BINF 5003: Data Mining, Modeling, and Biostatistics

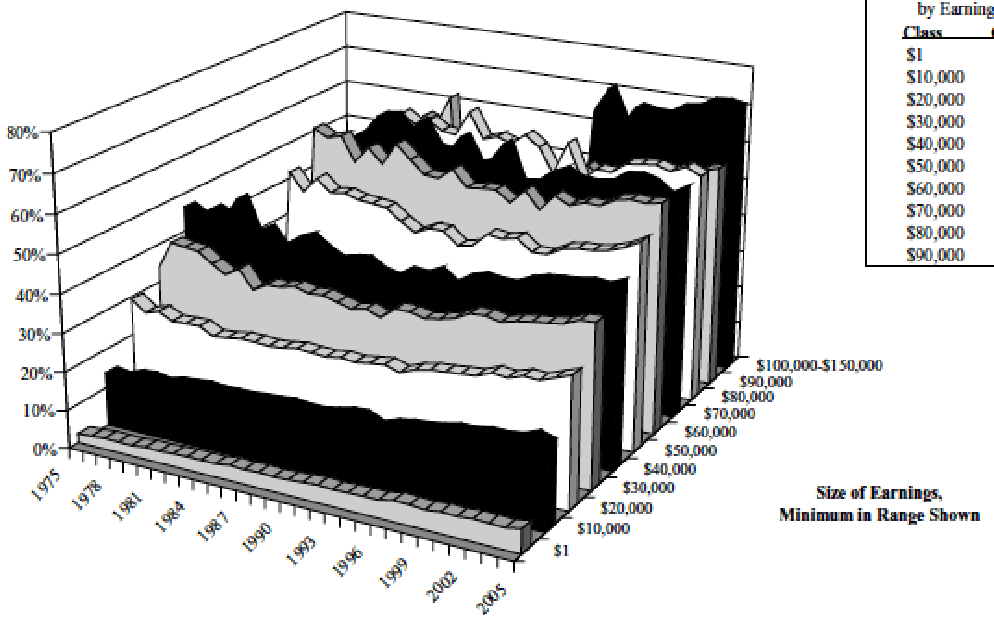
Week 5

Module 3 – Visualization

Overview

- Anatomy of a “good” figure
- Harnessing the strengths of coding – repetitive tasks
- Formatting a whole figure panel
- Wrangling + visualization

What do we like or dislike about this plot?



Inset: Percent of People Rounding by Earnings Class in 1990

Class	CPS	IPUMS
$1	4.0%	4.8%
$10,000	16.0%	18.2%
$20,000	25.8%	25.4%
$30,000	34.0%	32.5%
$40,000	37.8%	36.2%
$50,000	46.0%	42.2%
$60,000	56.7%	50.6%
$70,000	64.3%	49.8%
$80,000	60.4%	52.9%
$90,000	11.1%	47.7%

Source: Author's calculations, March CPS, various years.

Fig. 1. Average Propensity to Round Earnings by Year and Earnings Group.

What do we like or dislike about this plot?

Like

- One nicely labeled axis
- There is a figure legend

Like to improve

- Raw data is not necessary in the main figure of a plot
- What do the colours mean?
- Label the other axis

