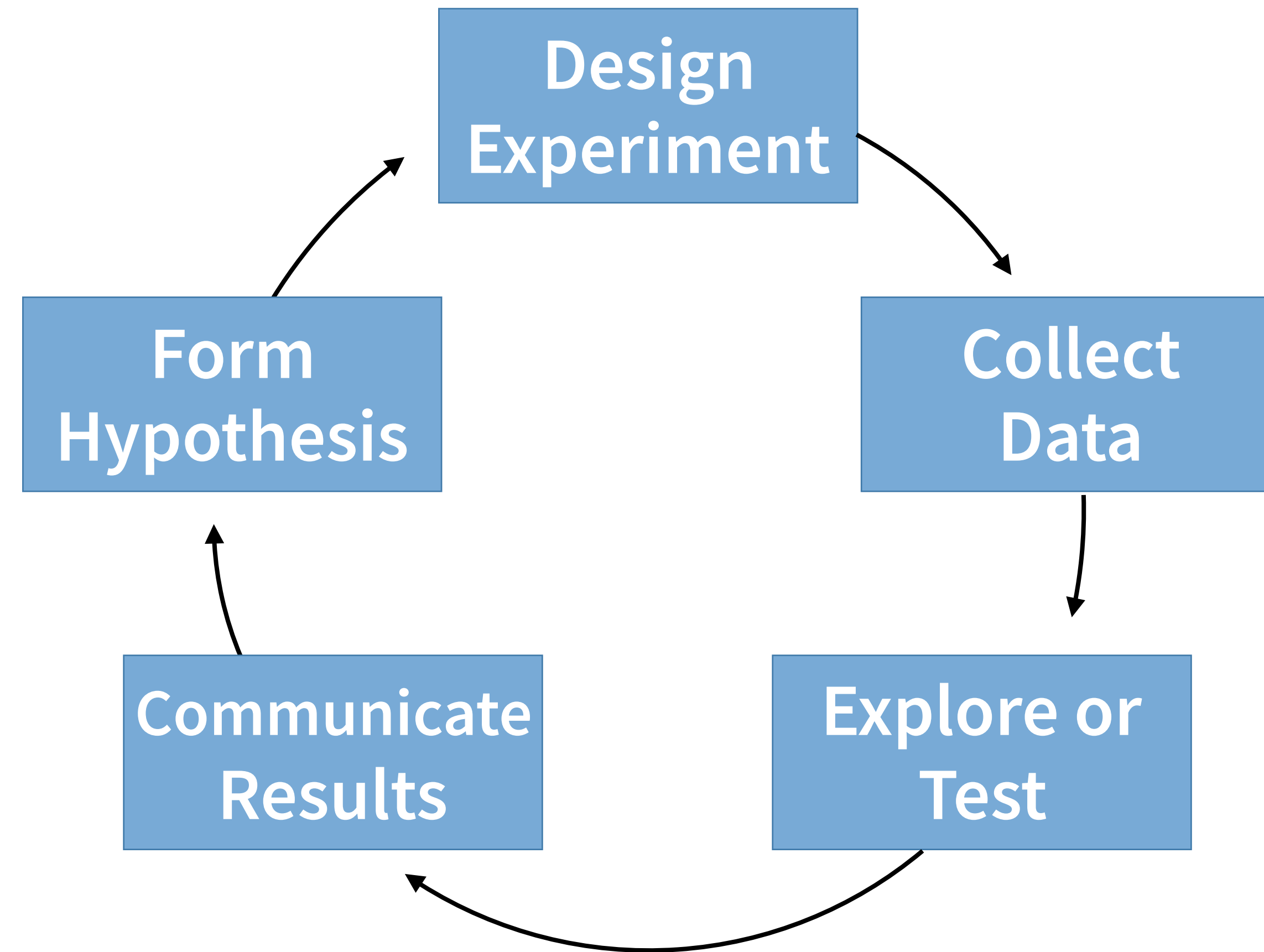
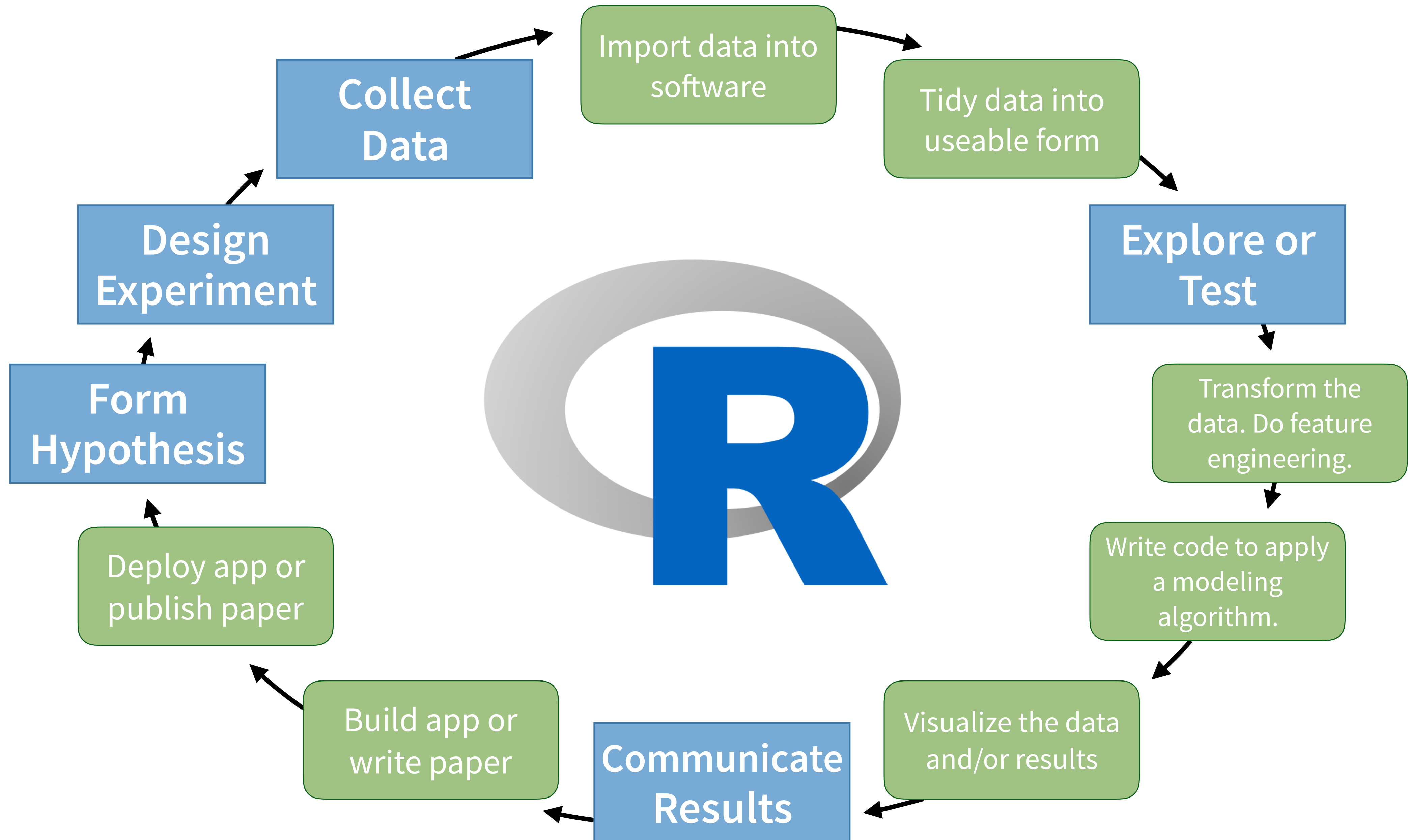


# Why learn the Tidyverse?

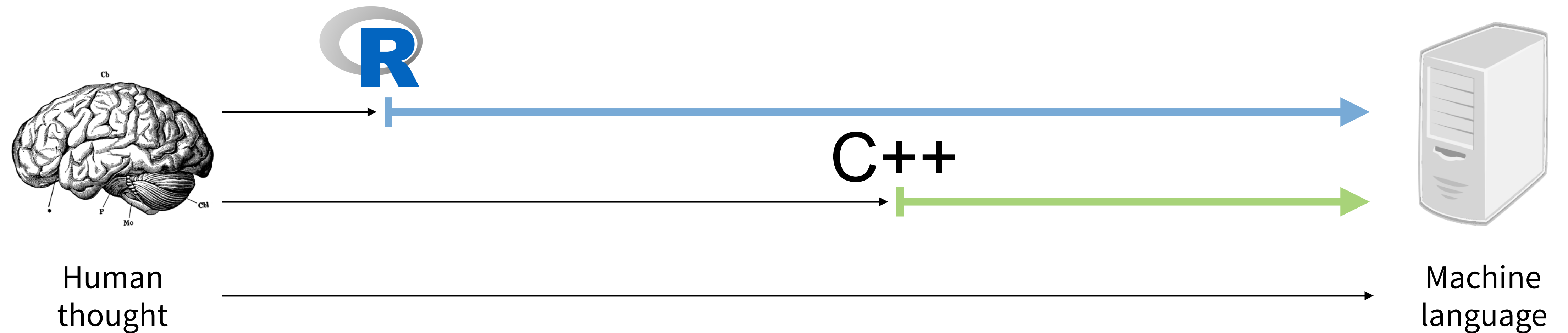


# "Data Science"



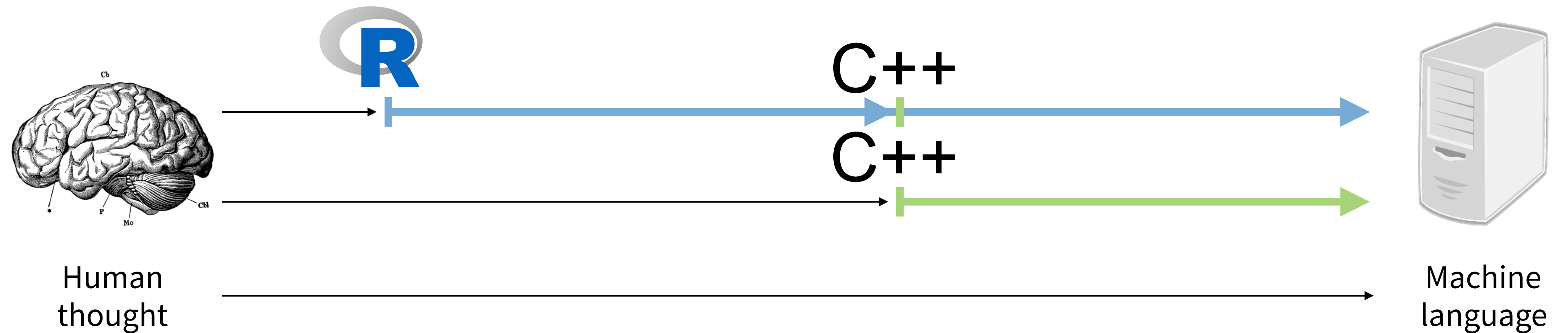


# R - A computer language for scientists

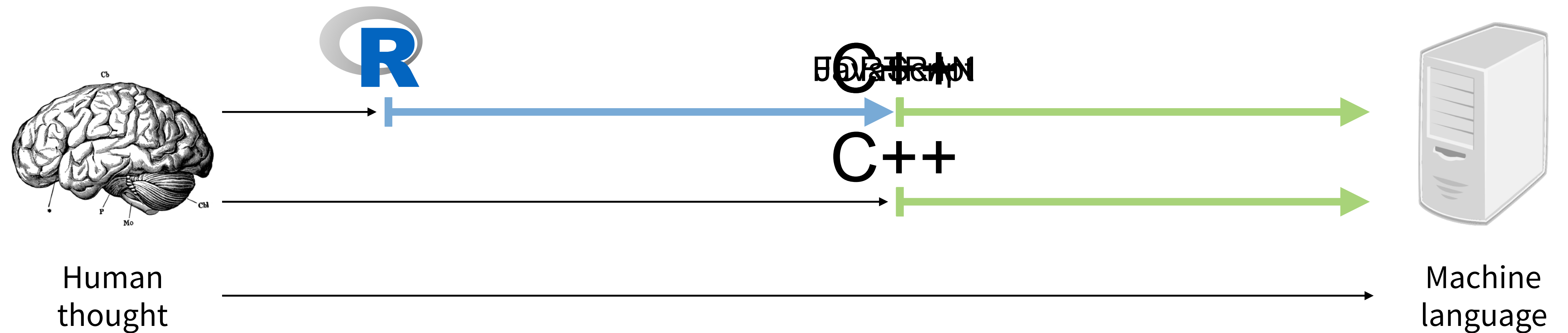




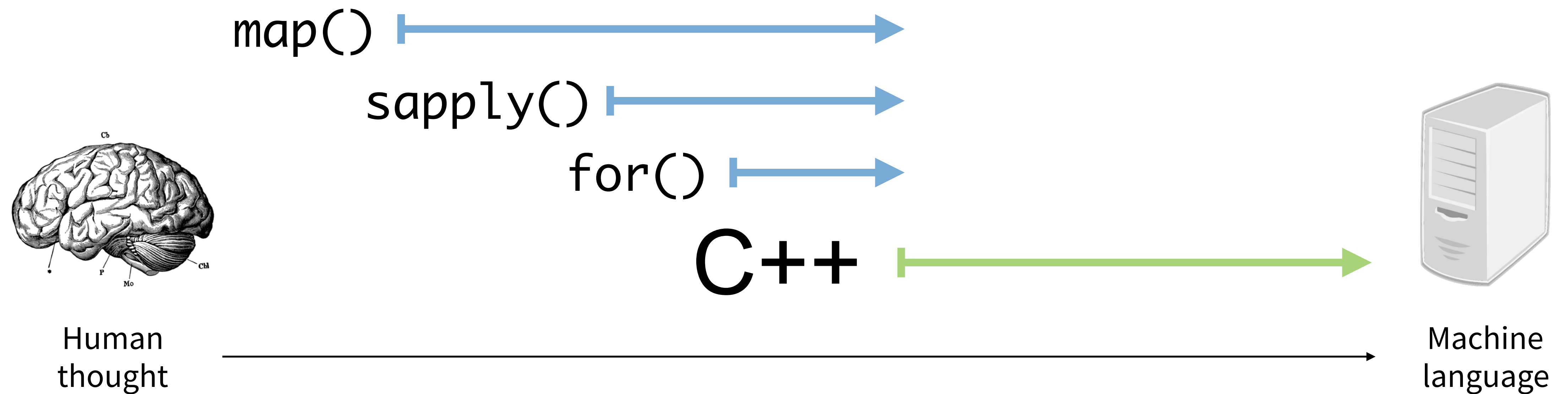
# R - A computer language for scientists



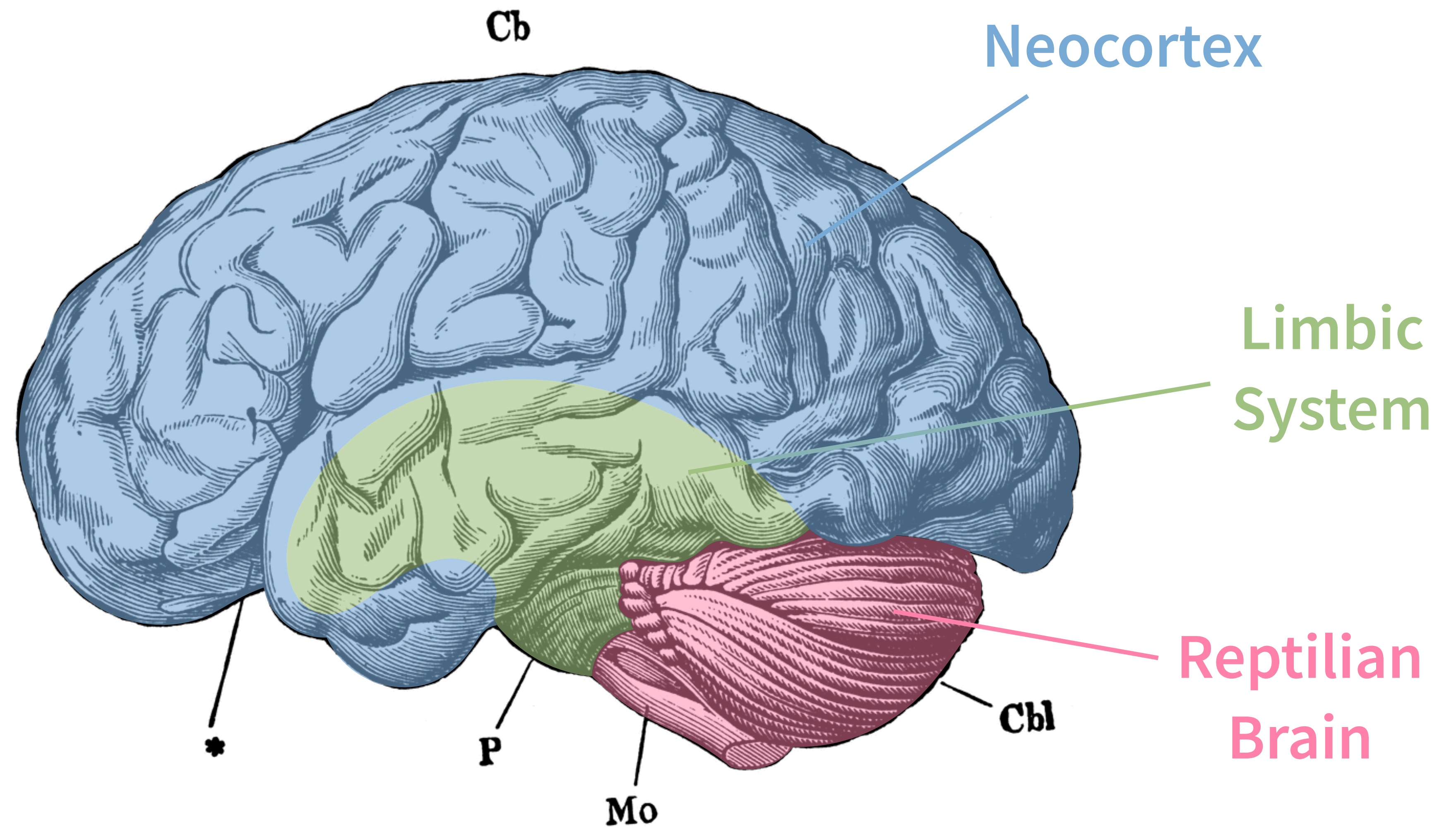
# **R** - A computer language for scientists



