Data Science
CS301

Playing With R

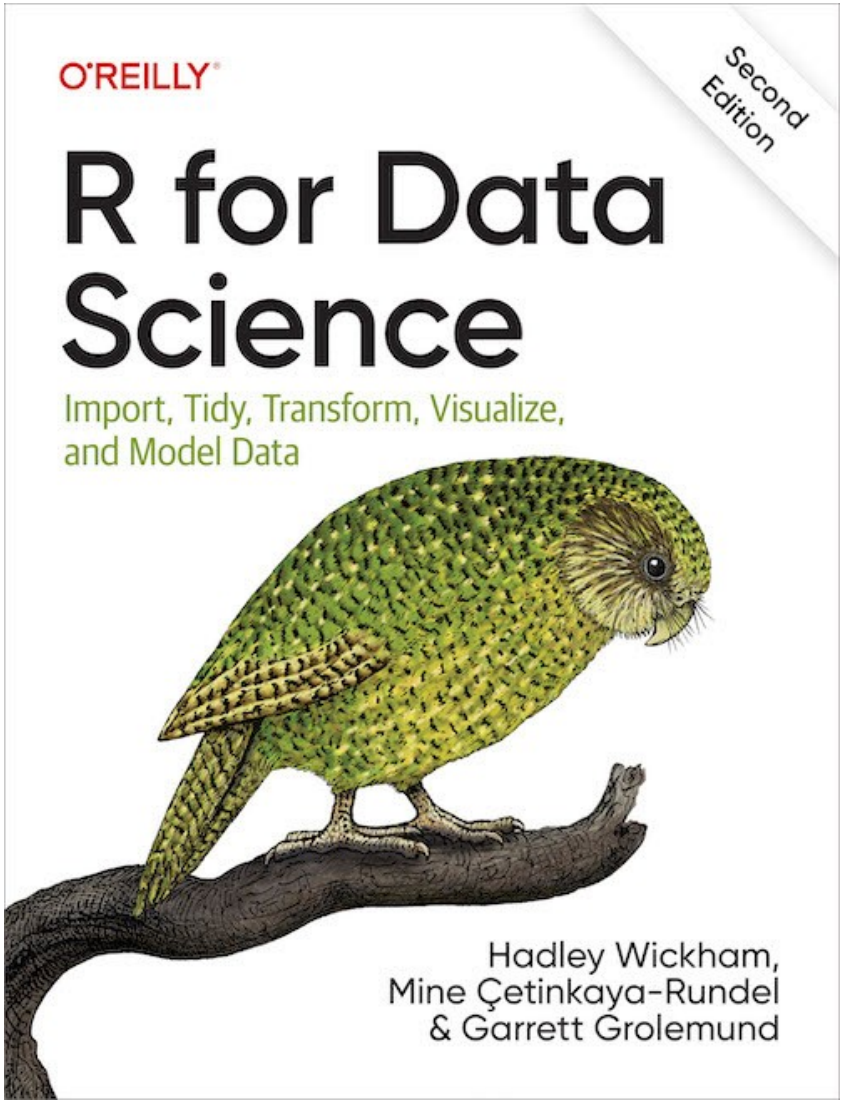
Week 3
Fall 2024
Oliver BONHAM-CARTER

For Your Own Analysis?

- **BUT! What if you are working on a project and no tools currently exist?!**

Develop Your Own Tools!!

Where in the Web?