What do we like or dislike about this plot

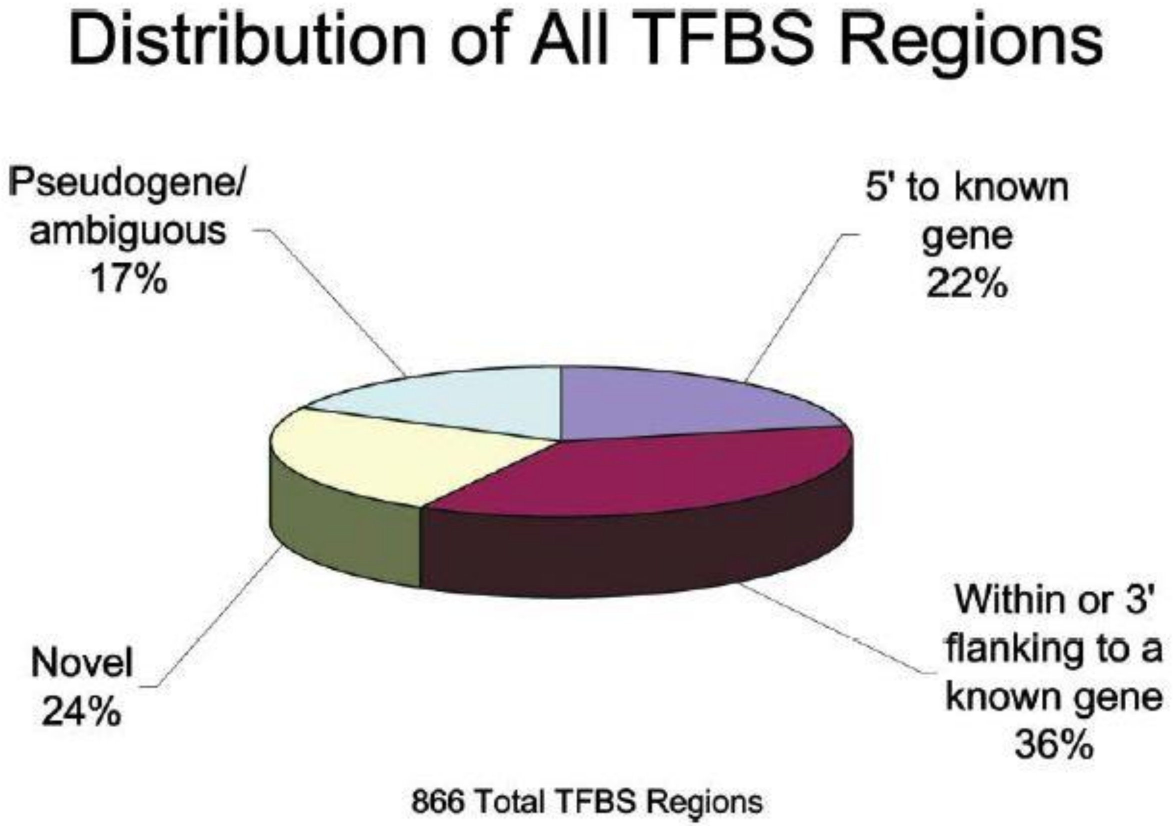


Figure 1. Classification of TFBS Regions

TFBS regions for Sp1, cMyc, and p53 were classified based upon proximity to annotations (RefSeq, Sanger hand-curated annotations, GenBank full-length mRNAs, and Ensembl predicted genes). The proximity was calculated from the center of each TFBS region. TFBS regions were classified as follows: within 5 kb of the 5′ most exon of a gene, within 5 kb of the 3′ terminal exon, or within a gene, novel or outside of any annotation, and pseudogene/ambiguous (TFBS overlapping or flanking pseudogene annotations, limited to chromosome 22, or TFBS regions falling into more than one of the above categories).

What do we like or dislike about this plot?