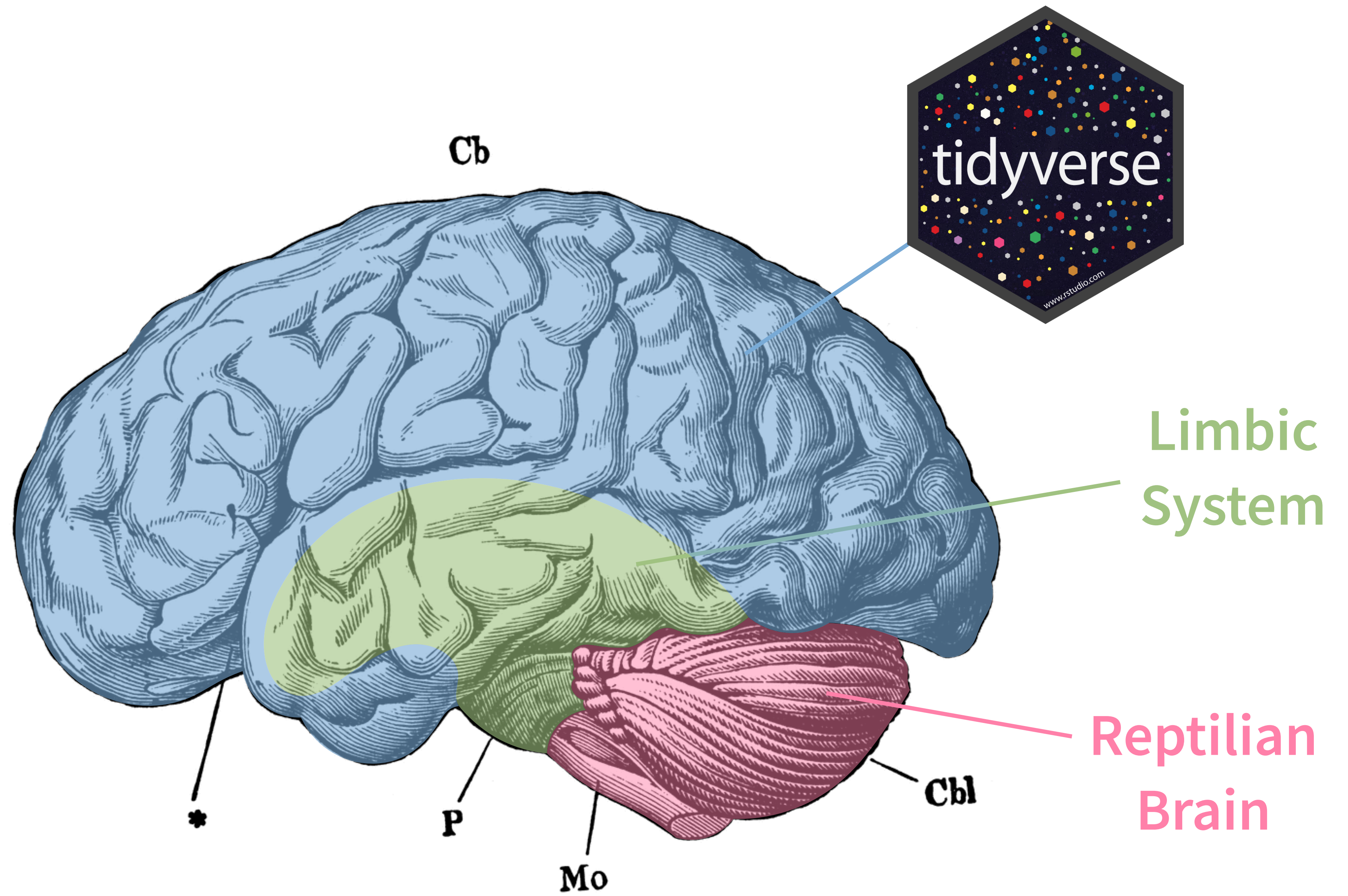
# R - A computer language for scientists











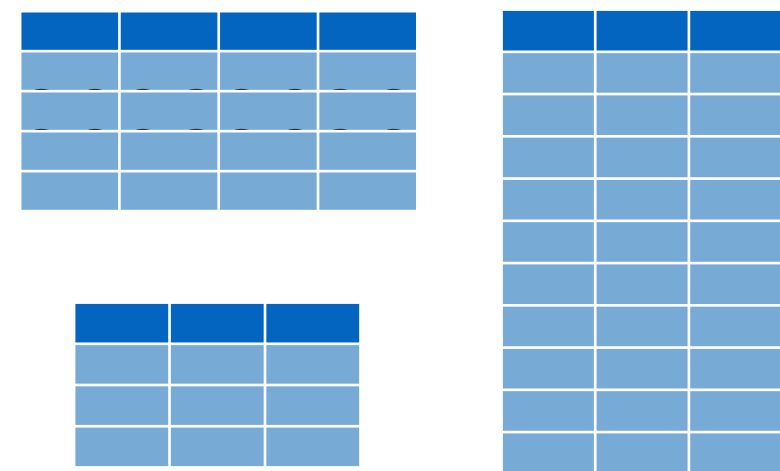


# Tidyverse

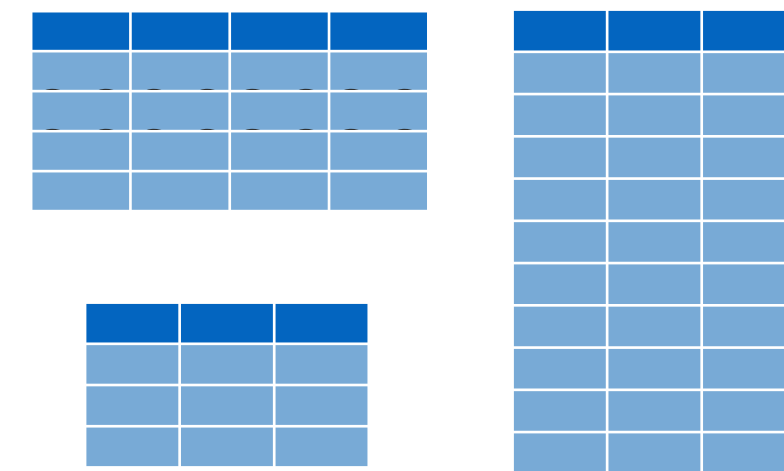


# R Packages

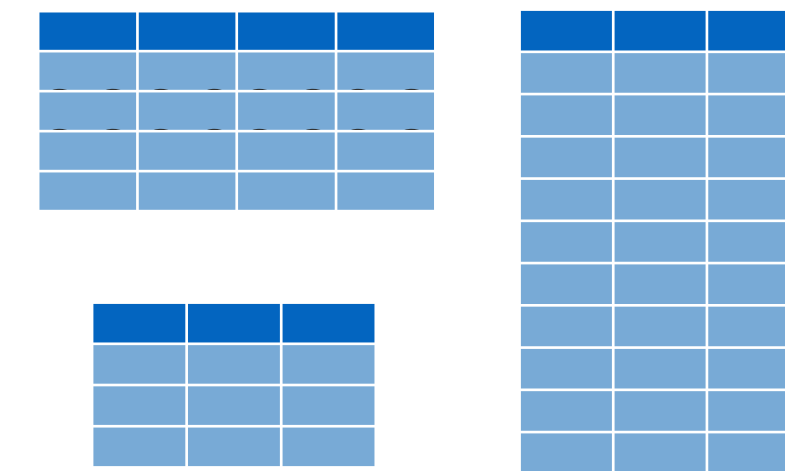




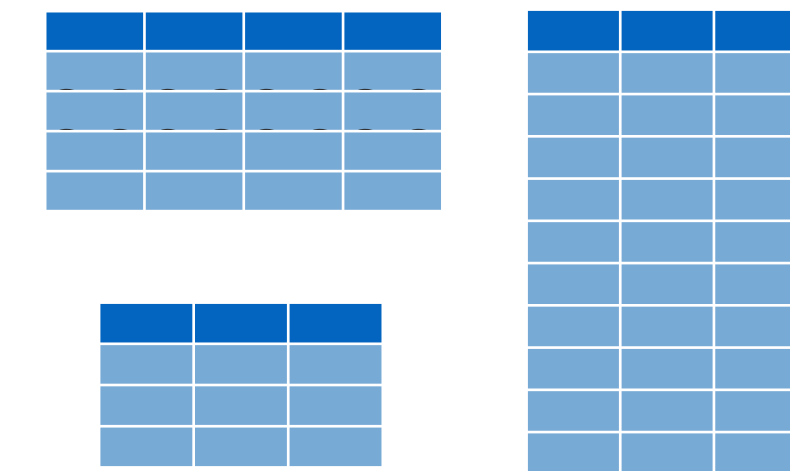
function1()  
function2()  
function3()  
function4()



function5()  
function6()  
function7()  
function8()



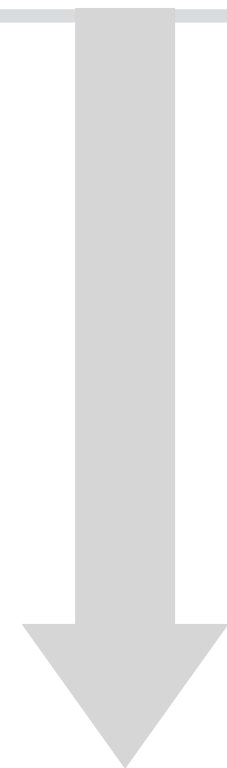
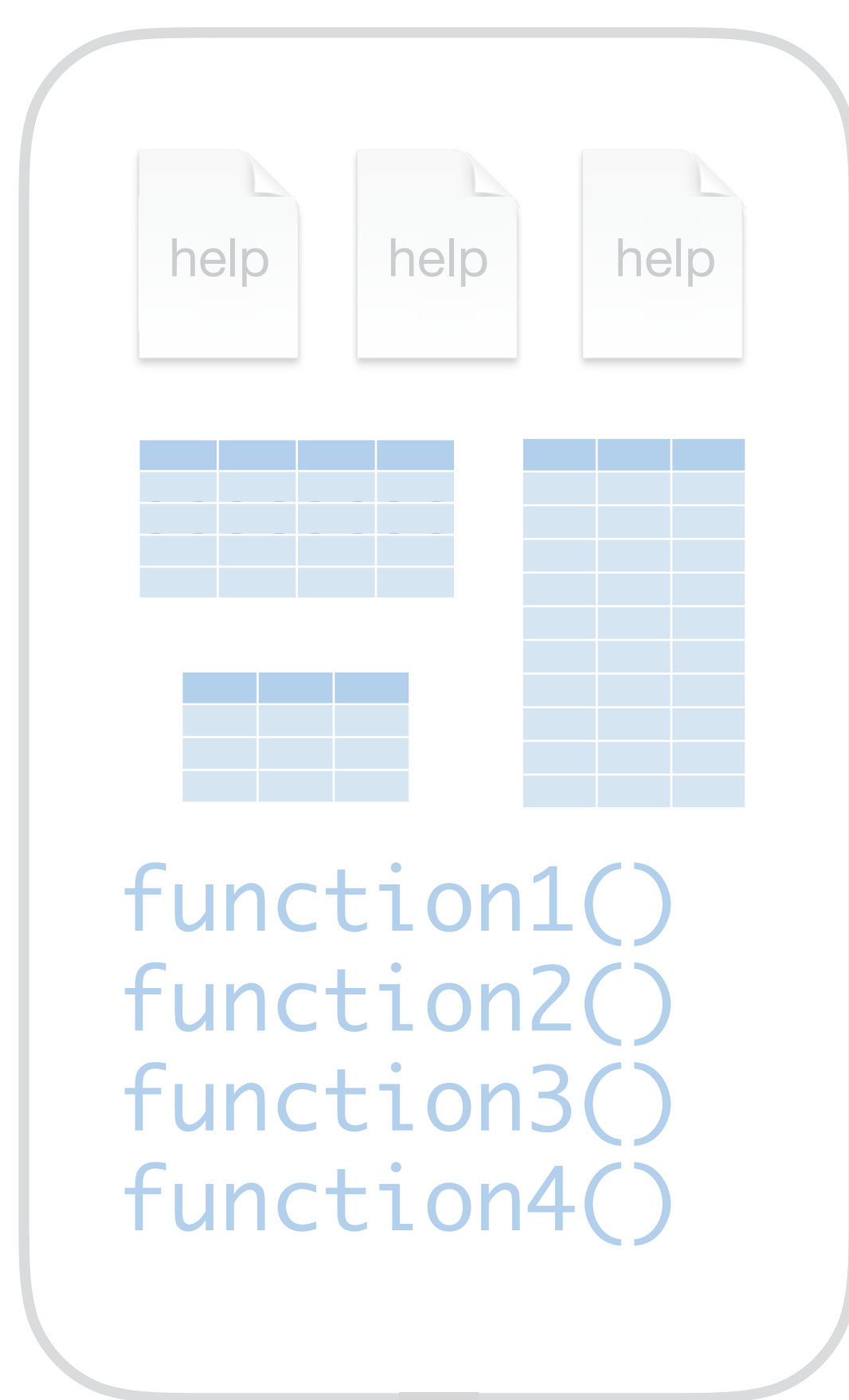
function9()  
functionA()  
functionB()  
functionC()