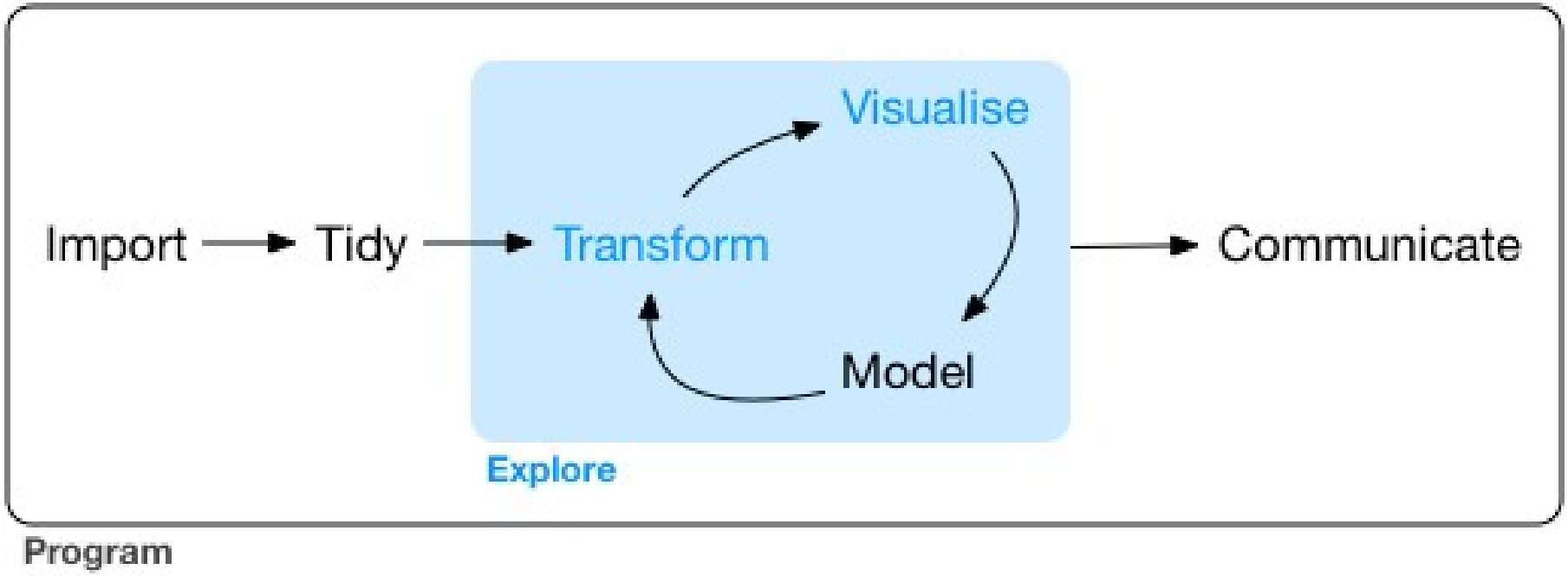
Web:

Chap 1: Data Visualization

– https://r4ds.hadley.nz/data-visualize

Explore the Data Of Your World

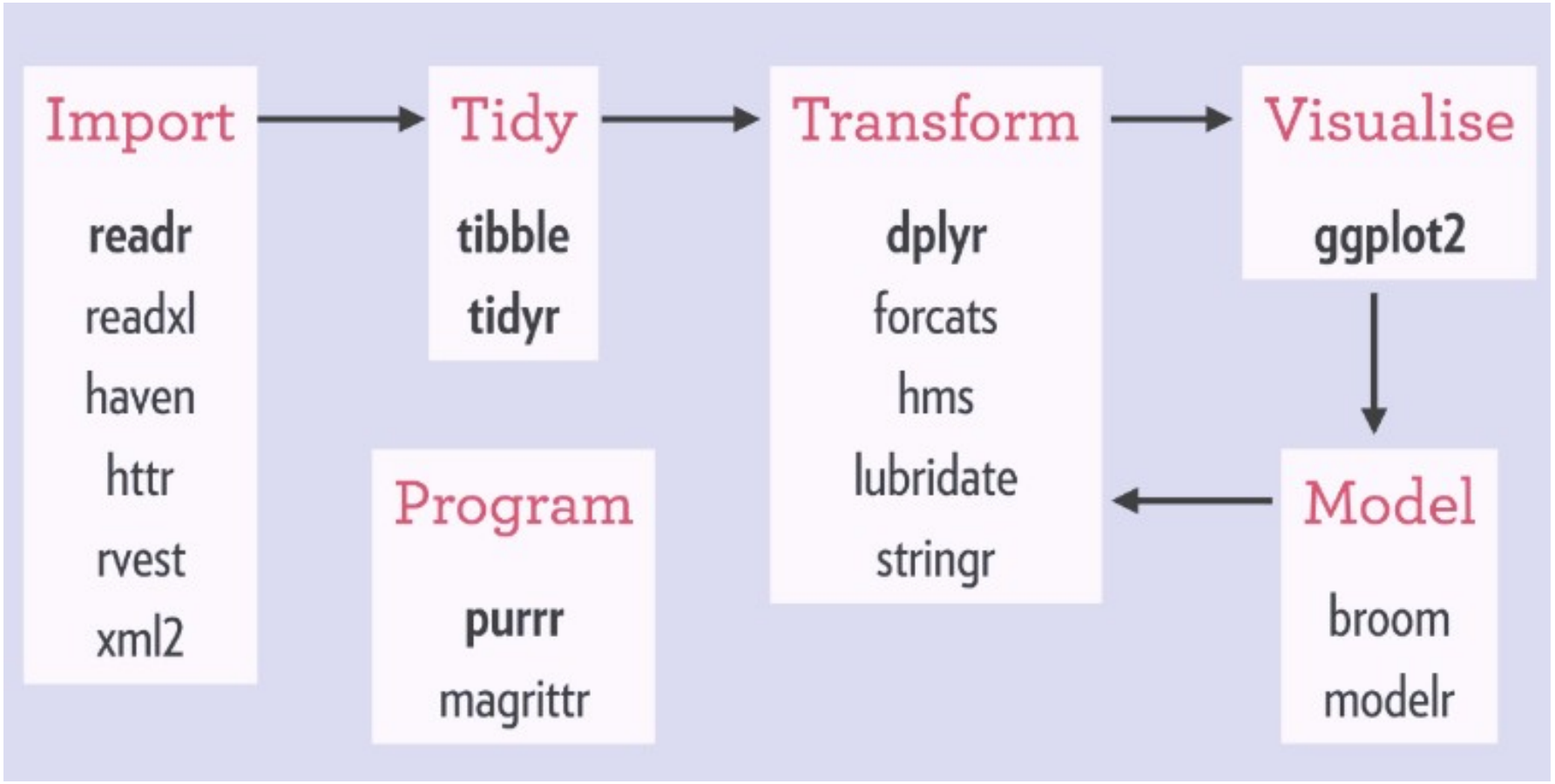
"Data exploration is the art of looking at your data, rapidly generating hypotheses, testing them, then repeating again and again..."