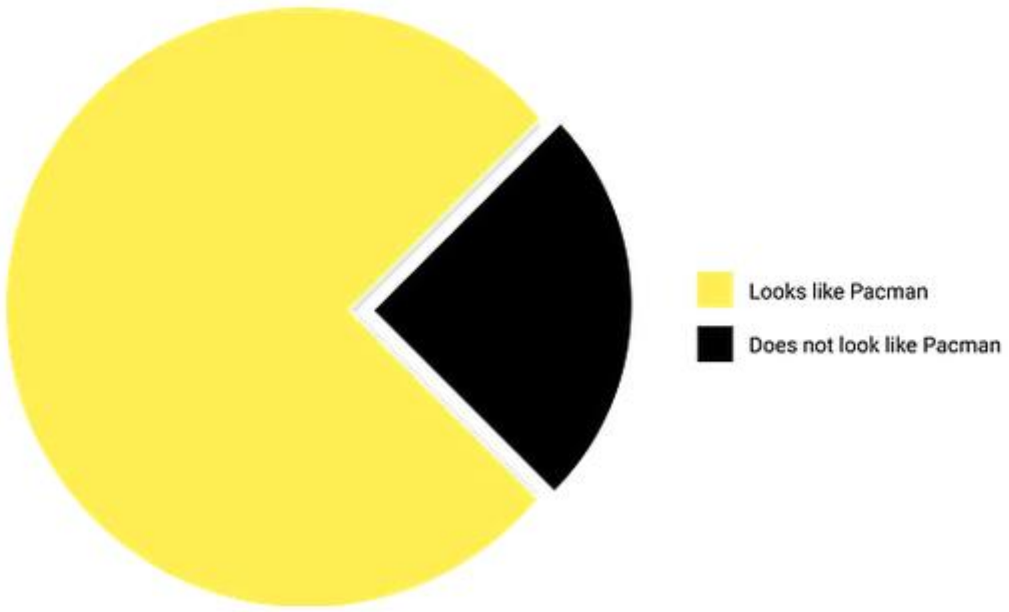
Like

- Slices are clearly labelled
- Explanation in the figure legend is helpful

Like to improve

- Pie charts make it difficult to compare the groups
  - 22% vs 24%
- 3D often adds nothing but confusion to interpreting a plot

One of the few good reasons to use a Pie Chart



Shifting gears: Repetitive tasks

- When you do a task 3 or more times, it is worth considering:
  - Optimizing your code to be generalizable
  - Writing a custom function

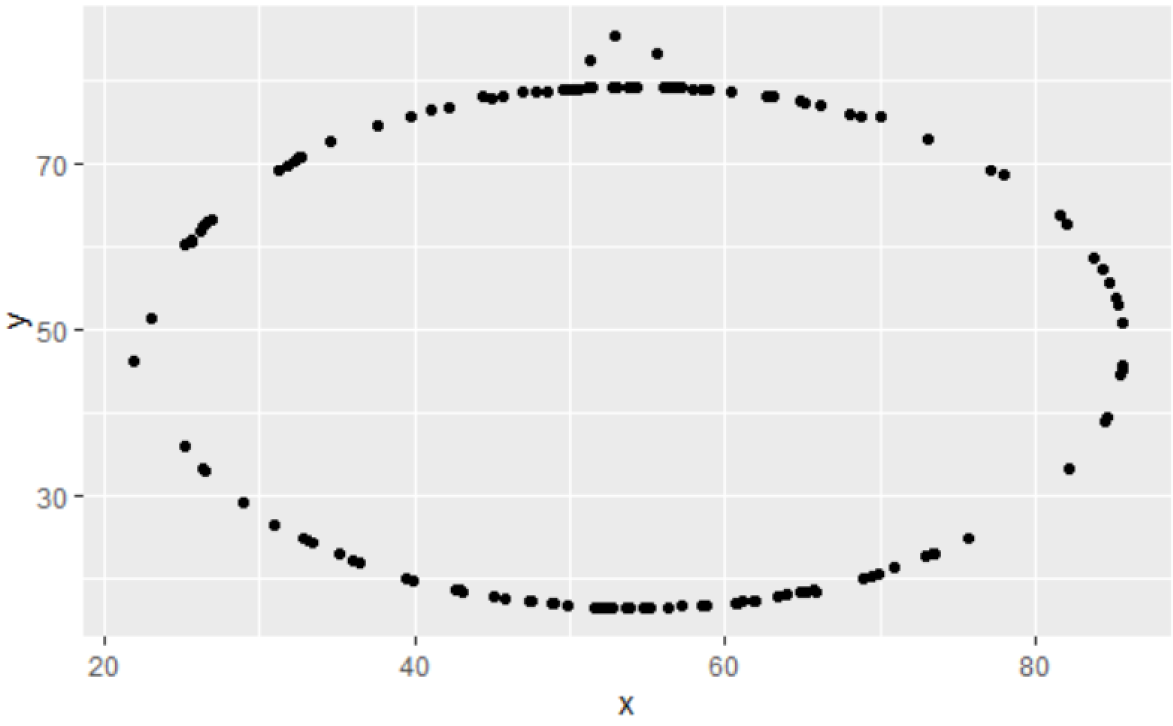
- Save time
- Prevent errors when copy and pasting or typos