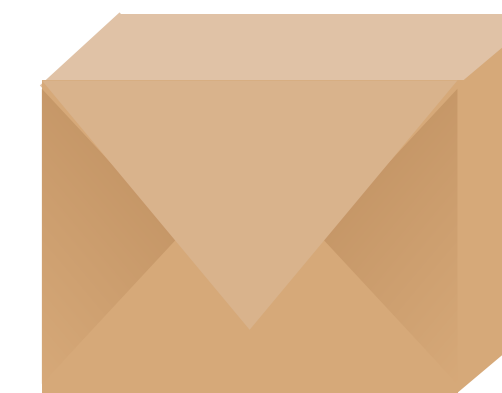
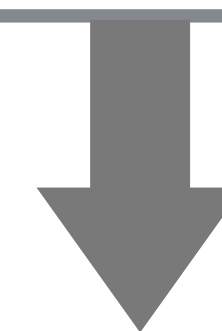
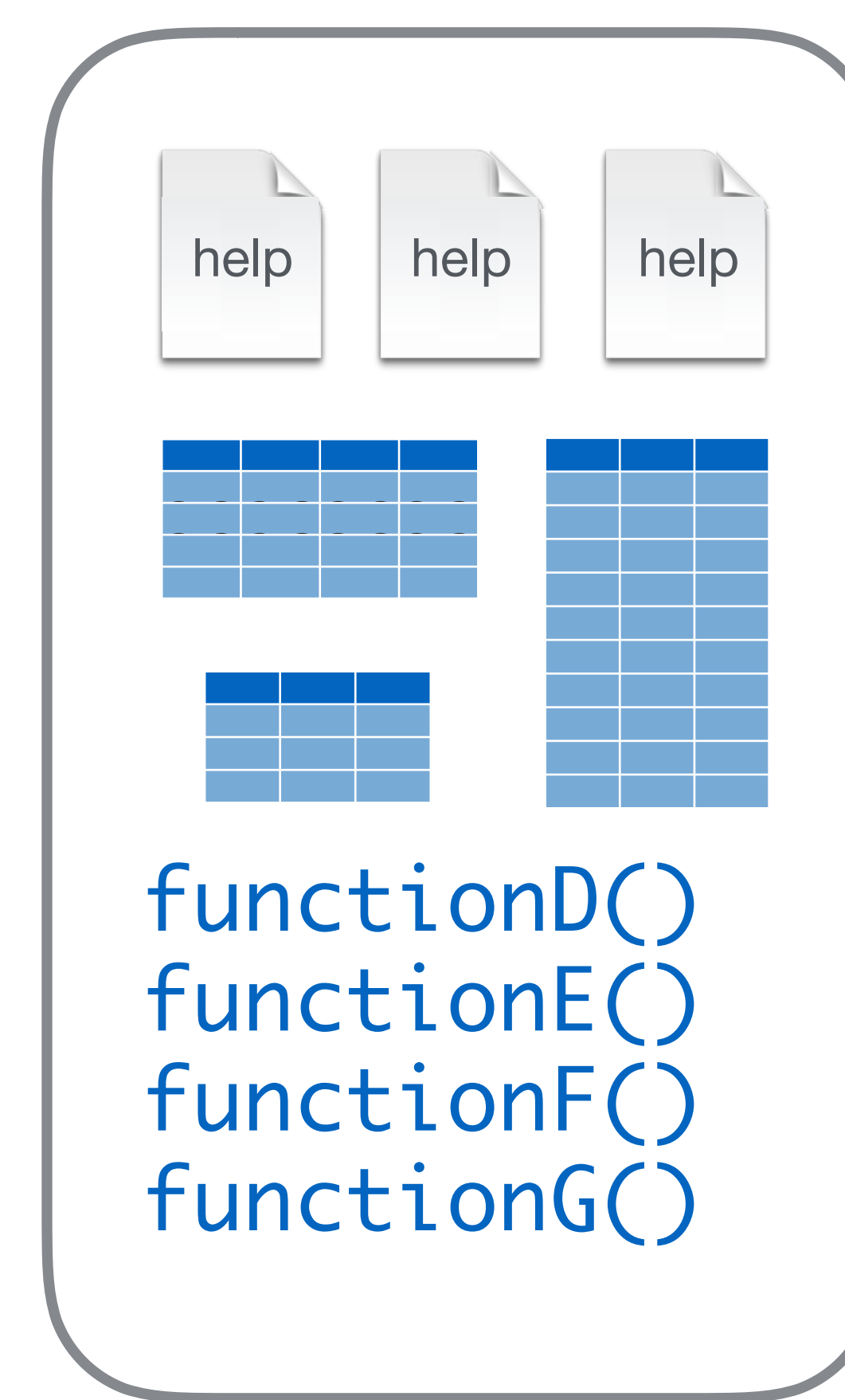
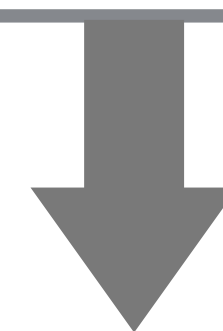
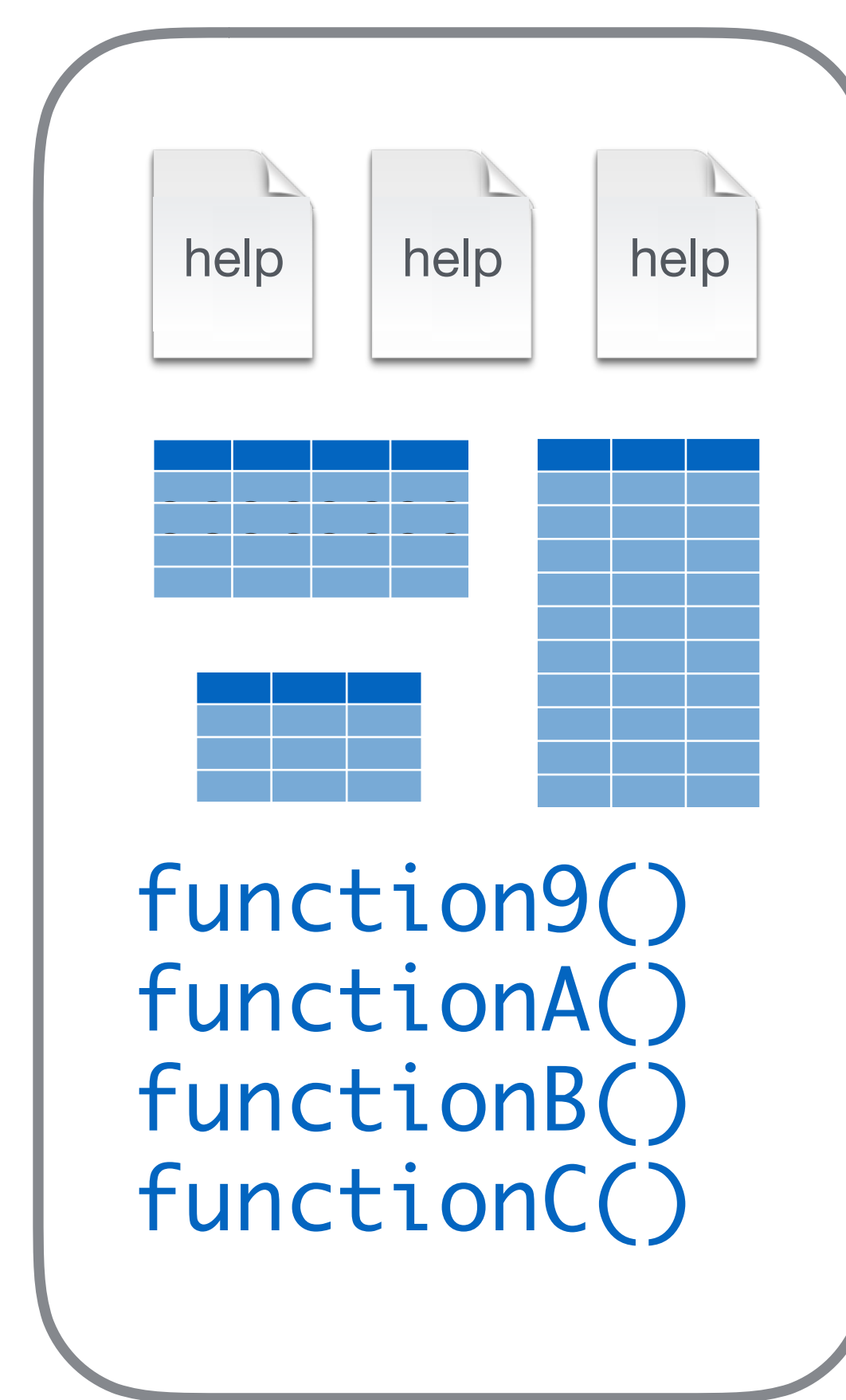
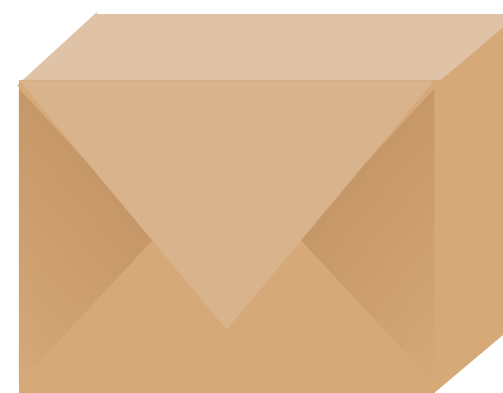
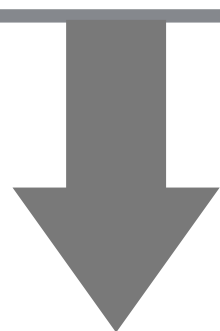
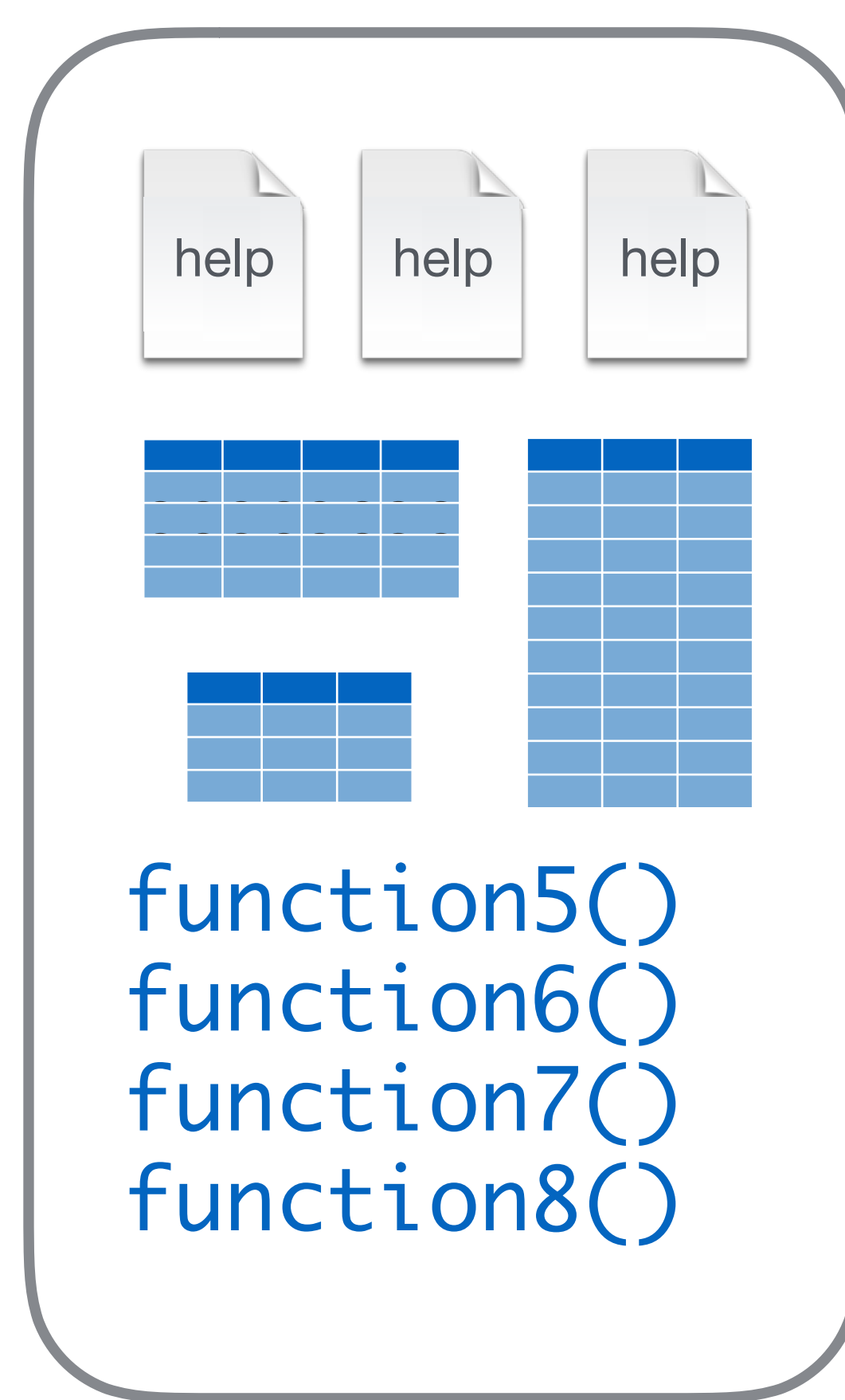
functionD()  
functionE()  
functionF()  
functionG()







Base R



R Packages

# Using packages

**1**

```
install.packages("foo")
```

Downloads files to computer

**1 x per computer**

# Using packages

**1**

```
install.packages("foo")
```

Downloads files to computer

**1 x per computer**

**2**

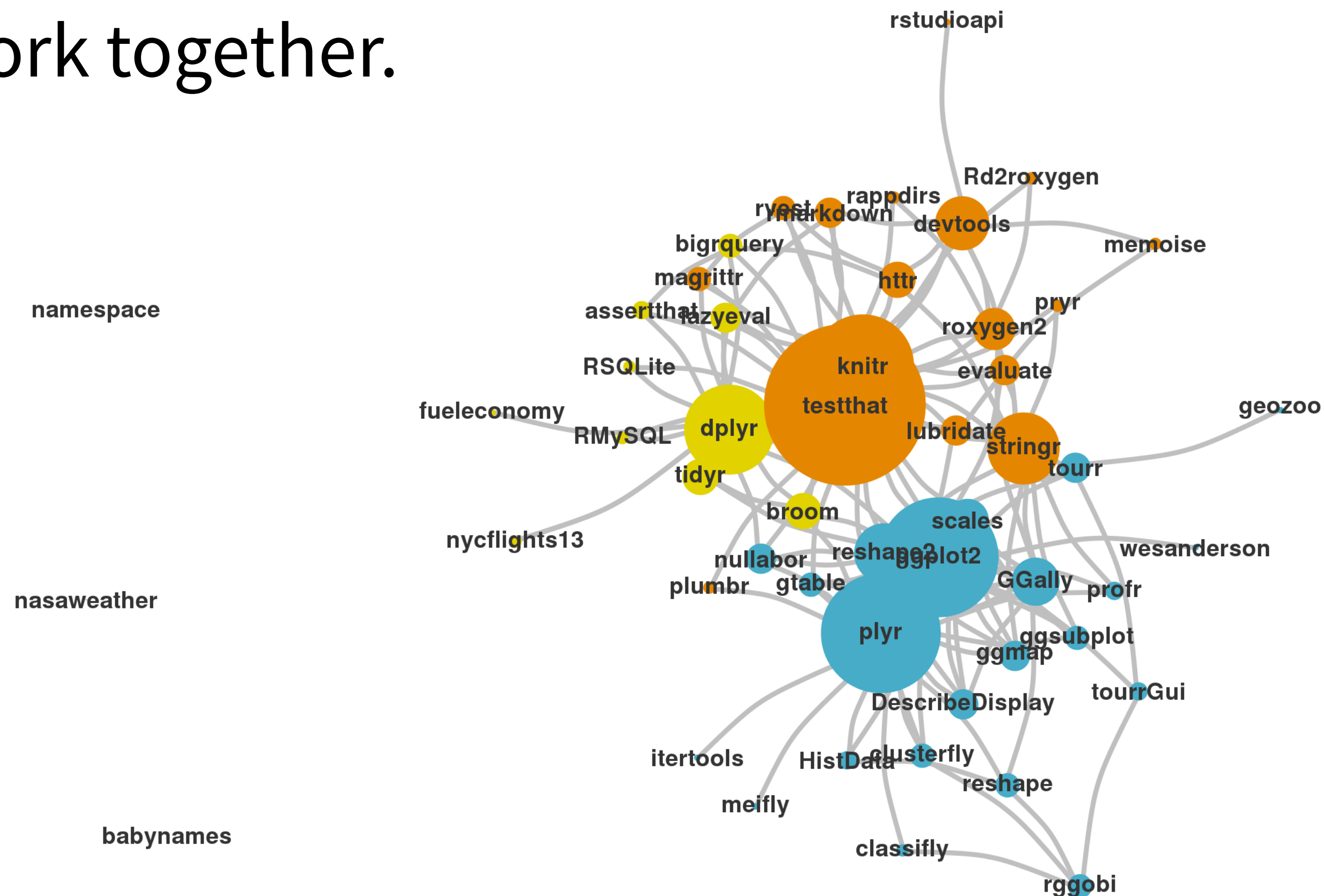
```
library("foo")
```

Loads package

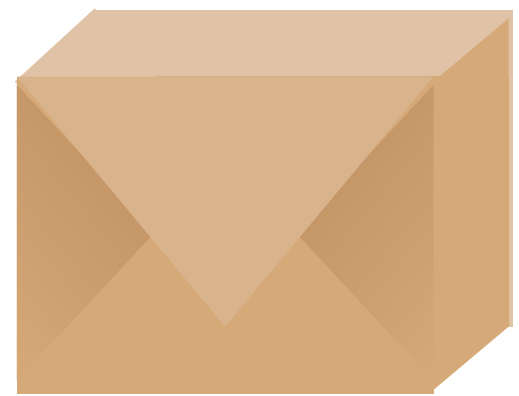
**1 x per R Session**

# The Tidyverse

A collection of modern R packages that share common philosophies, embed best practices, and are designed to work together.



# tidyverse



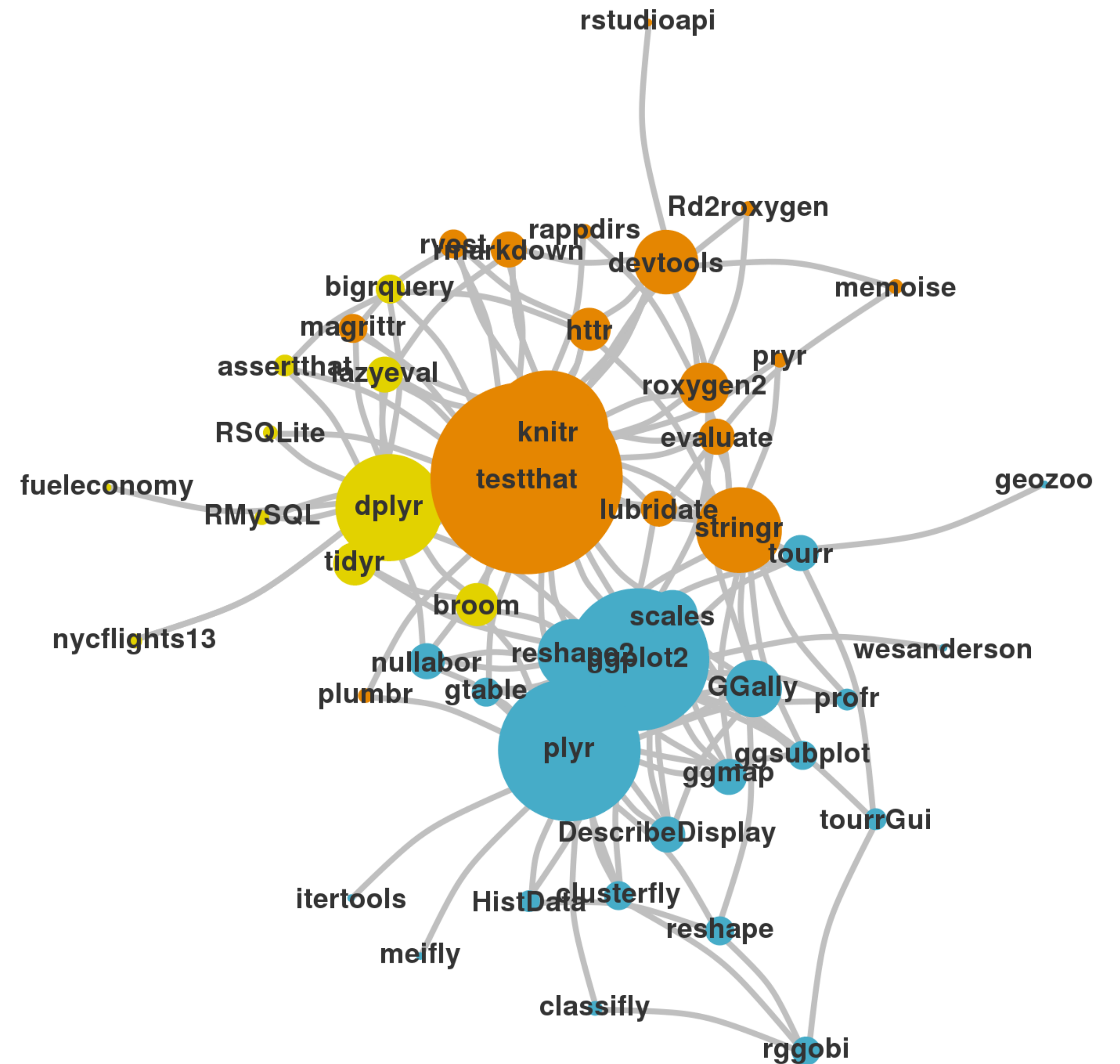
An R package that serves as a short cut for installing and loading the components of the tidyverse.

```
library("tidyverse")
```

```
install.packages("tidyverse")
```

does the equivalent of

```
install.packages("ggplot2")
install.packages("dplyr")
install.packages("tidyr")
install.packages("readr")
install.packages("purrr")
install.packages("tibble")
install.packages("hms")
install.packages("stringr")
install.packages("lubridate")
install.packages("forcats")
install.packages("DBI")
install.packages("haven")
install.packages("httr")
install.packages("jsonlite")
install.packages("readxl")
install.packages("rvest")
install.packages("xml2")
install.packages("modelr")
install.packages("broom")
```





```
install.packages("tidyverse")
```

does the equivalent of

```
install.packages("ggplot2")
install.packages("dplyr")
install.packages("tidyr")
install.packages("readr")
install.packages("purrr")
install.packages("tibble")
install.packages("hms")
install.packages("stringr")
install.packages("lubridate")
install.packages("forcats")
install.packages("DBI")
install.packages("haven")
install.packages("httr")
install.packages("jsonlite")
install.packages("readxl")
install.packages("rvest")
install.packages("xml2")
install.packages("modelr")
install.packages("broom")
```

```
library("tidyverse")
```

does the equivalent of

```
library("ggplot2")
library("dplyr")
library("tidyr")
library("readr")
library("purrr")
library("tibble")
```



```
install.packages("tidyverse")
```

does the equivalent of

```
install.packages("ggplot2")
install.packages("dplyr")
install.packages("tidyr")
install.packages("readr")
install.packages("purrr")
install.packages("tibble")
install.packages("hms")
install.packages("stringr")
install.packages("lubridate")
install.packages("forcats")
install.packages("DBI")
install.packages("haven")
install.packages("httr")
install.packages("jsonlite")
install.packages("readxl")
install.packages("rvest")
install.packages("xml2")
install.packages("modelr")
install.packages("broom")
```

```
library("tidyverse")
```

does the equivalent of

```
library("ggplot2")
library("dplyr")
library("tidyr")
library("readr")
library("purrr")
library("tibble")
```

**Visualization tools**

**Six functions**

- arrange()
- filter()
- select()
- mutate()
- summarise()
- group\_by()

# Tidy tools



# Tidy tools

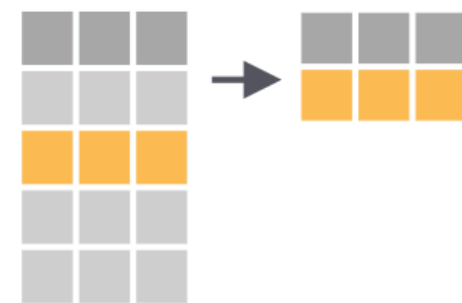
Functions are easiest to use when they are:

1. **Simple** - They do one thing, and they do it well
2. **Composable** - They can be combined with other functions for multi-step operations

# 1. Simple - They do one thing, and they do it well



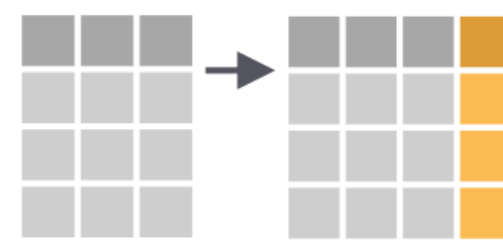
**arrange(.data, ...)**  
Order rows by values of a column (low to high), use with **desc()** to order from high to low.



