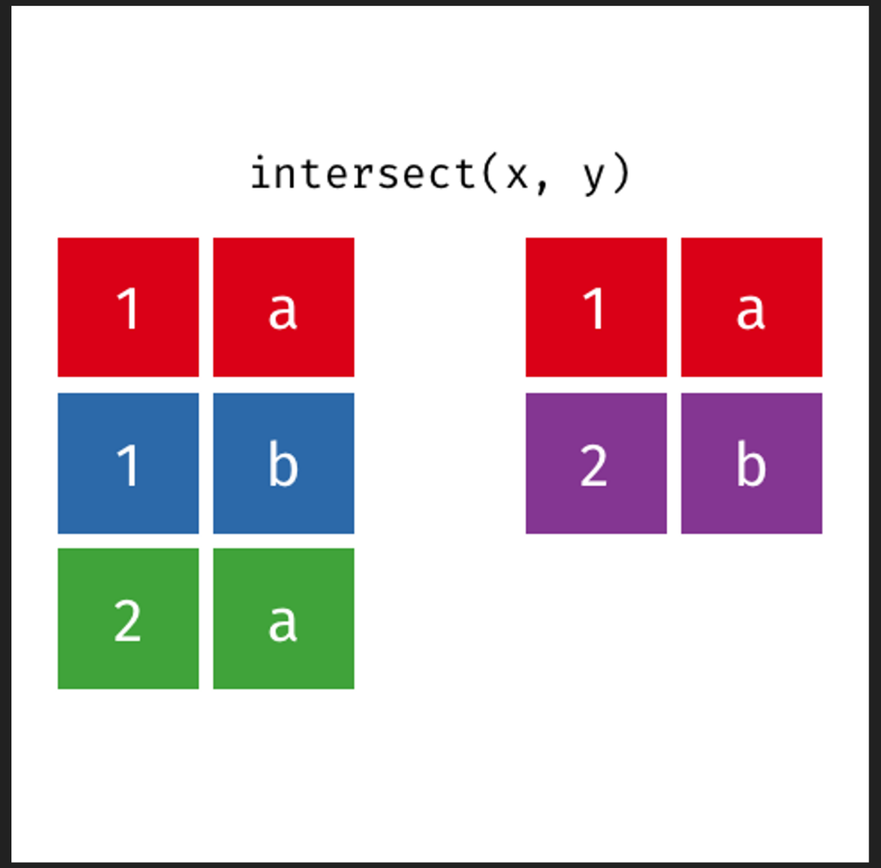
x1	x2
A	1
B	2
C	3
D	4

setdiff()