Intermediate technique: Generalized code

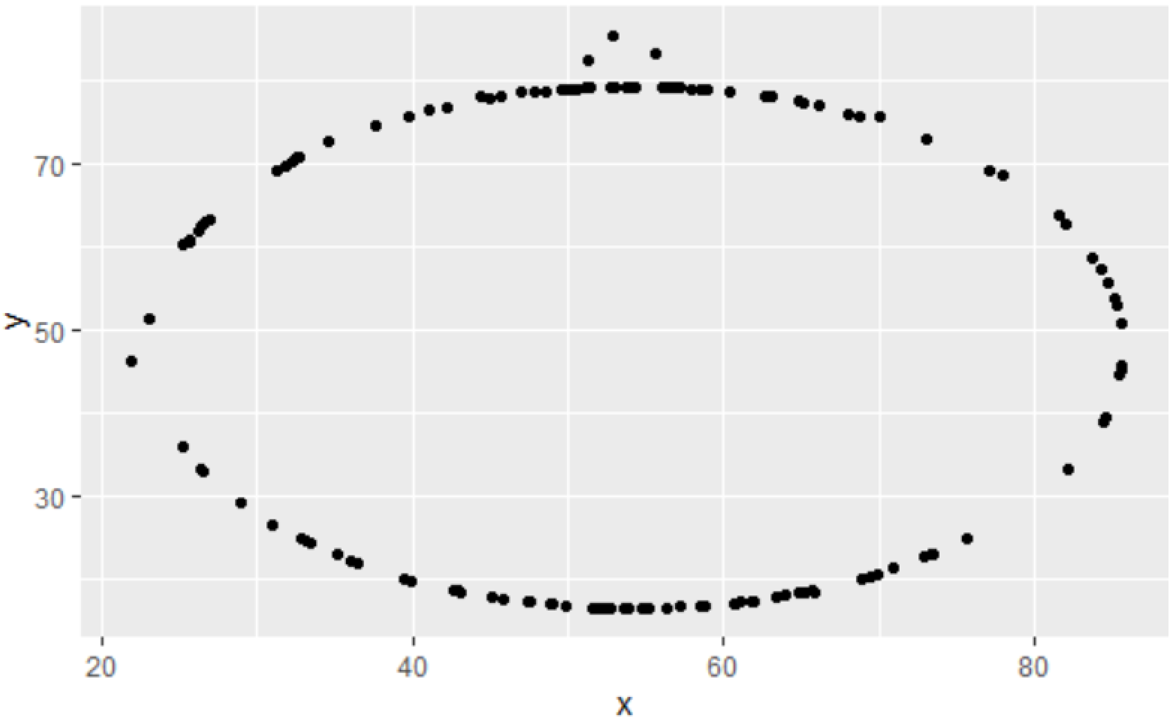
- Move all the variables that will be modified to the start of the code
  - Similar to loading in the libraries and data at the start of the workflow
- May not have a large impact on simple code, but will simplify complex code
- Still start with writing one instance of your code before converting it to be generalizable

Move edited variables to the start

```{r}
datasaurus_dozen %>%
  filter(dataset == "circle") %>% # change object name
  ggplot(aes(x=x, y=y)) +
  geom_point()
```




```{r}
dataset_name <- "circle" # new addition
datasaurus_dozen %>%
  filter(dataset == dataset_name) %>% # change object name
  ggplot(aes(x=x, y=y)) +
  geom_point()
```