**filter(.data, ...)**  
Extract rows that meet logical criteria.



**select(.data, ...)**  
Extract columns by name.



**mutate(.data, ...)**  
Compute new column(s).



**summarise(.data, ...)**  
Compute table of summaries. Use **group\_by()** to compute groupwise summaries.

**2. Composable** - They can be combined with other functions for multi-step operations

**%>%**



# pipes

**`x %>% f(y)`**  
becomes **`f(x, y)`**



`gapminder`

`arrange(_____, desc(pop))`

# Shortcut to type %>%

**Cmd** + **Shift** + **M** (Mac)

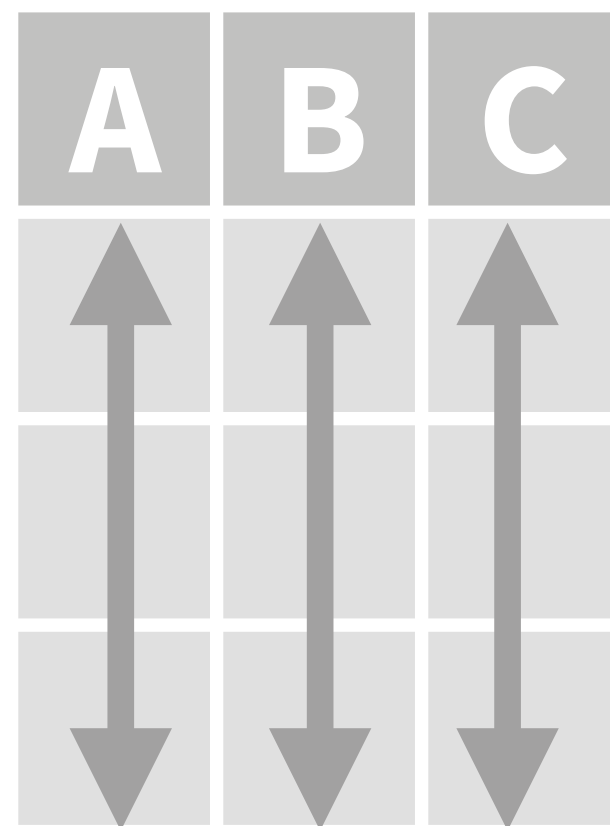
**Ctrl** + **Shift** + **M** (Windows)

# Tidy Data



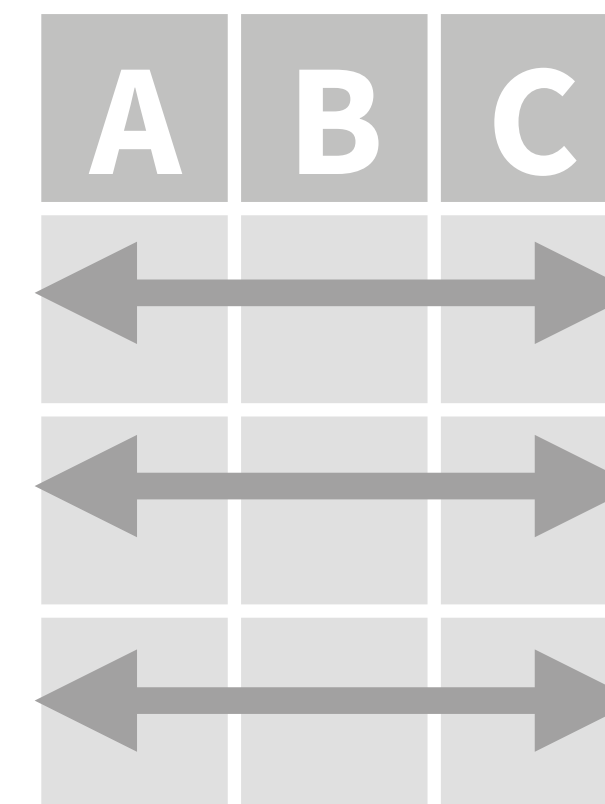


# Tidy data

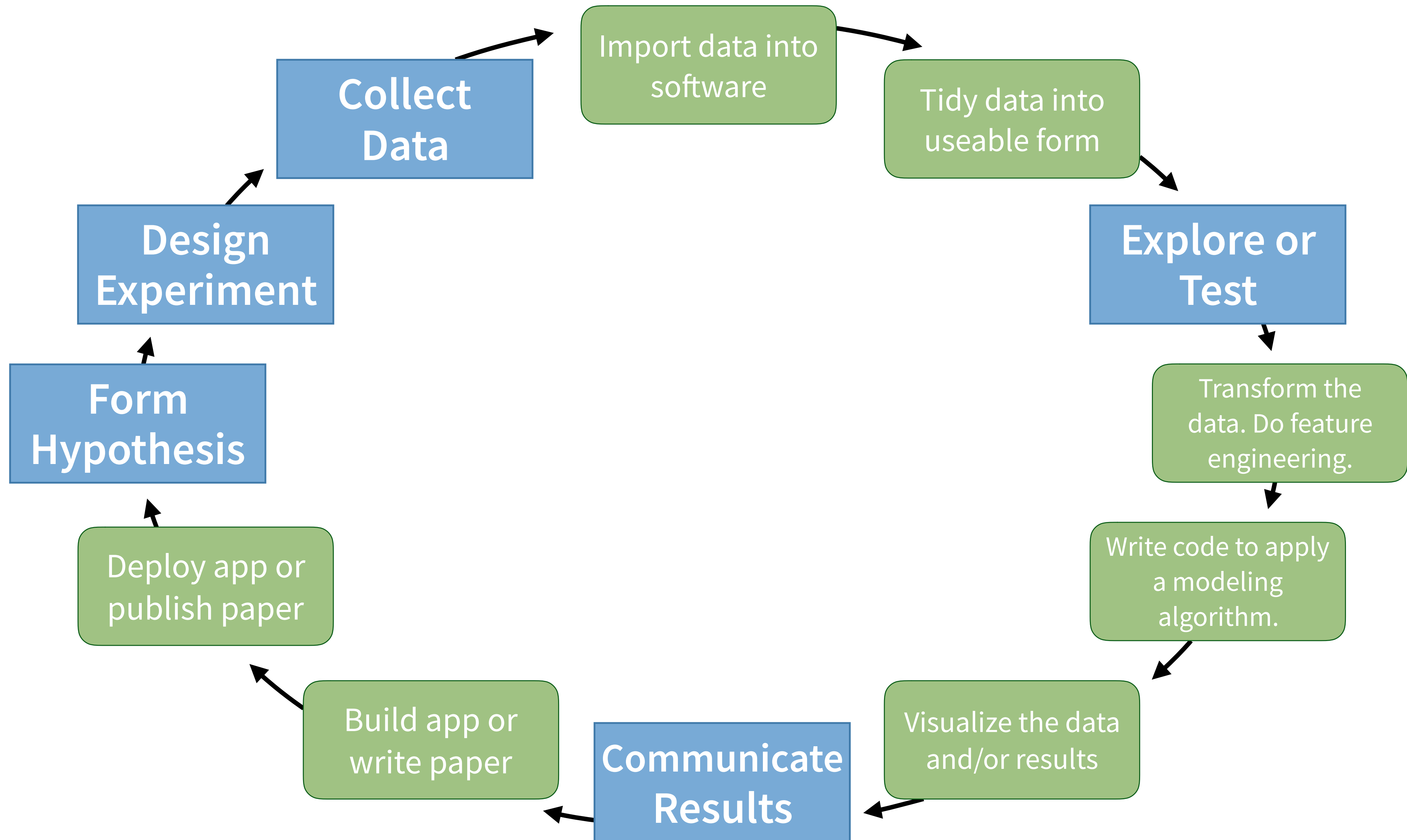


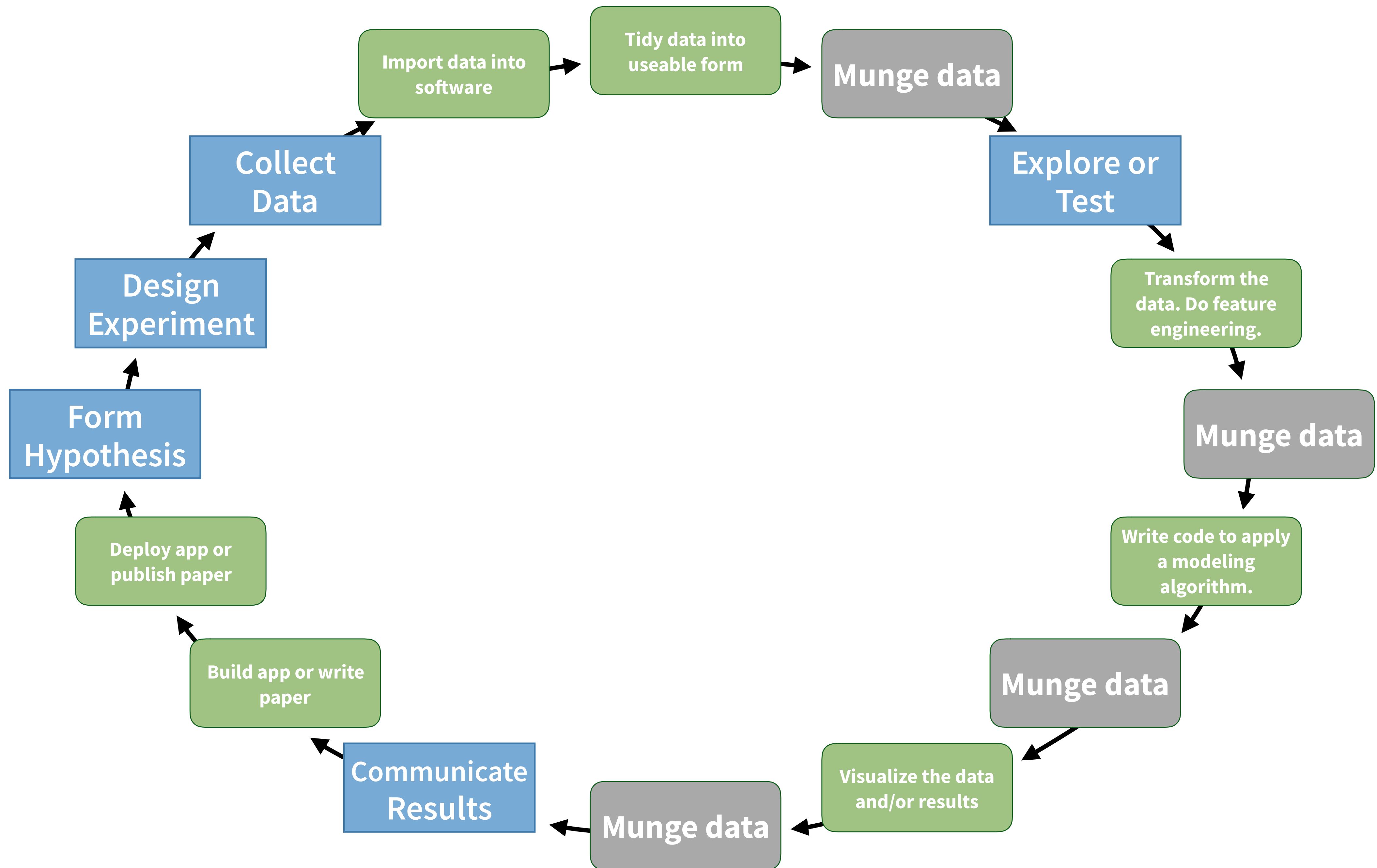
Each **variable** is in its own **column**

&



Each **observation**, or **case**, is in its own **row**





```
install.packages("tidyverse")
```

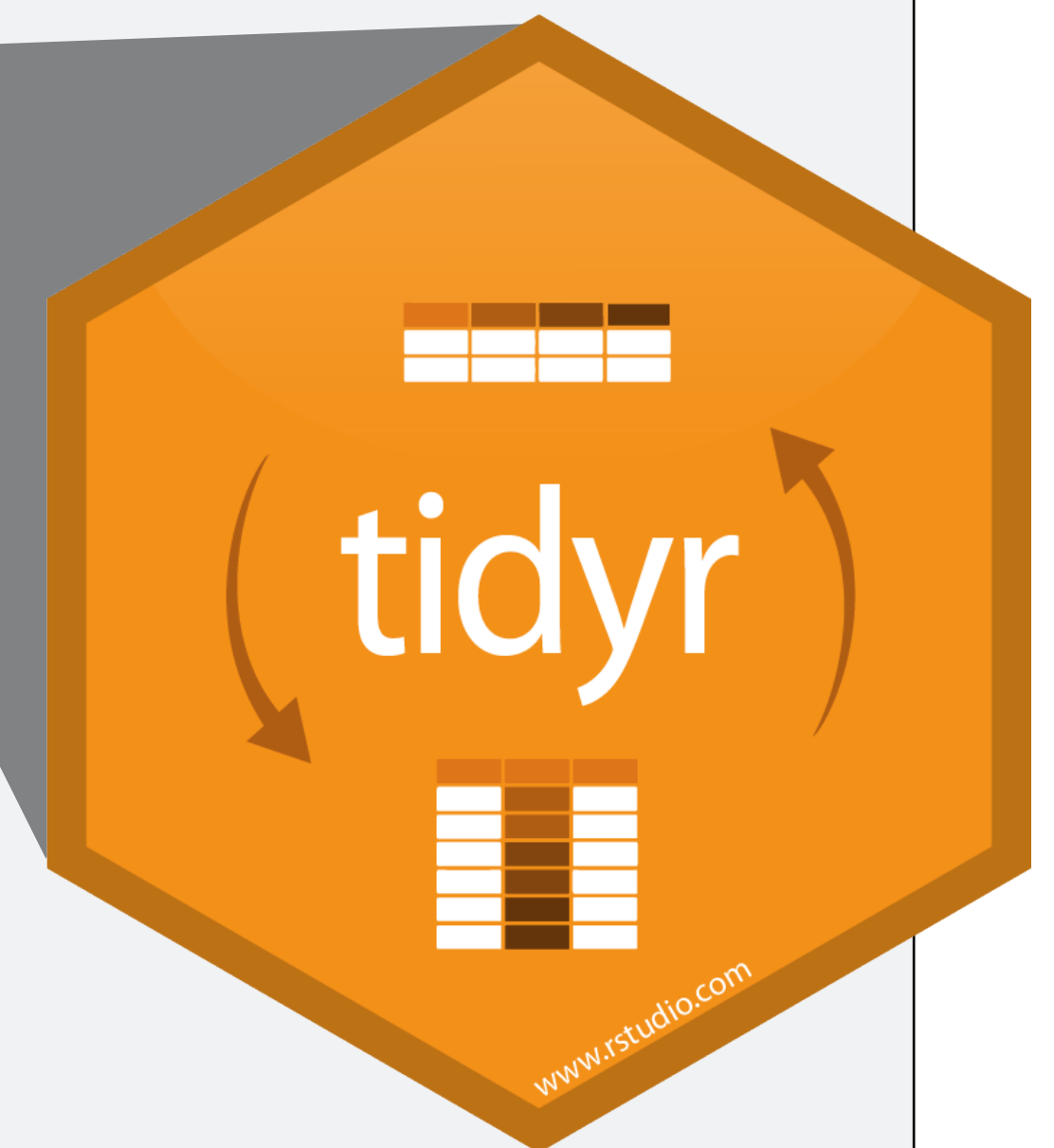
does the equivalent of

```
install.packages("ggplot2")  
install.packages("dplyr")  
install.packages("tidyr")  
install.packages("readr")  
install.packages("purrr")  
install.packages("tibble")  
install.packages("hms")  
install.packages("stringr")  
install.packages("lubridate")  
install.packages("forcats")  
install.packages("DBI")  
install.packages("haven")  
install.packages("httr")  
install.packages("jsonlite")  
install.packages("readxl")  
install.packages("rvest")  
install.packages("xml2")  
install.packages("modelr")  
install.packages("broom")
```

```
library("tidyverse")
```

does the equivalent of

```
library("ggplot2")  
library("dplyr")  
library("tidyr")  
library("readr")  
library("purrr")  
library("tibble")
```