x1	x2
A	1
D	4

Part 3: Wrangling data w/ *dplyr*

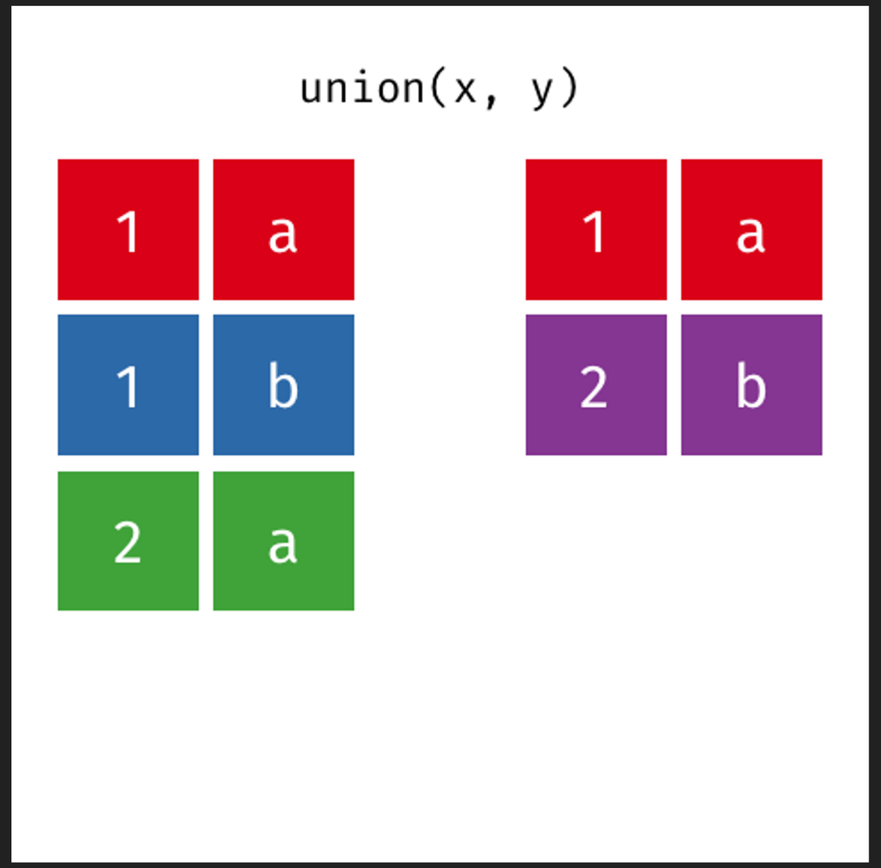
Set operations



gadenbuie/tidyexplain

Part 3: Wrangling data w/ *dplyr*

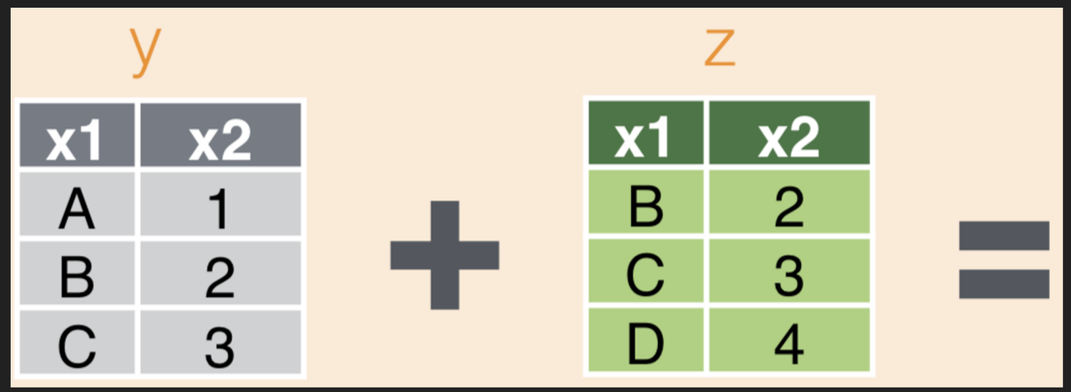
Set operations



gadenbuie/tidyexplain

Part 3: Wrangling data w/ *dplyr*

Binding



bind_rows()

bind_cols()