Import : Bringing in the raw data to work on it
Tidy: Cleaning it up so that numbers are numbers and etc.
Transform: Converting the data into something more *convenient* to use
Visualize: Finding general trends in data
Model: Testing phases, learning how to predict from the data.
Communicate: Publish and change the world!

Tidyverse's Packages

The steps of the Tidyverse canonical data science workflow, as well as, the individual packages that the steps involve.



Go Visual!

The Tidyverse library in R: a coherent system of packages for data manipulation, exploration and visualization

https://www.tidyverse.org/

Install the Library

```
> install.packages("tidyverse")
also installing the dependencies ‘colorspace’, ‘sys’
t’, ‘ps’, ‘sass’, ‘cachem’, ‘memoise’, ‘base64enc’,
st’, ‘fastmap’, ‘farver’, ‘labeling’, ‘munsell’, ‘RC
rewer’, ‘viridisLite’, ‘rematch’, ‘askpass’, ‘bit64’
ettyunits’, ‘processx’, ‘evaluate’, ‘highr’, ‘yaml’,
n’, ‘bslib’, ‘htmltools’, ‘jquerylib’, ‘tinytex’, ‘b
rts’, ‘ellipsis’, ‘generics’, ‘glue’, ‘assertthat’,
```

- For the **first** use, you need to **install** the library software to your computer with,
  - *Install.packages(“tidyverse”)*

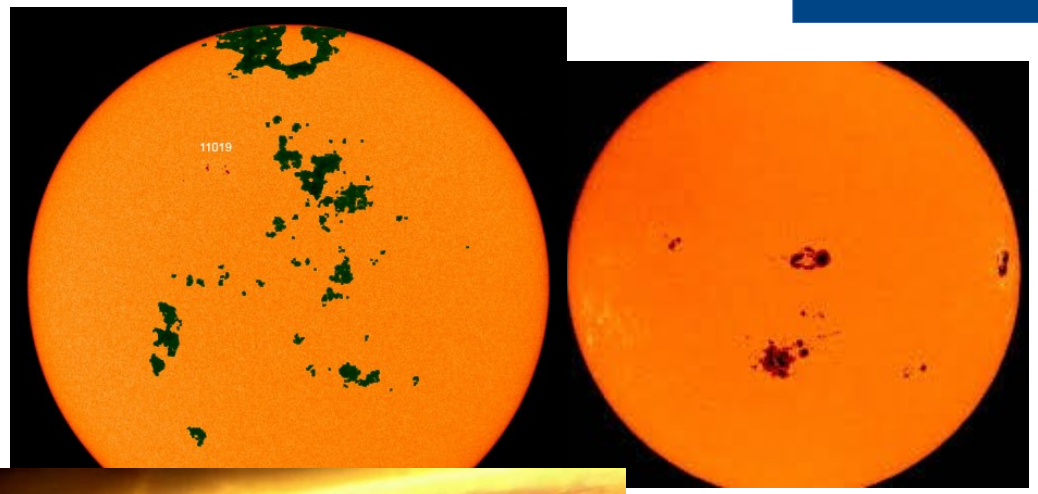
Load the Library

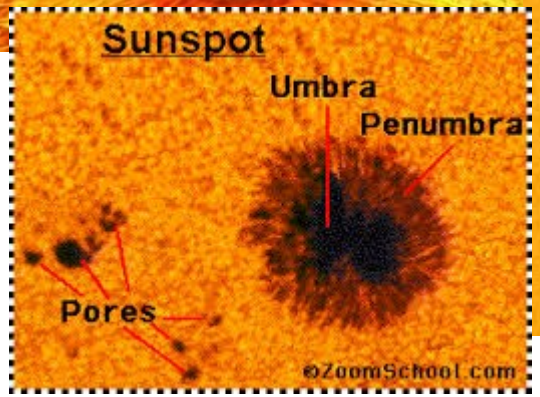
```
> library(tidyverse)
── Attaching packages ─────────────── tidyverse 1.3.2 ──
✔ ggplot2 3.4.0      ✔ purrr   1.0.1
✔ tibble  3.1.8      ✔ dplyr   1.0.10
✔ tidyr   1.3.0      ✔ stringr 1.5.0
✔ readr   2.1.3      ✔ forcats 0.5.2
── Conflicts ──────────────────── tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
```