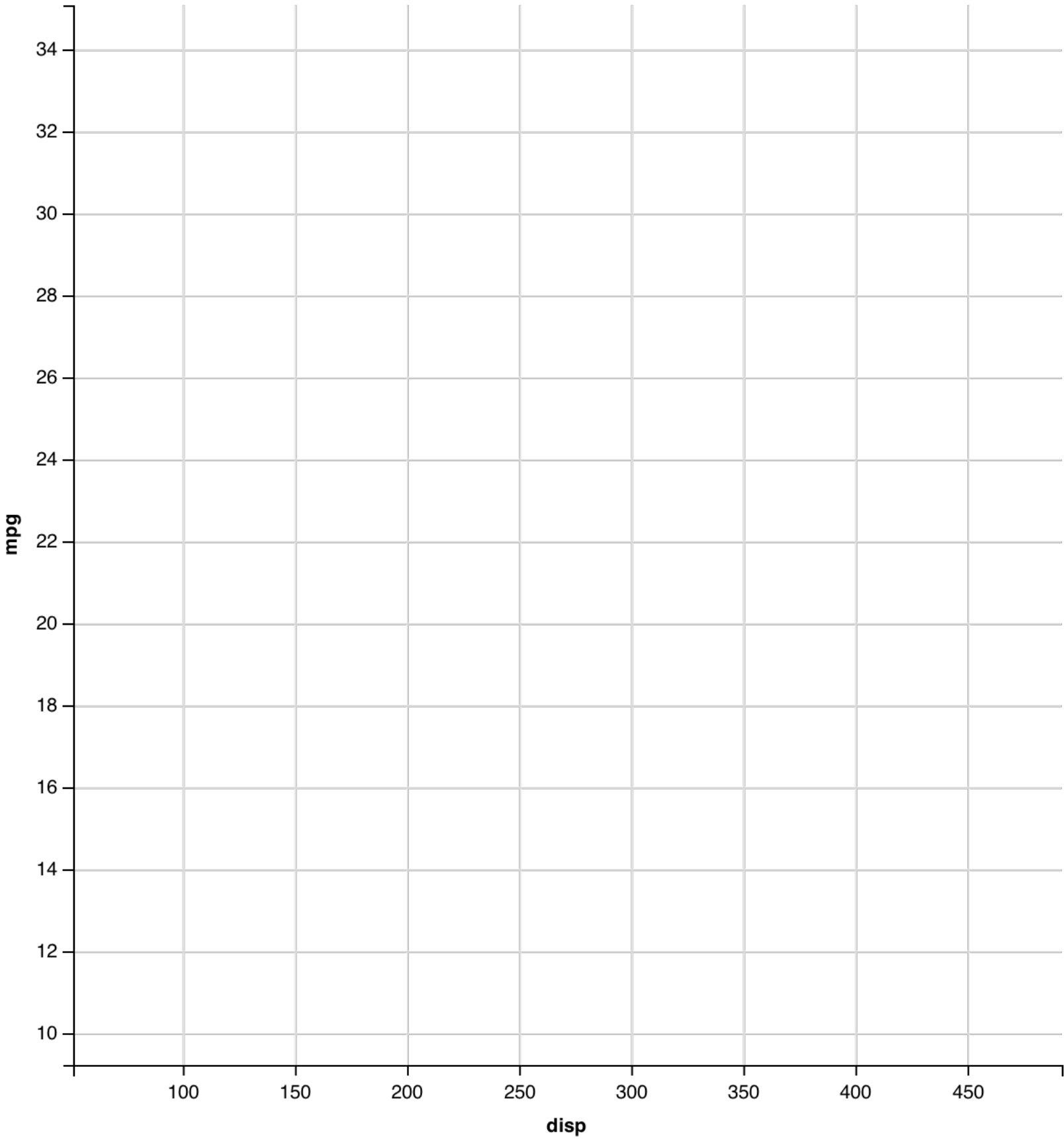
# Grammar of Graphics



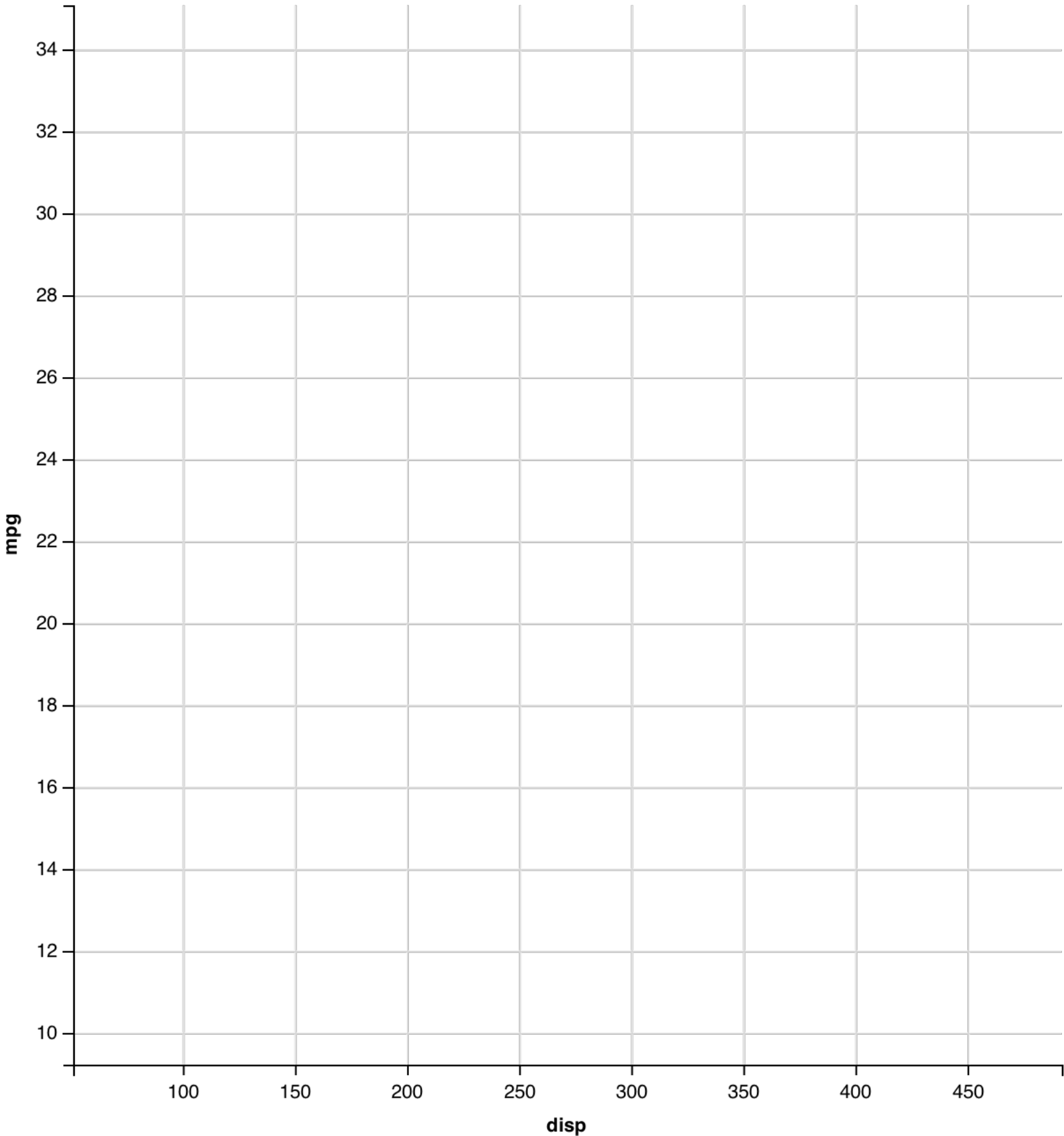
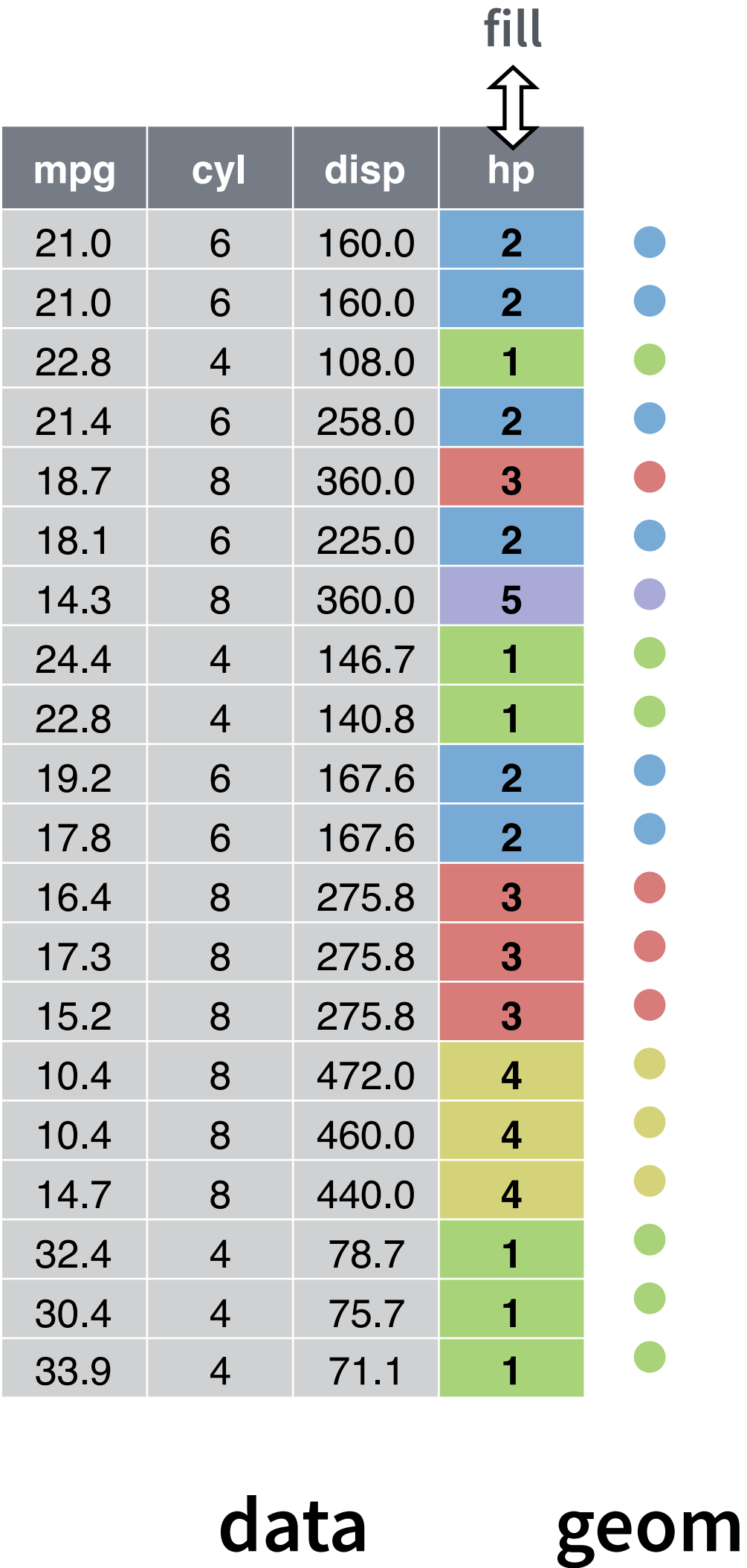
mpg	cyl	displacement	horsepower
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1

data

geom



mappings



mappings

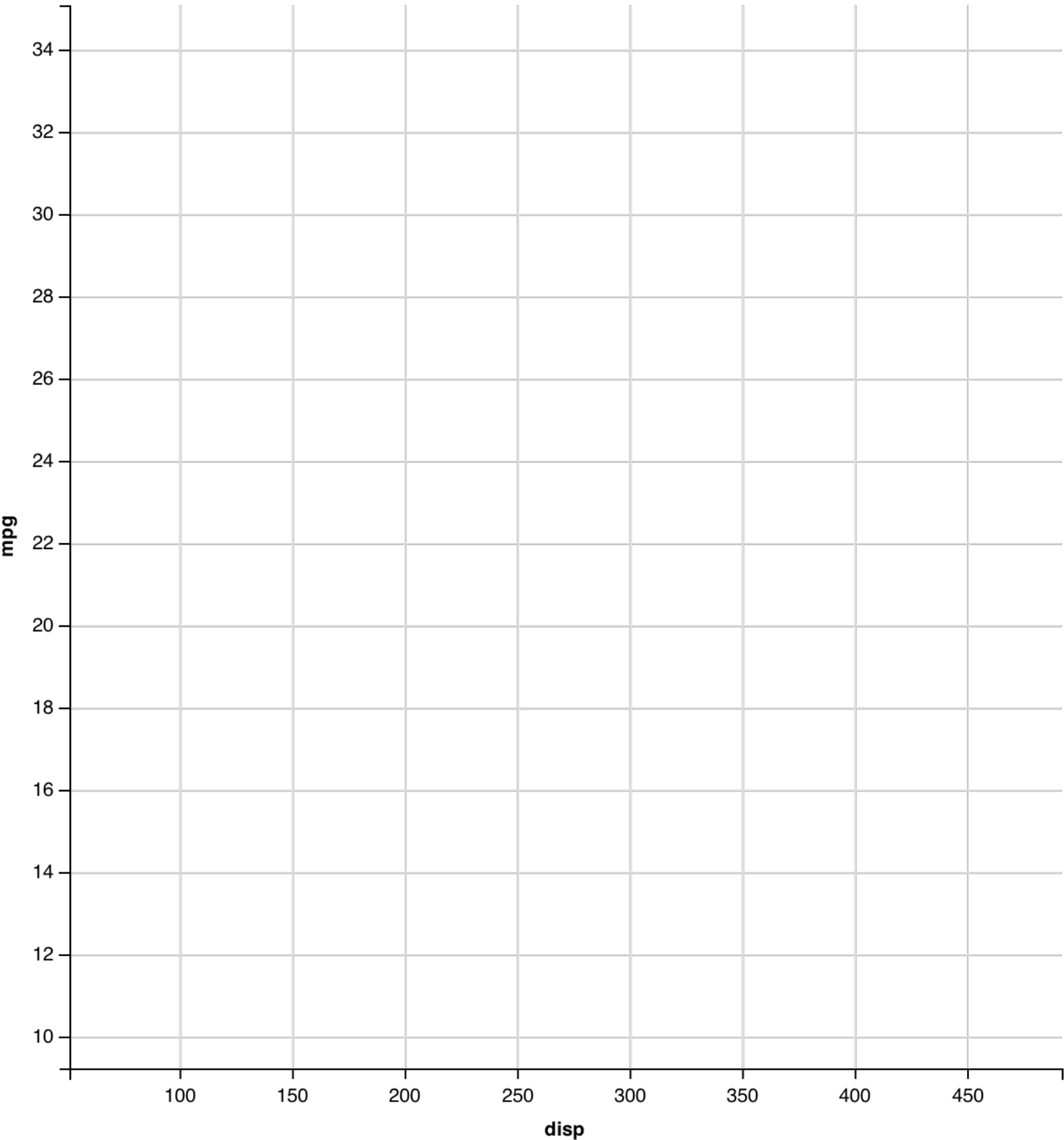
shape

fill

mpg	cyl	displacement	horsepower
21.0	6 +	160.0	2
21.0	6 +	160.0	2
22.8	4 ●	108.0	1
21.4	6 +	258.0	2
18.7	8 ◆	360.0	3
18.1	6 +	225.0	2
14.3	8 ◆	360.0	5
24.4	4 ●	146.7	1
22.8	4 ●	140.8	1
19.2	6 +	167.6	2
17.8	6 +	167.6	2
16.4	8 ◆	275.8	3
17.3	8 ◆	275.8	3
15.2	8 ◆	275.8	3
10.4	8 ◆	472.0	4
10.4	8 ◆	460.0	4
14.7	8 ◆	440.0	4
32.4	4 ●	78.7	1
30.4	4 ●	75.7	1
33.9	4 ●	71.1	1