Intermediate technique: Custom functions

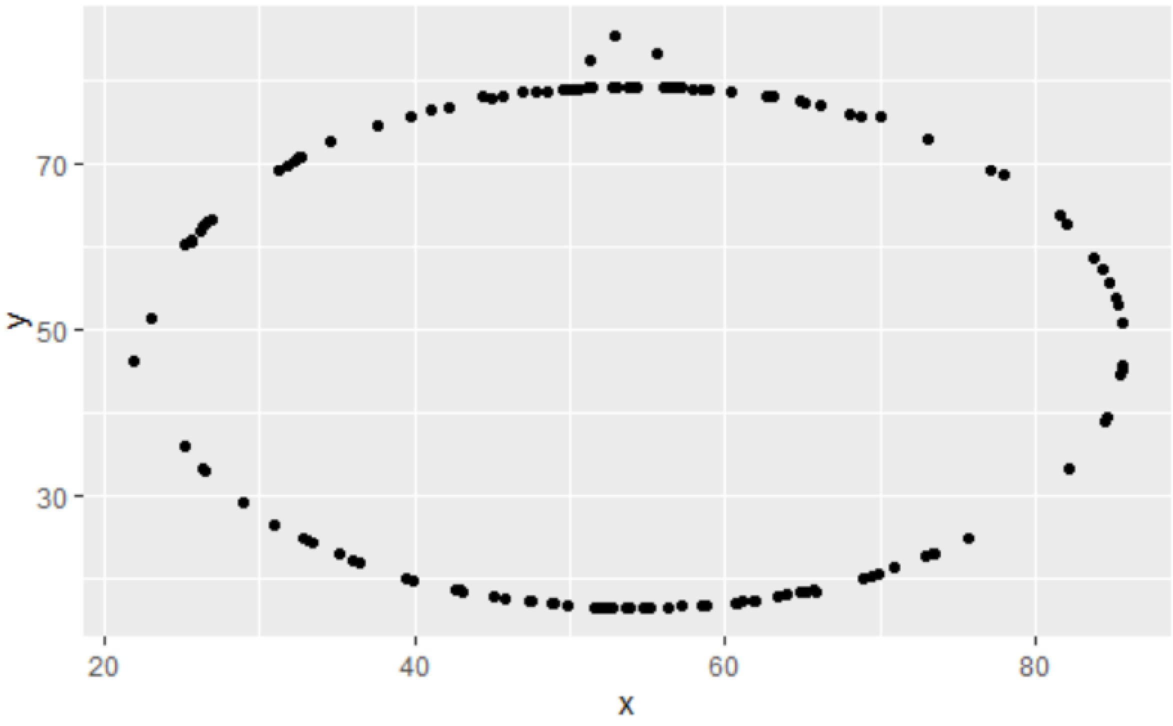
- Really easy to convert from generalized code to a custom function
  - Can be a difficult jump from unformatted code straight to a function!

- Use the function `function()` to create a new function

- Curly brackets {} to enter multi-line command
  - Indentations can help keep your code organized

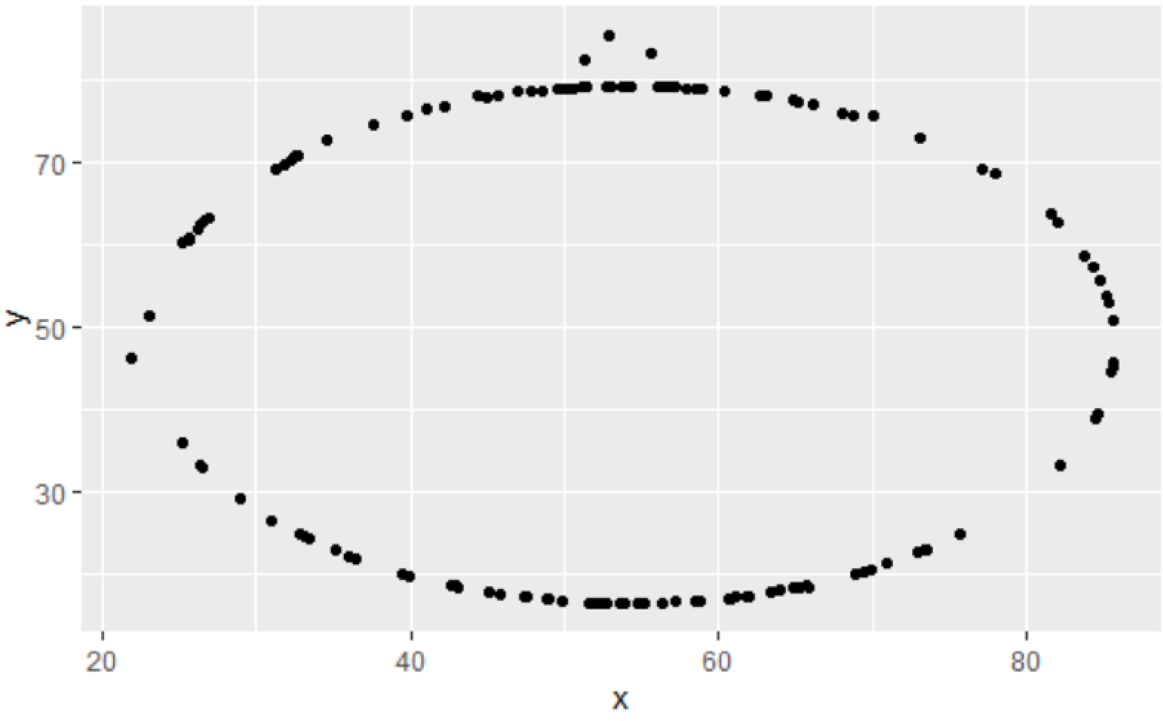
Custom functions

```{r}
datasaurus_dozen %>%
  filter(dataset == "circle") %>% # change object name
  ggplot(aes(x=x, y=y)) +
  geom_point()
```




```{r}
dino_plot <- function(data_name) {
  datasaurus_dozen %>%
    filter(dataset == data_name) %>% # change object name
    ggplot(aes(x=x, y=y)) +
    geom_point()
}

dino_plot("circle")
```