x1	x2
A	1
B	2
C	3
B	2
C	3
D	4

x1	x2	x1	x2
A	1	B	2
B	2	C	3
C	3	D	4

Back to RStudio

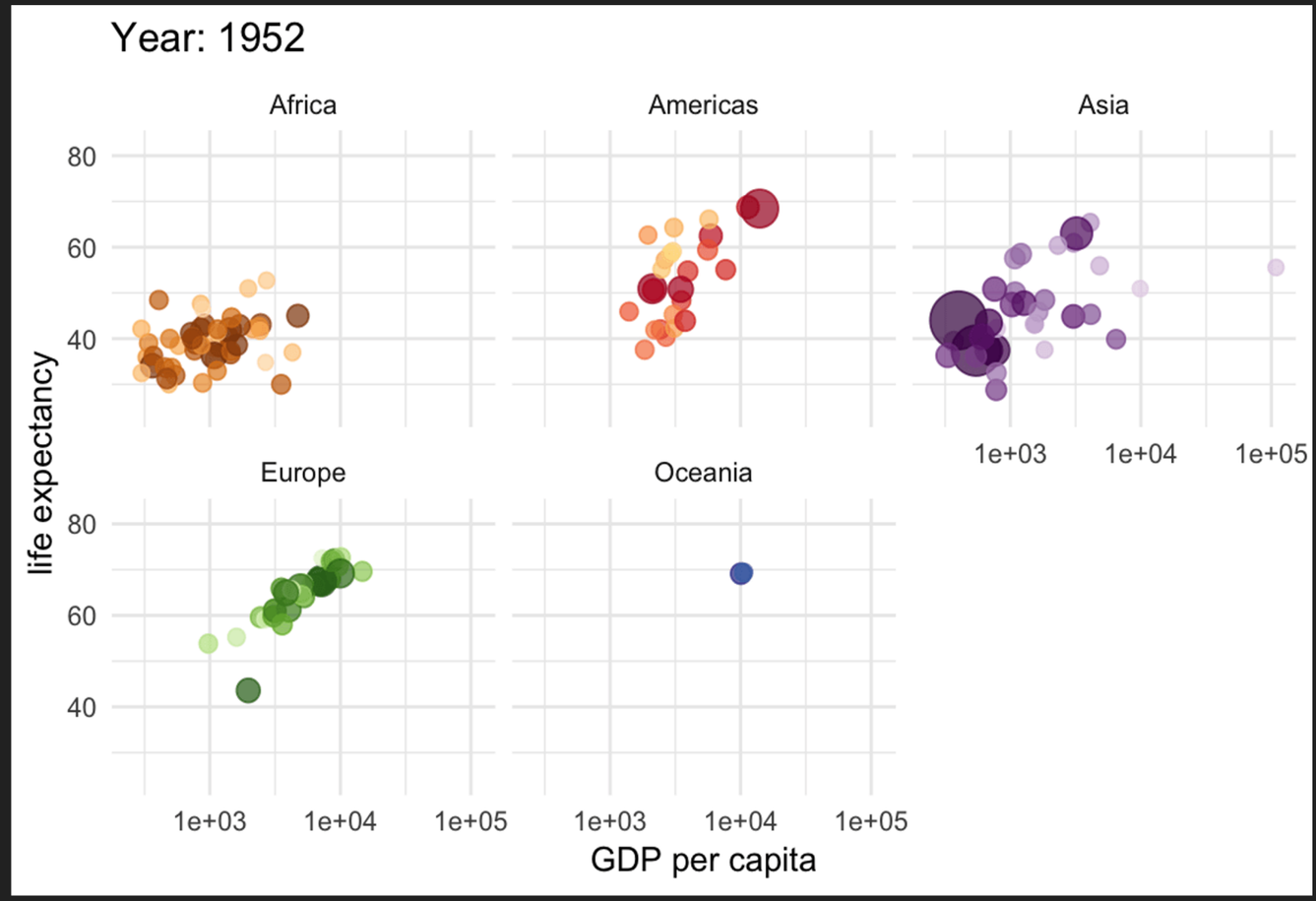
Part 4: Visualizing tidy data w/ ggplot2

Geometry of graphics

- **data**: Must be a data frame
- **aes**thetics: How your data are represented visually
  - x, y, color, size, shape, etc.
- **geom**etry: Geometries of plotted objects
  - points, lines, boxplot, polygons, etc.
- and *other customizations*

Part 4: Visualizing tidy data w/ ggplot2

gganimate: aha!