data

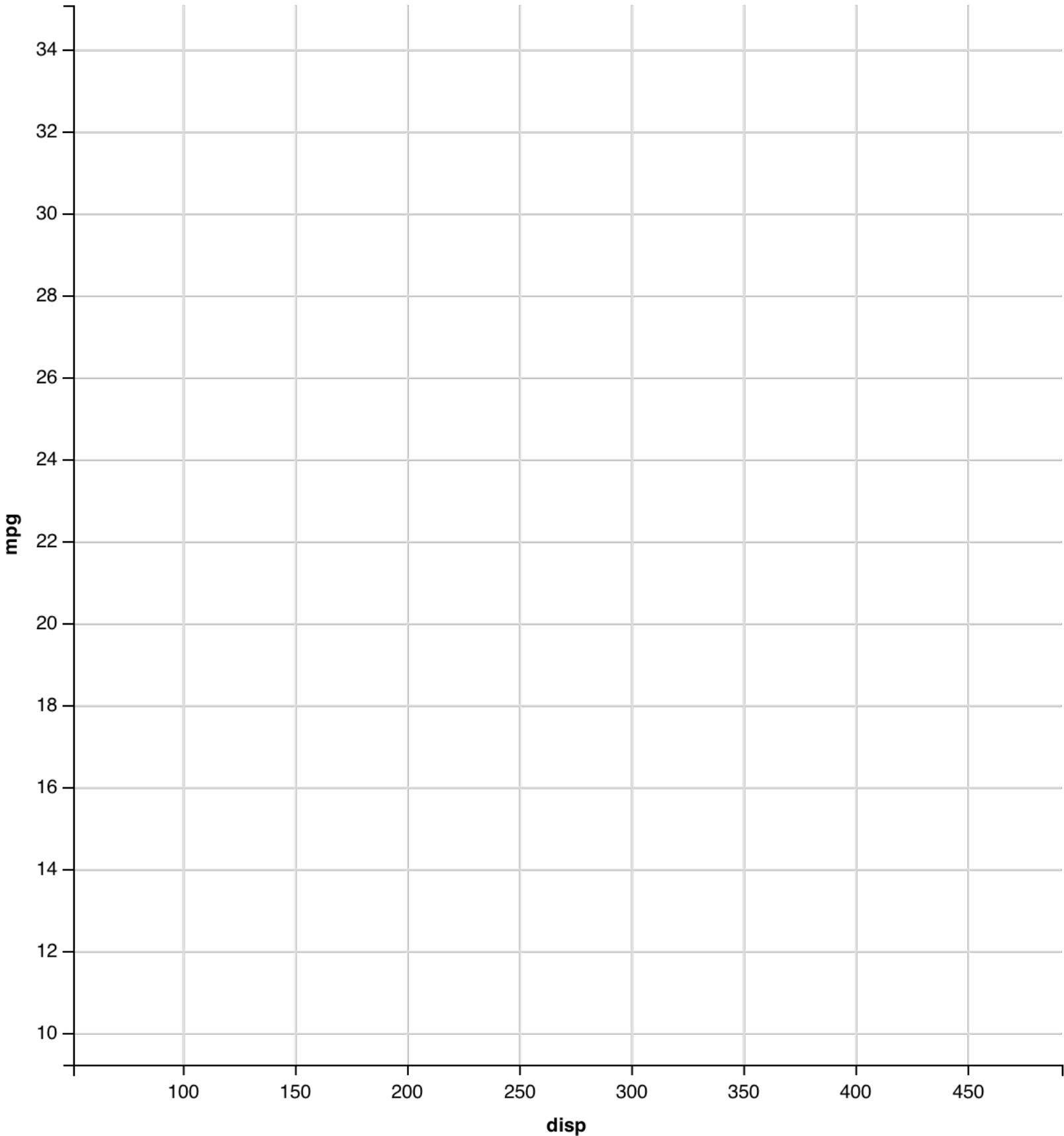
geom





mappings

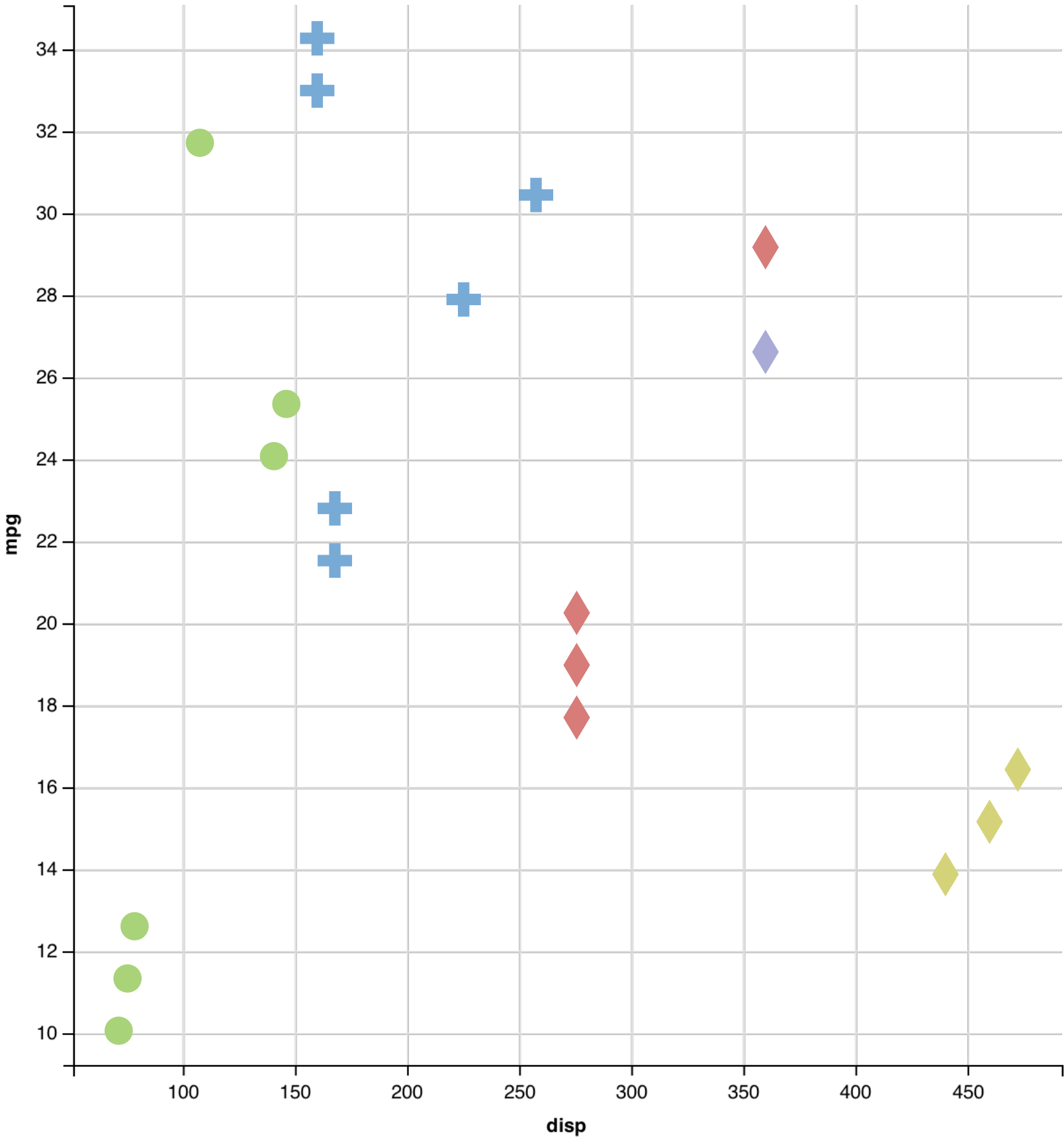
mpg	shape	x	fill	
	cyl	dispx	hp	
21.0	6	160.0	2	+
21.0	6	160.0	2	+
22.8	4	108.0	1	●
21.4	6	258.0	2	+
18.7	8	360.0	3	◆
18.1	6	225.0	2	+
14.3	8	360.0	5	◆
24.4	4	146.7	1	●
22.8	4	140.8	1	●
19.2	6	167.6	2	+
17.8	6	167.6	2	+
16.4	8	275.8	3	◆
17.3	8	275.8	3	◆
15.2	8	275.8	3	◆
10.4	8	472.0	4	◆
10.4	8	460.0	4	◆
14.7	8	440.0	4	◆
32.4	4	78.7	1	●
30.4	4	75.7	1	●
33.9	4	71.1	1	●



data geom

mappings

y	shape	x	fill
mpg	cyl	displacement	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1



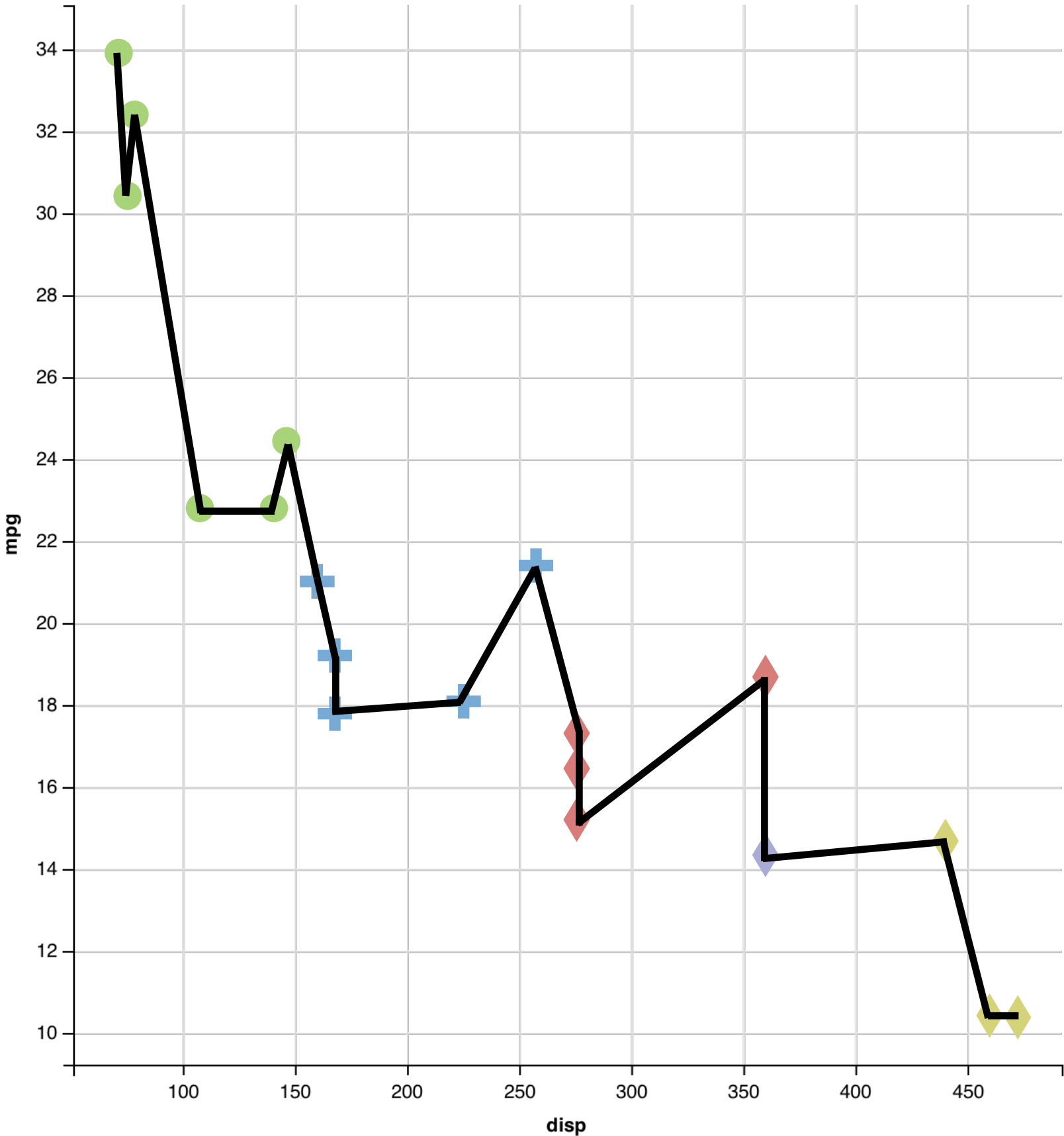
data geom

mappings

y	shape	x	fill
mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1

data

geom  
points  
lines



mappings

y

↑

mpg	cyl	dispx	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

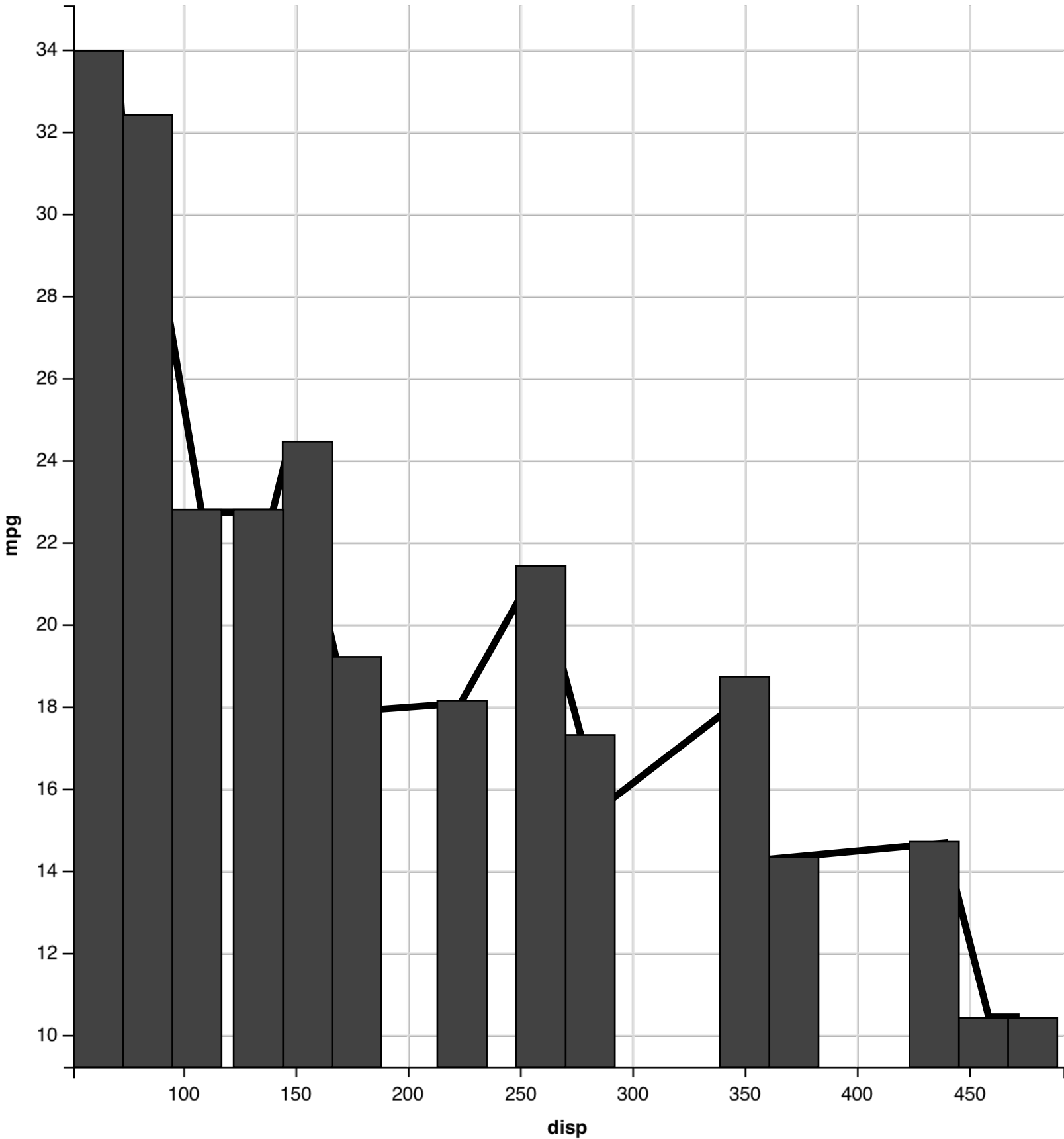
+

+

+

+

+

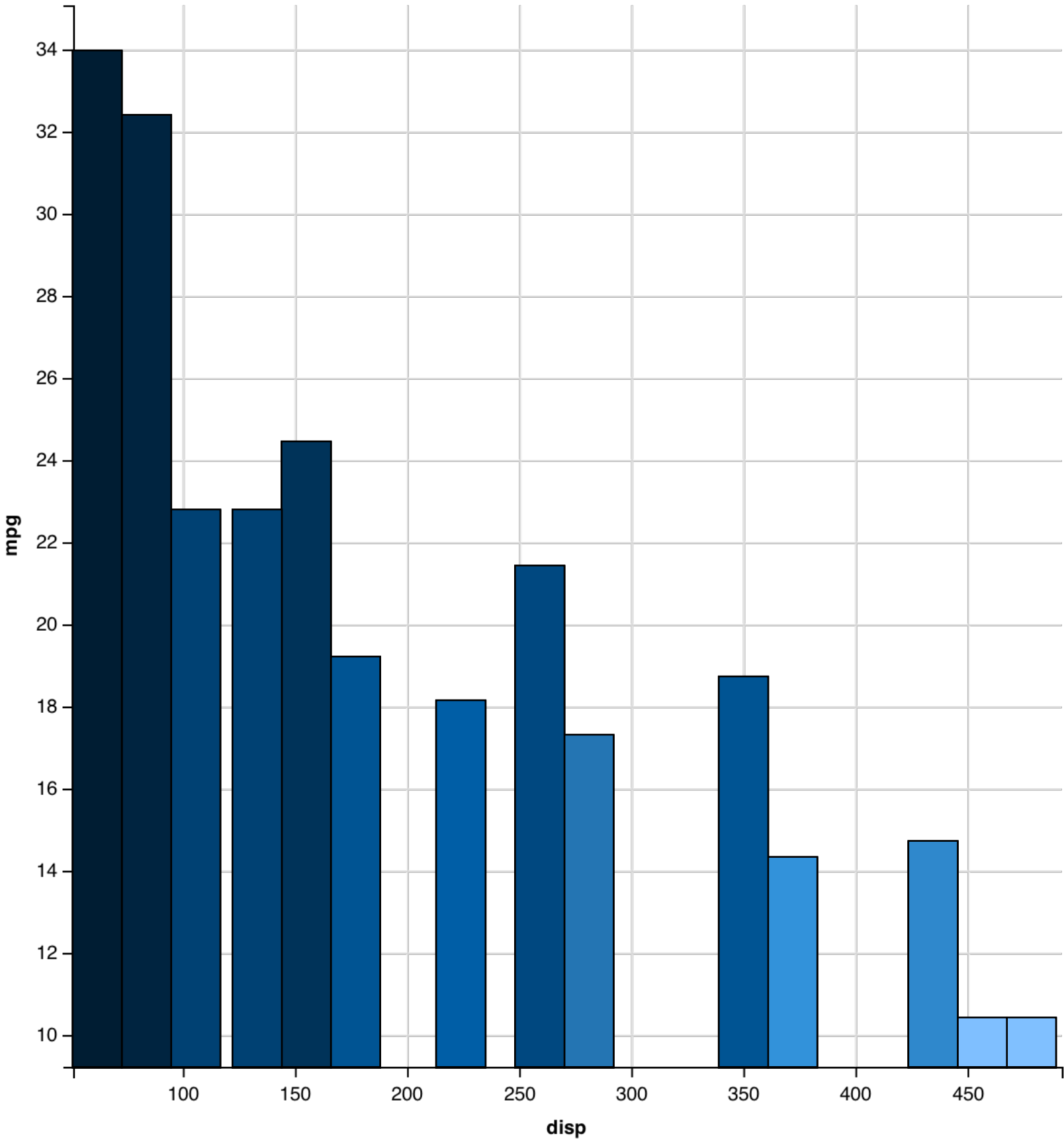
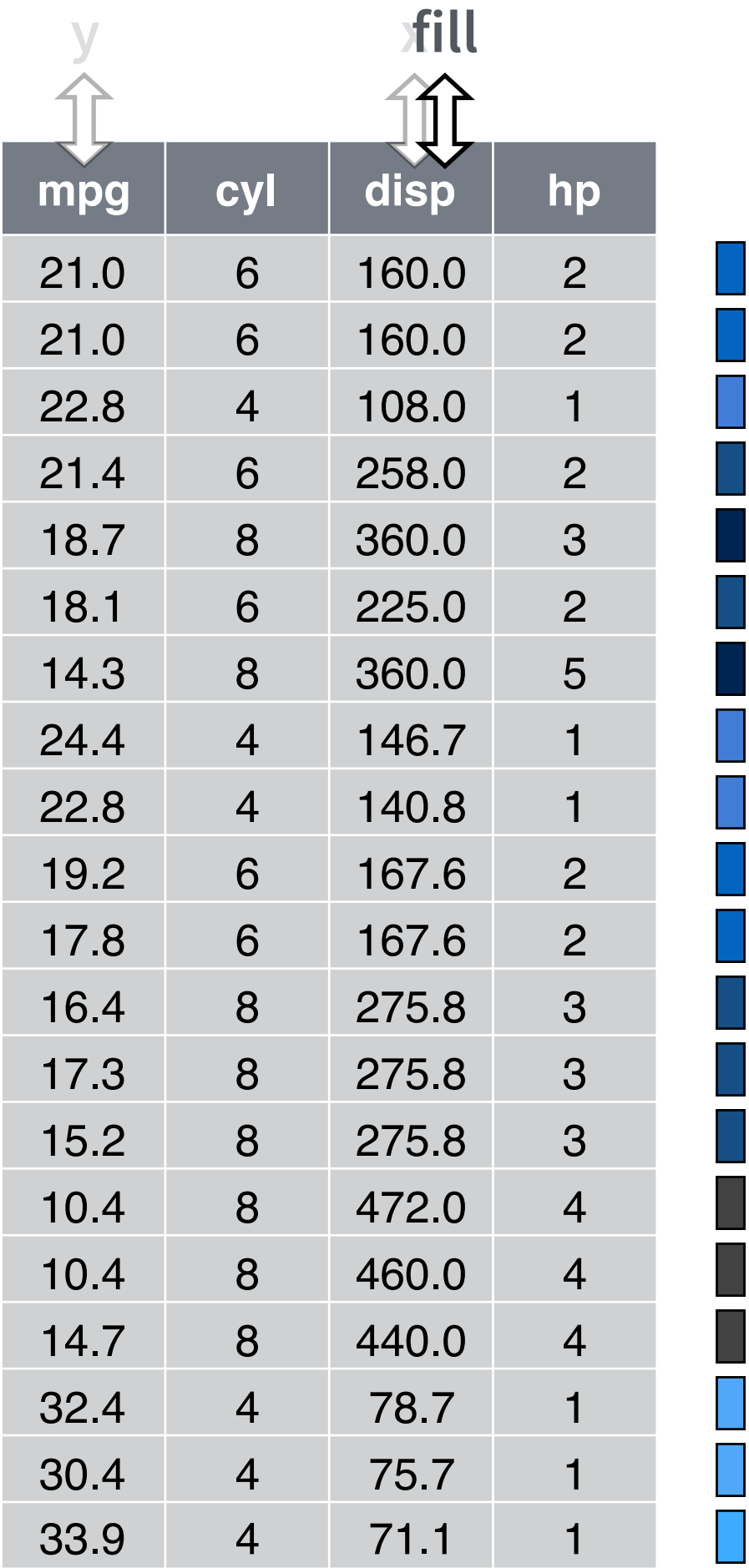


data

geom

- points
- lines
- bars

mappings



data

geom

points

lines

bars

# To make a graph

[template]

```
ggplot(data = <DATA>) +  
  <GEOM_FUNCTION>(mapping = aes(<MAPPINGS>))
```