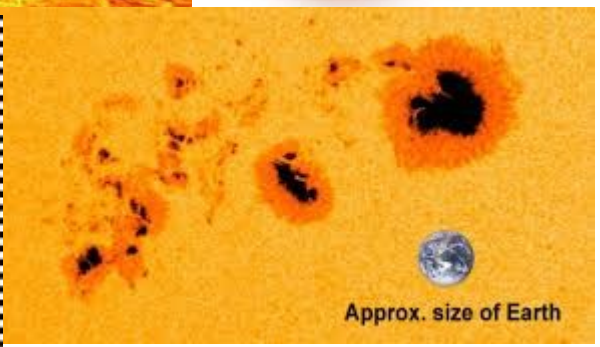
- Once installed, you only need to **call** (or **load**) the library with,
  - *library(tidyverse)*

Exploring Sun-Spot Data

- Sunspots – magnetic disturbances on the sun that can be observed from Earth
- Spots cycles are noted to repeatedly increase and then decrease over time







Articulating the Research Question

- Is there a pattern of any *kind* in this data?
- Is there something periodic about the sunspot data?
- Can we collect come evidence of a pattern in the data?
- Could we use this pattern to predict?
- What does a pattern look like in the data?





Load and Plot Sunspot Data

```
#Load library
library(tidyverse)

# find your sandbox file
sunData <- read.table(file.choose(), header =
TRUE, sep = ",")

# See what the data looks like
View(sunData)
```

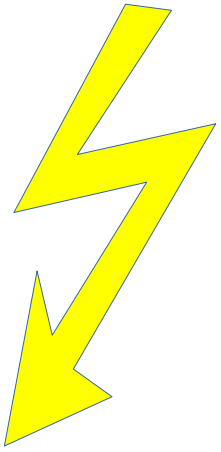
```
# Plot the data:
ggplot(data = sunData) +  geom_point(mapping = aes(x =
fracOfYear, y = sunspotNum))

# Save a file to the Desktop/ (or wherever) if you
want...
ggsave("~/Desktop/myplot.png")
```

file: sandbox/sunspots_data.csv.r

Try This Plot!

```
ggplot(data = sunData) +  geom_point(
    mapping = aes(
      x = fracOfYear,
      y = sunspotNum,
      color = numObs
      )
    )
```

What did you find?
Change your *x* and *y axis coordinates to continue exploring!*



```
ggplot(data = mpg) +
geom_point( mapping = aes( x = displ, y = hwy) )
```

Link to the data (set is called, 'mpg')

- ggplot(data = mpg)

Function to design a scatter plot

- geom_point()

Compute the geometry of point placement on canvas

- mapping = ...

Compute the aesthetics of the plot (titles, color, point type, etc)

- aes(x = displ, y = hwy)

Consider these ...

```
names(sunData)

ggplot(data = sunData) +  geom_point(mapping = aes(x = fracOfYear, y =
sunspotNum))

# Add a smooth line to see general trends
ggplot(data = sunData) +  geom_point(mapping = aes(x = fracOfYear, y =
sunspotNum)) + geom_smooth(mapping = aes(x = fracOfYear, y =
sunspotNum))

# Color by year
ggplot(data = sunData) +  geom_point(mapping = aes(x = fracOfYear, y =
sunspotNum, color = fracOfYear)) + geom_smooth(mapping = aes(x =
fracOfYear, y = sunspotNum))

# Color by month
ggplot(data = sunData) +  geom_point(mapping = aes(x = fracOfYear, y =
sunspotNum, color = month)) + geom_smooth(mapping = aes(x = fracOfYear,
y = sunspotNum, color = fracOfYear))
```

Run this code
to make other plots.
What do you see?

More Practice?!

```
#Load library
library(tidyverse)

# find and then choose a new dataset
data() # I choose ChickWeight

# See what the data looks like
View(ChickWeight)

# Make a plot!
ggplot(data = ChickWeight) +  geom_point(mapping = aes(x
= weight, y = Chick, color = Time))

# make more plots, then choose a new dataset,and repeat
```

Change you *x* and *y* to make other plots.
What do you see?

file: sandbox/chickWeight.r

Even More Practice?!

```
# Select other datasets from the R to try out some
plots

# Find a set
data()

# See what the data looks like
View(myChosenDataSet)

# Adapt your ggplot() function to plot something
```