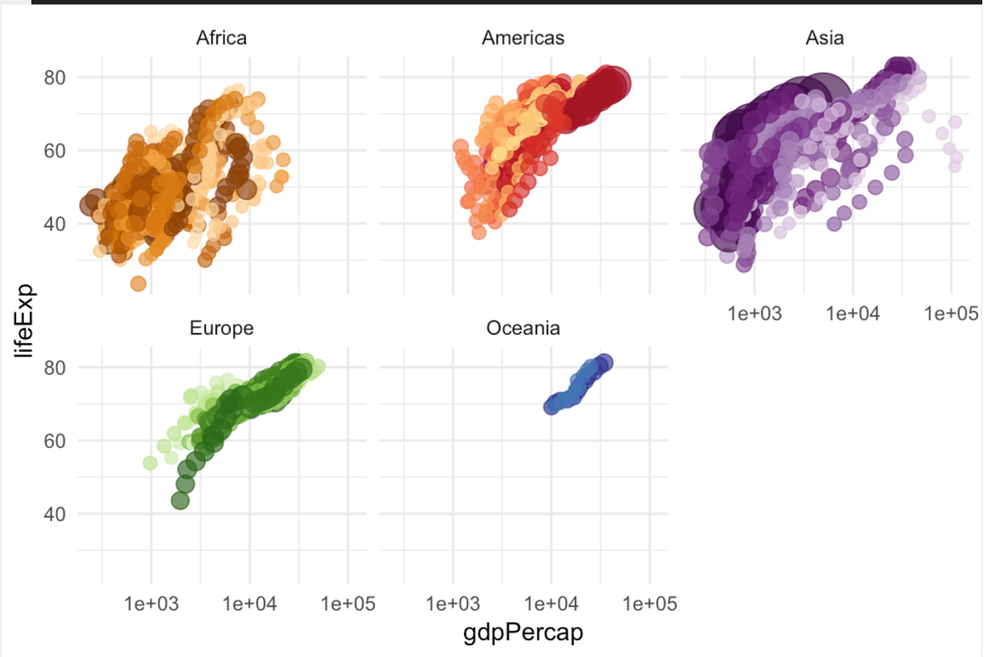


thomasp85/
gganimate

Part 4: Visualizing tidy data w/ ggplot2

gapminder: static plot

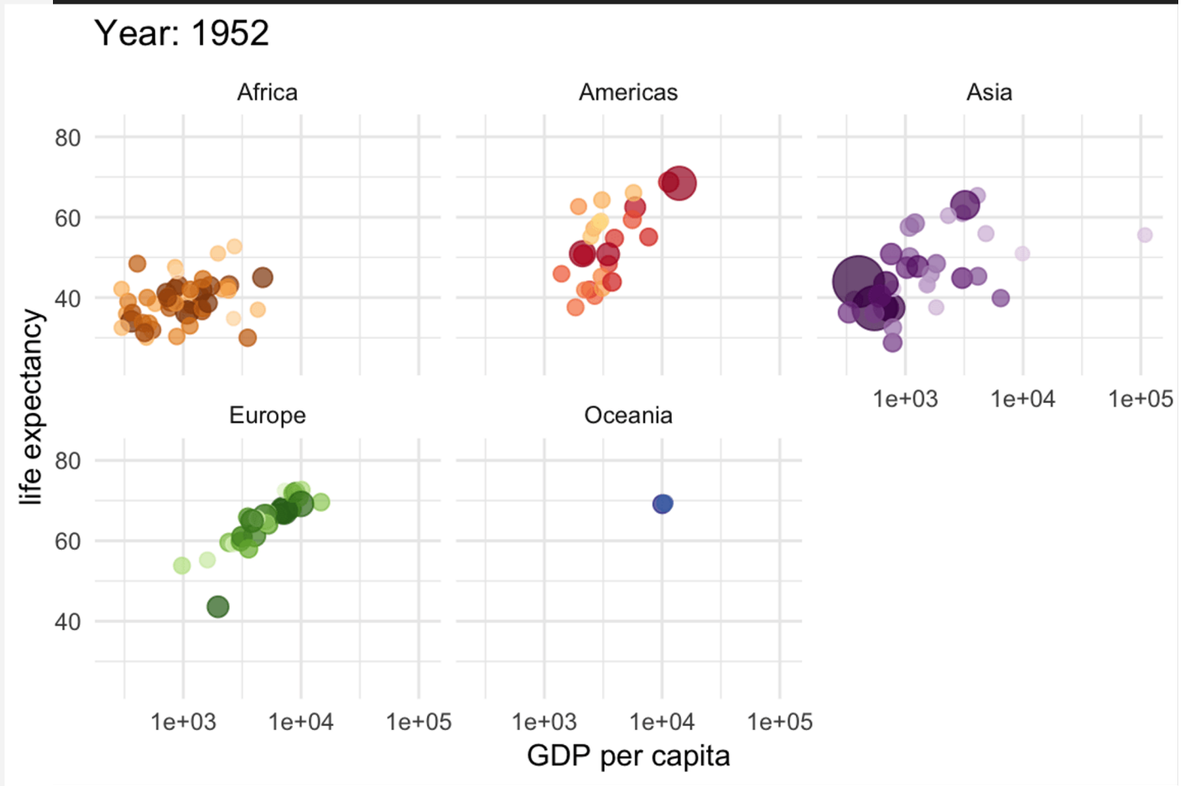
```
library(gapminder)
ggplot(gapminder,
       aes(gdpPercap, lifeExp,

           size=pop,colour=country)) +
  geom_point(alpha = 0.7,
                          show.legend = FALSE)
 +
  scale_colour_manual(values=country_colors) +
  scale_size(range=c(2, 12)) +
  scale_x_log10() +
  facet_wrap(~continent) +
  theme_minimal()
```



Part 4: Visualizing tidy data w/ ggplot2

gapminder: dynamic plot

```
ggplot(gapminder,
       aes(gdpPercap, lifeExp,
           size=pop, colour=country)) +
  … … … +
  theme_minimal() +
  # Here comes the gganimate part!
  labs(title = 'Year: {frame_time}',
       x = 'GDP per capita',
       y = 'life expectancy') +
  transition_time(year) +
  ease_aes('linear')
```