# To make a graph

## 1. Pick a **data** set

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1

**data**

```
ggplot(data = <DATA>) +  
  <GEOM_FUNCTION>(mapping = aes(<MAPPINGS>))
```

# To make a graph

mpg	cyl	disp	hp
21.0	6	160.0	2
21.0	6	160.0	2
22.8	4	108.0	1
21.4	6	258.0	2
18.7	8	360.0	3
18.1	6	225.0	2
14.3	8	360.0	5
24.4	4	146.7	1
22.8	4	140.8	1
19.2	6	167.6	2
17.8	6	167.6	2
16.4	8	275.8	3
17.3	8	275.8	3
15.2	8	275.8	3
10.4	8	472.0	4
10.4	8	460.0	4
14.7	8	440.0	4
32.4	4	78.7	1
30.4	4	75.7	1
33.9	4	71.1	1

data

geom

1. Pick a **data** set

```
ggplot(data = <DATA>) +  
  <GEOM_FUNCTION>(mapping = aes(<MAPPINGS>))
```

2. Choose a **geom**  
to display cases



# To make a graph

mappings

mpg	cyl	disp	hp	fill	geom
21.0	6	160.0	2	blue	●
21.0	6	160.0	2	blue	●
22.8	4	108.0	1	green	●
21.4	6	258.0	2	blue	●
18.7	8	360.0	3	red	●
18.1	6	225.0	2	blue	●
14.3	8	360.0	5	purple	●
24.4	4	146.7	1	green	●
22.8	4	140.8	1	green	●
19.2	6	167.6	2	blue	●
17.8	6	167.6	2	blue	●
16.4	8	275.8	3	red	●
17.3	8	275.8	3	red	●
15.2	8	275.8	3	red	●
10.4	8	472.0	4	yellow	●
10.4	8	460.0	4	yellow	●
14.7	8	440.0	4	yellow	●
32.4	4	78.7	1	green	●
30.4	4	75.7	1	green	●
33.9	4	71.1	1	green	●

data      geom

1. Pick a **data** set

```
ggplot(data = <DATA>) +  
  <GEOM_FUNCTION>(mapping = aes(<MAPPINGS>))
```

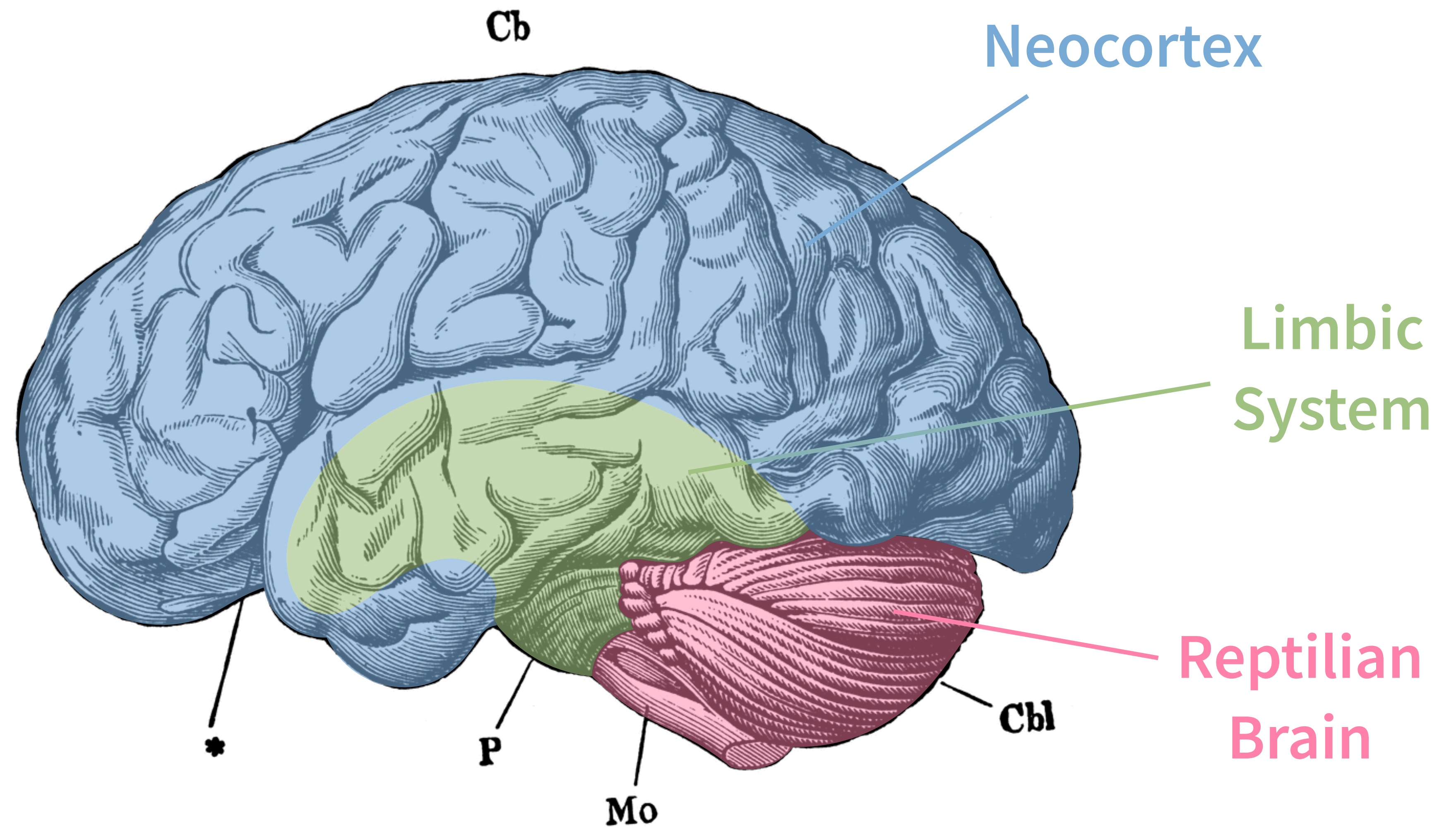
2. Choose a **geom**  
to display cases

3. **Map** aesthetic  
properties to  
variables

# Wrap up



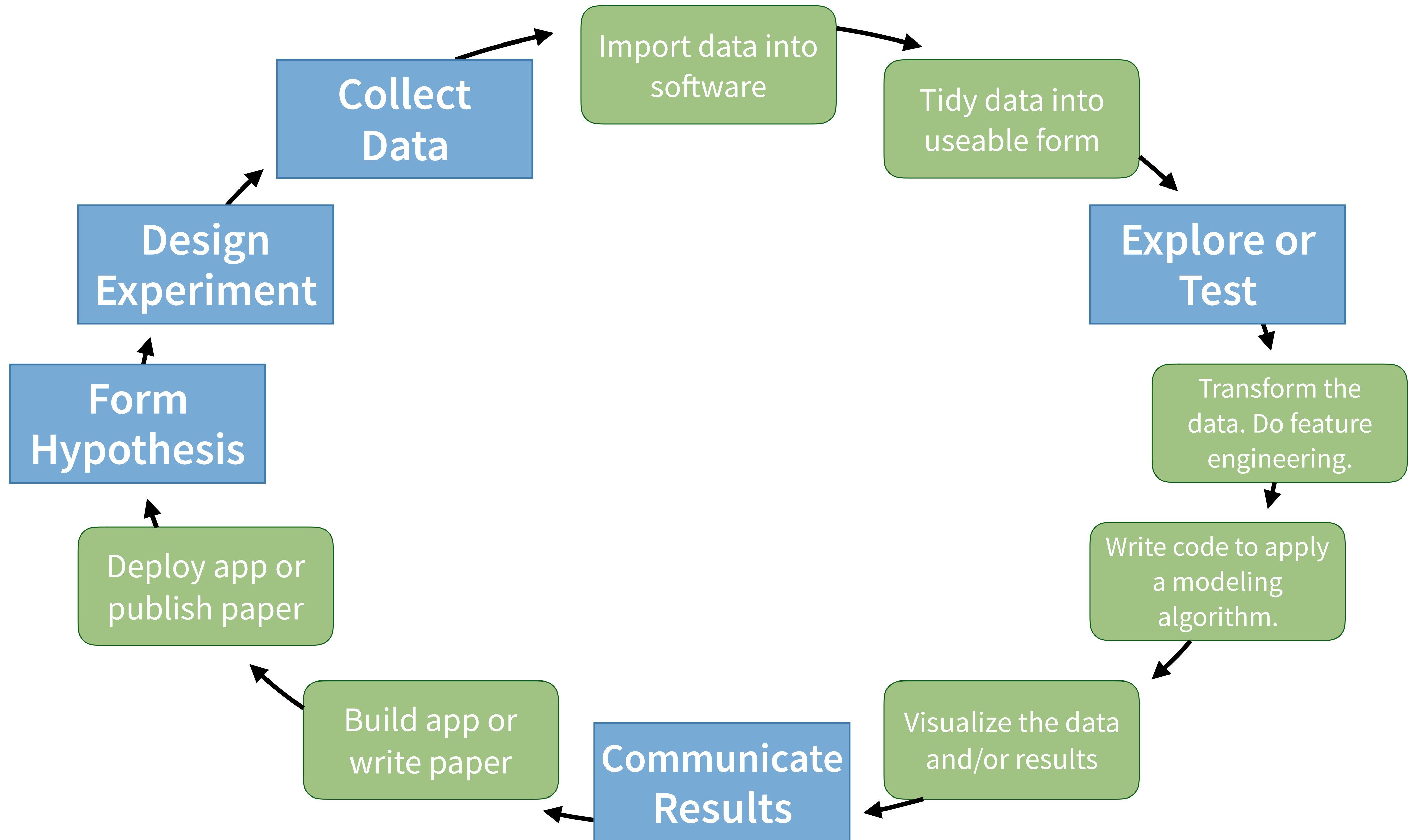




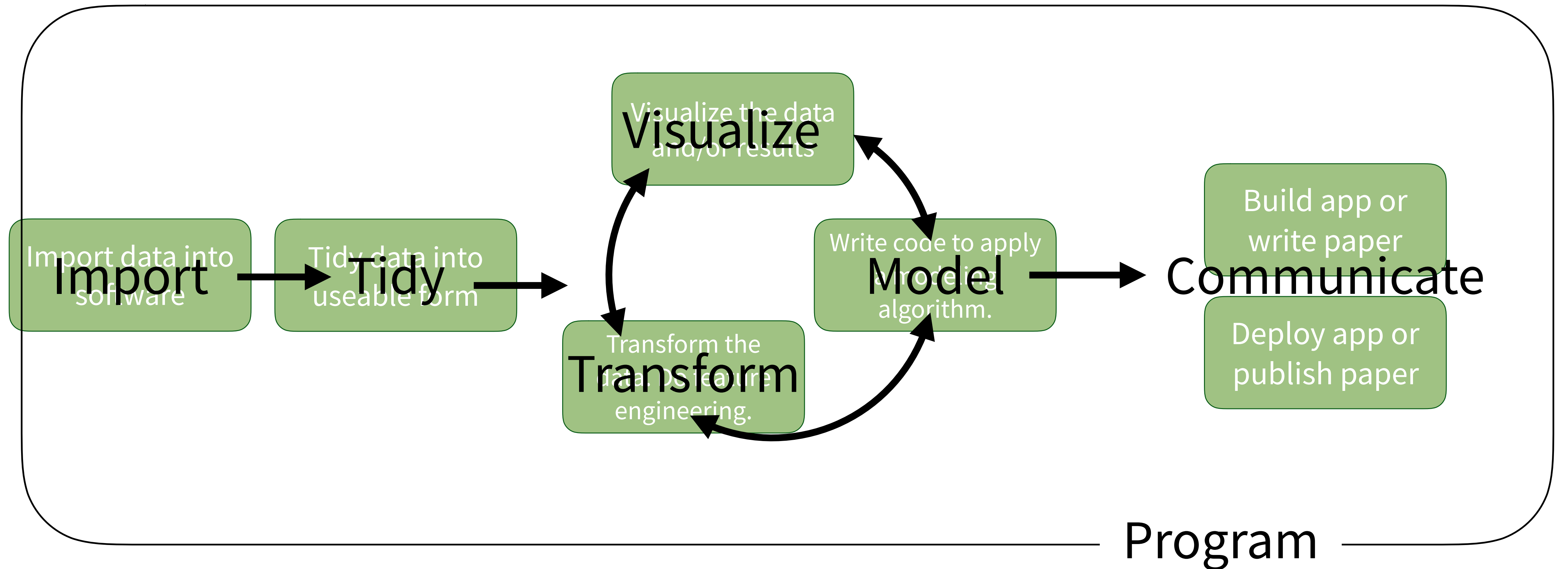




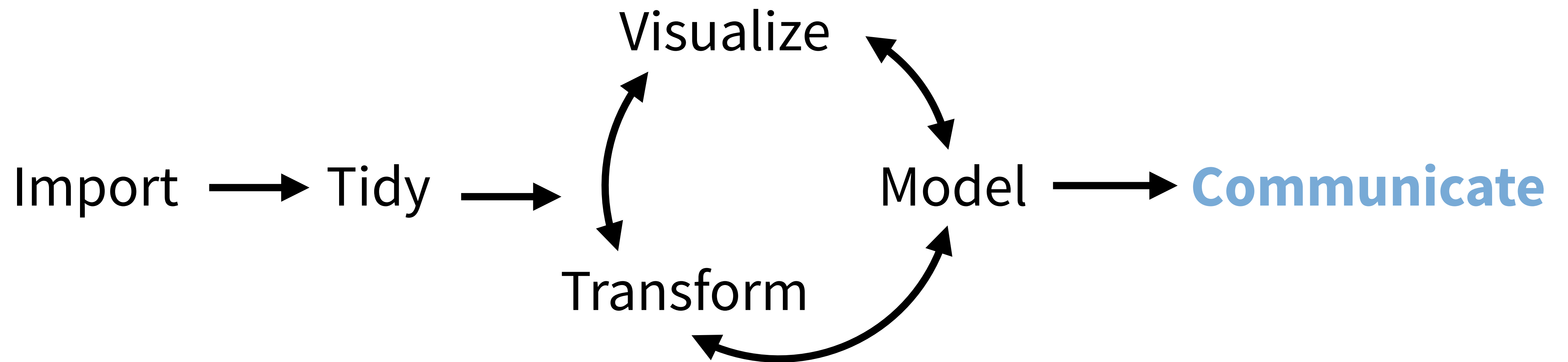




# (Applied) Data Science

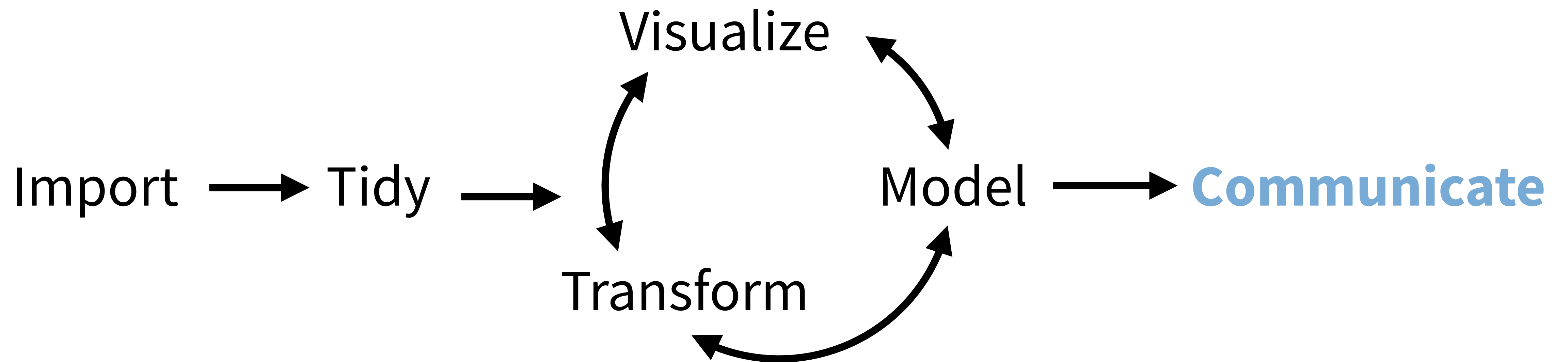


# (Applied) Data Science



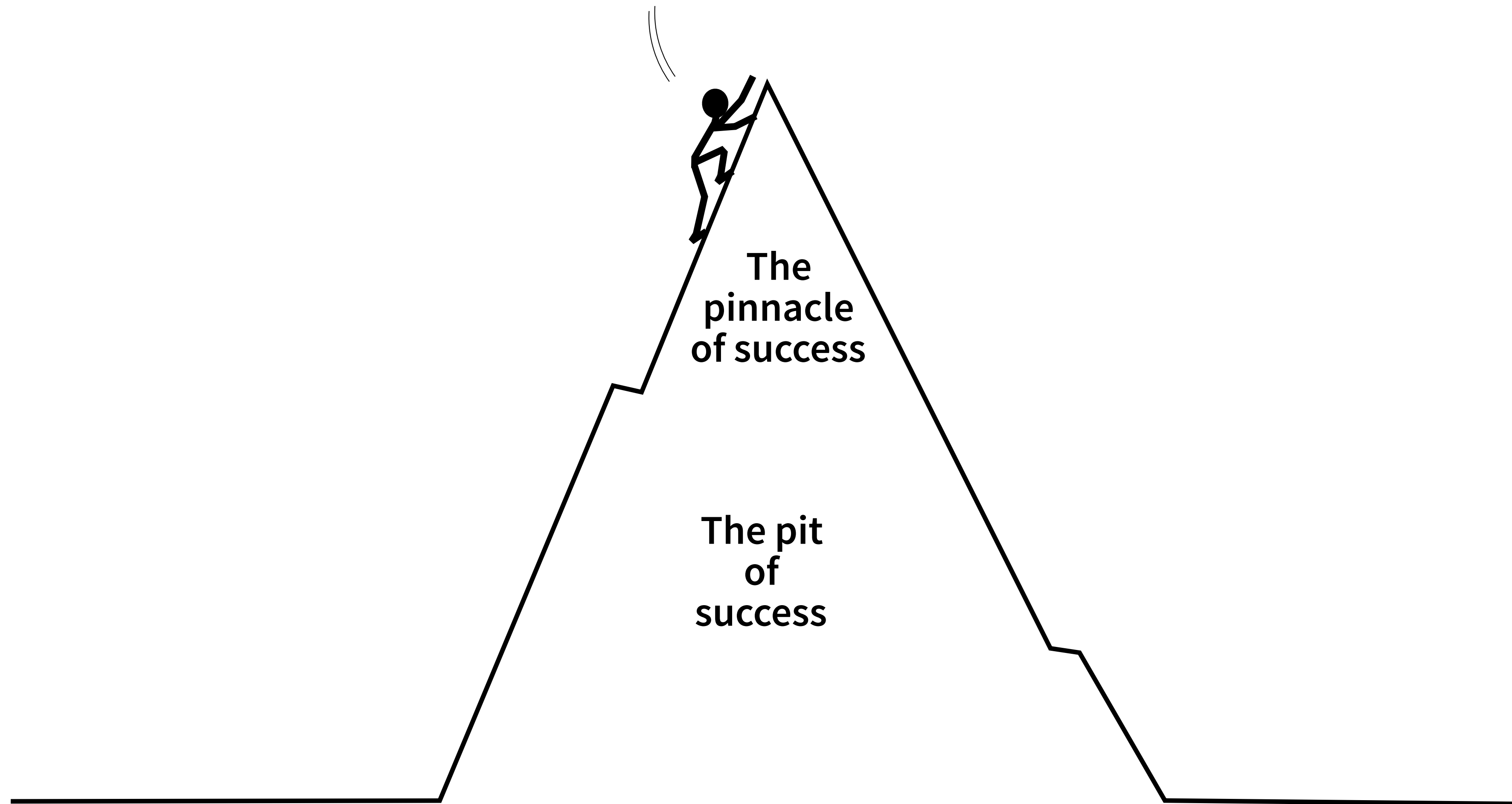
Program

# (Applied) Data Science



Program





# tidyverse.org

tidyverse website • tidyweb x Garrett

tidyverse.org

## The tidyverse

### Components

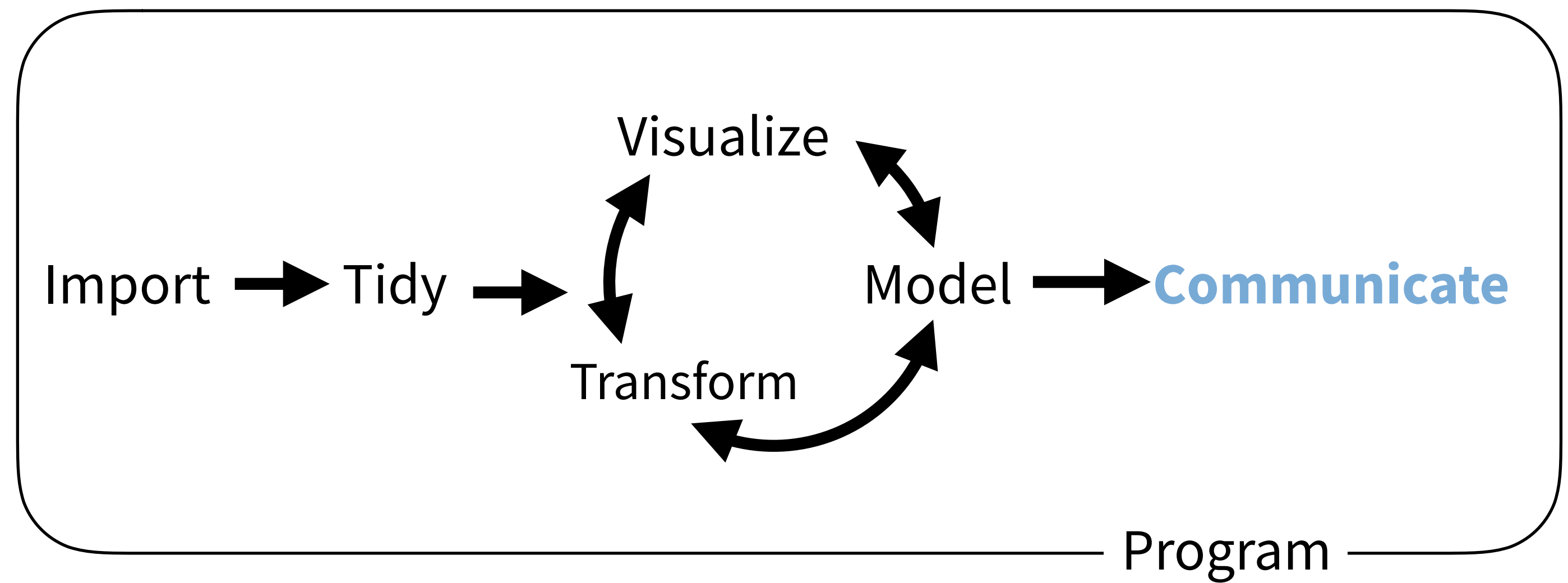
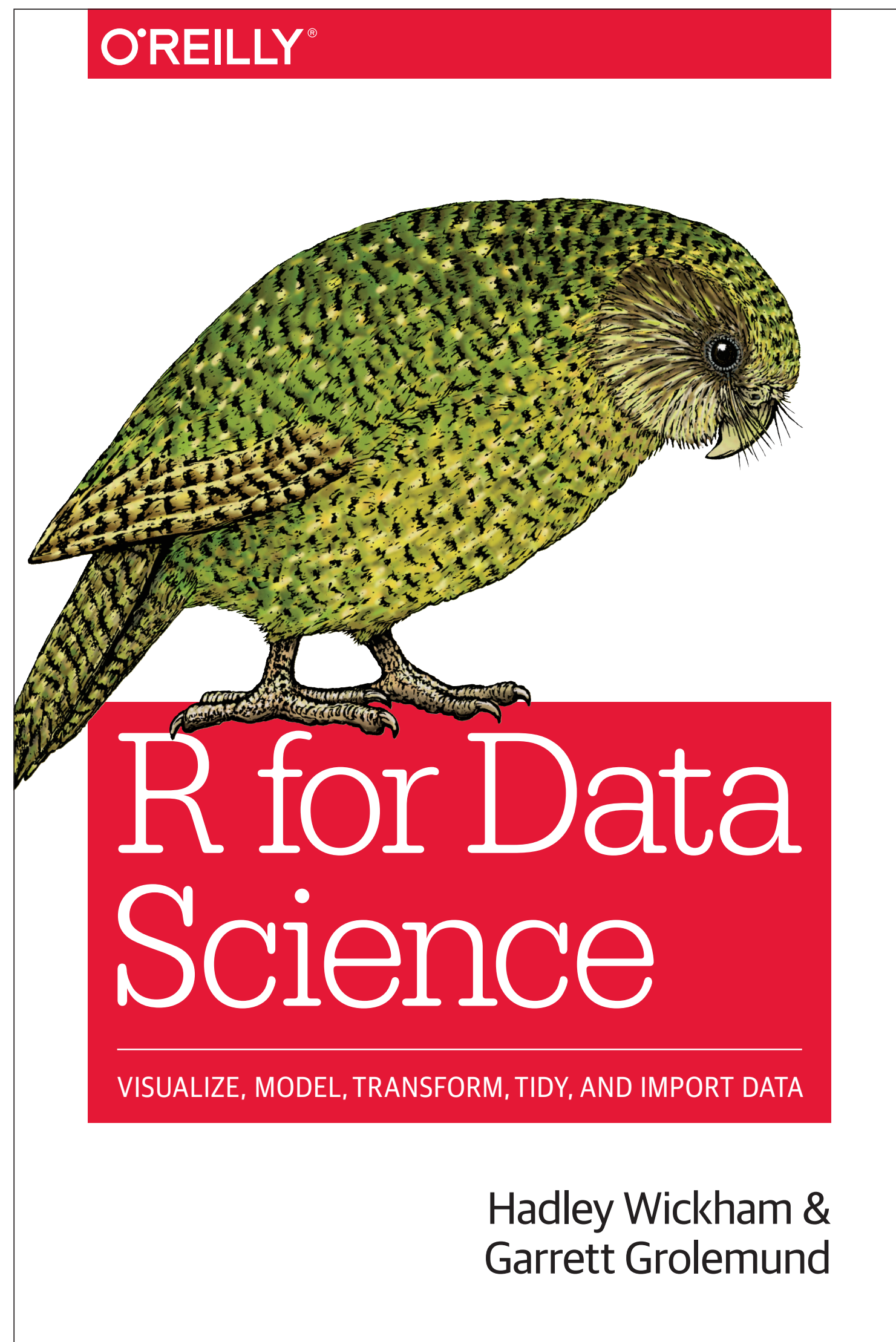
#tidyverse tweets

**PJ B** @ProfBforBargain  
I may be excommunicated for it, but readxl and lubridate are heretical packages that should not belong to the tidyverse. [#rstats](#) [#tidyverse](#)  
27m

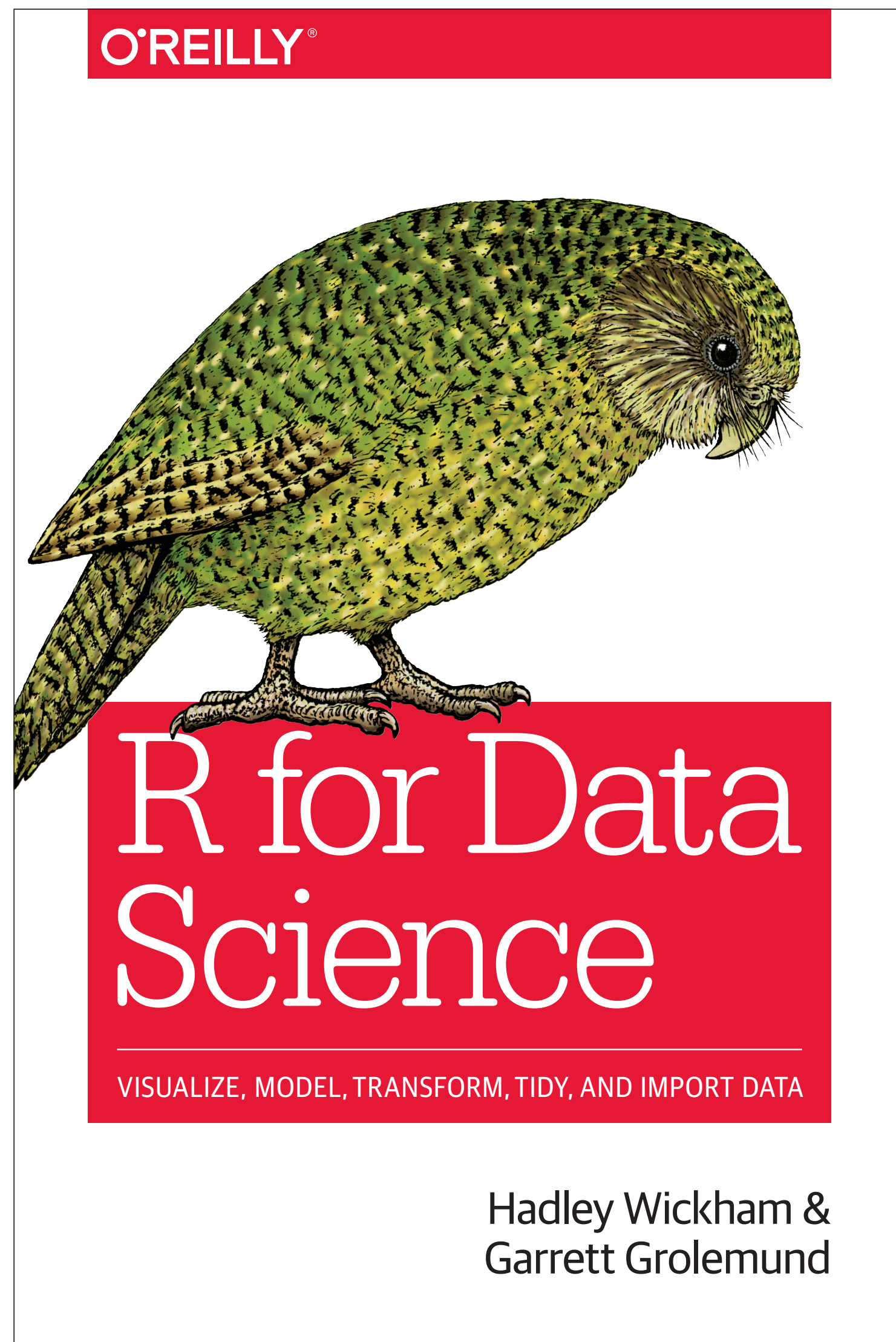
**Martin Monkman** @monkmanmh  
Essential reading as-is. I think I'll leave a paper copy on the lunch room table. [#tidyverse](#) [#datascience](#) Thanks [@kwbroman](#) & [@kara\\_woo](#)  
2h

**Hilary Robbins** @hilaryarobbins  
Raise your hand if you constantly write broken [#tidyverse](#) code because of British vs. American spellings 🙋 [#rstats](#) [#summarise](#)  
5m

The tidyverse is a collection of R packages that share common philosophies and are designed to work together. This site is a work-in-progress guide to the tidyverse and its packages.







<http://r4ds.had.co.nz/>

Browser window showing the R for Data Science website. The URL is [r4ds.had.co.nz/data-visualisation.html](http://r4ds.had.co.nz/data-visualisation.html). The page title is "R for Data Science".

Left sidebar (Table of Contents):

- Welcome
- 1 Introduction
- I Explore
- 2 Introduction
- 3 Data visualisation
  - 3.1 Introduction
  - 3.2 First steps
  - 3.3 Aesthetic mappings
  - 3.4 Common problems
  - 3.5 Facets
  - 3.6 Geometric objects**
  - 3.7 Statistical transformations
  - 3.8 Position adjustments
  - 3.9 Coordinate systems
  - 3.10 The layered grammar of graphics
- 4 Workflow: basics
- 5 Data transformation
- 6 Workflow: scripts

Main content area:

### 3.6 Geometric objects

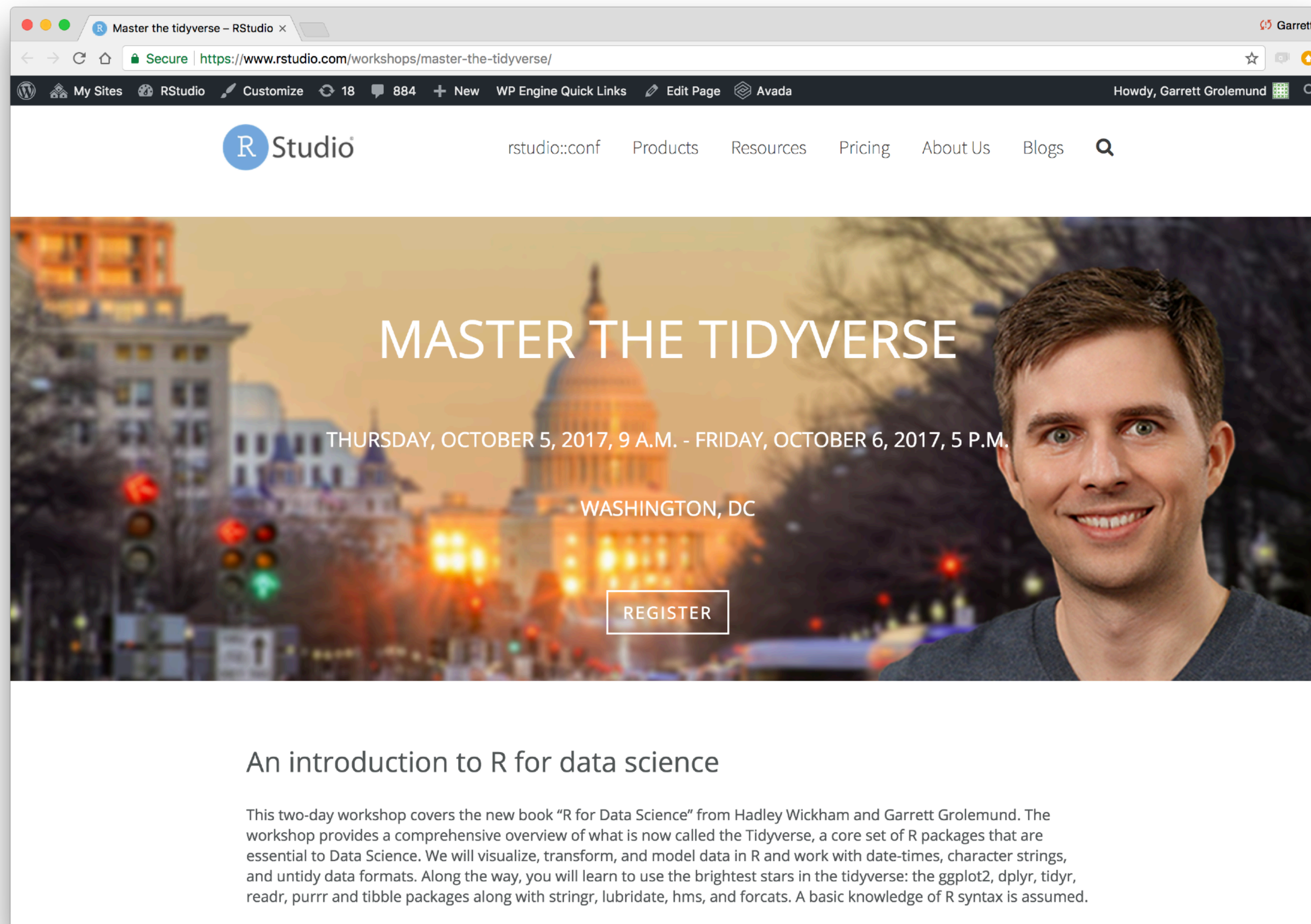
How are these two plots similar?

Both plots contain the same x variable, the same y variable, and both describe the same data. But the plots are not identical. Each plot uses a different visual object to represent the data. In ggplot2 syntax, we say that they use different **geoms**.

A **geom** is the geometrical object that a plot uses to represent data. People often describe plots by the type of geom that the plot uses. For example, bar charts use bar geoms, line charts use line geoms, boxplots use boxplot geoms, and so on. Scatterplots break the trend; they use the point geom. As we see above, you can use different geoms to plot the same data. The plot on the left uses the point geom, and the plot on the right uses the smooth geom, a smooth line



# Thank You



The screenshot shows a web browser window with the URL <https://www.rstudio.com/workshops/master-the-tidyverse/>. The page features the RStudio logo and navigation links: rstudio::conf, Products, Resources, Pricing, About Us, and Blogs. The main content area has a background image of the US Capitol building at night and a portrait of Garrett Golemund. The text on the page reads: "MASTER THE TIDYVERSE", "THURSDAY, OCTOBER 5, 2017, 9 A.M. - FRIDAY, OCTOBER 6, 2017, 5 P.M.", "WASHINGTON, DC", and a "REGISTER" button. Below this, a section titled "An introduction to R for data science" describes the workshop as a two-day event covering the book "R for Data Science" by Hadley Wickham and Garrett Golemund, focusing on the Tidyverse ecosystem.

MASTER THE TIDYVERSE

THURSDAY, OCTOBER 5, 2017, 9 A.M. - FRIDAY, OCTOBER 6, 2017, 5 P.M.

WASHINGTON, DC

REGISTER

An introduction to R for data science

This two-day workshop covers the new book "R for Data Science" from Hadley Wickham and Garrett Golemund. The workshop provides a comprehensive overview of what is now called the Tidyverse, a core set of R packages that are essential to Data Science. We will visualize, transform, and model data in R and work with date-times, character strings, and untidy data formats. Along the way, you will learn to use the brightest stars in the tidyverse: the ggplot2, dplyr, tidyr, readr, purrr and tibble packages along with stringr, lubridate, hms, and forcats. A basic knowledge of R syntax is assumed.

[www.rstudio.com/workshops/](https://www.rstudio.com/workshops/)