Part 5: Export & Wrap-up

ggsave — Save your plots

write_delim — Save your data

Back to RStudio

Tidyverse Recap

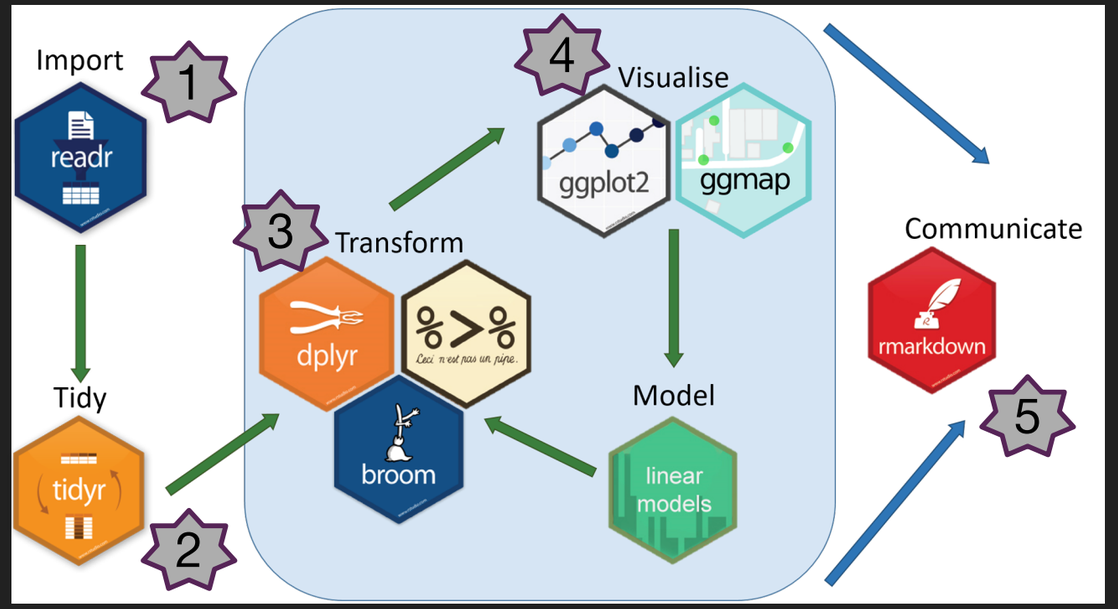
P1: Getting Started w/ readr ✓

P2: Reshaping data w/ tidyr ✓

P3: Data wrangling w/ dplyr ✓

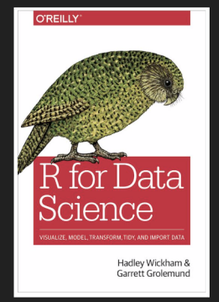
P4: DataViz w/ ggplot ✓

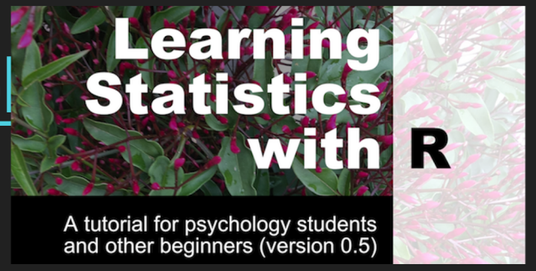
P5: Wrap-up w/ RMarkdown ✓



Resources

- **Hands-On Programming with R**: *Grolemund* **#HOPR**
  - https://rstudio-education.github.io/hopr/
- **R for Data Science:** *Wickham & Grolemund* **#R4DS**
  - https://r4ds.had.co.nz
- R Programming for Data Science: Peng
  - https://leanpub.com/rprogramming
- Learning Statistics with R: Navarro
  - https://learningstatisticswithr.com/boo





More Resources

- **R-Ladies EL material:** github.com/rladies-eastlansing
- **#TidyTuesday** challenges
- **gganimate**: thomasp85/gganimate
- **tidyexplain**: gadenbuie/tidyexplain
- **Pipes (%>%)**: datacamp.com/community/tutorials/pipe-r-tutorial
- **Distill (theme for RMarkdown)**: https://rstudio.github.io/distill
- Google & https://stackoverflow.com/ are your best friends!

Acknowledgements

- Arjun Krishnan, CMSE & BMB, MSU
- R-Ladies EL & my previous talks!
- JRaviLab & the Krishnan Lab
- The R&DS books
- The R-Ladies Global community

KrishnanLab

East Lansing

Questions? Comments?

East Lansing

janani@msu.edu

jravilab.github.io

github.com/jananiravi

twitter.com/janani137

eastlansing@rladies.org

rladies-eastlansing.github.io

github.com/rladies-eastlansing

twitter.com/RLadiesELansing

JOIN US!

JRaviLab, MSU