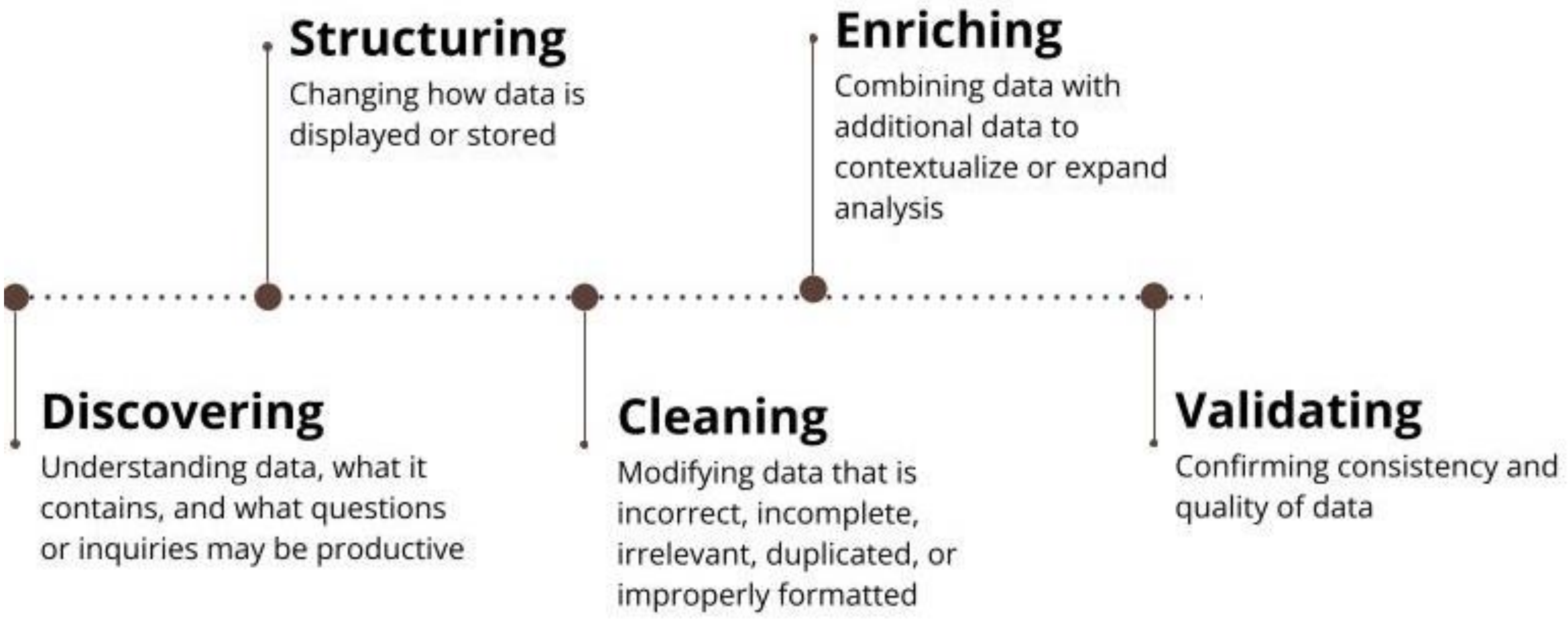


Remember, it all starts with your data

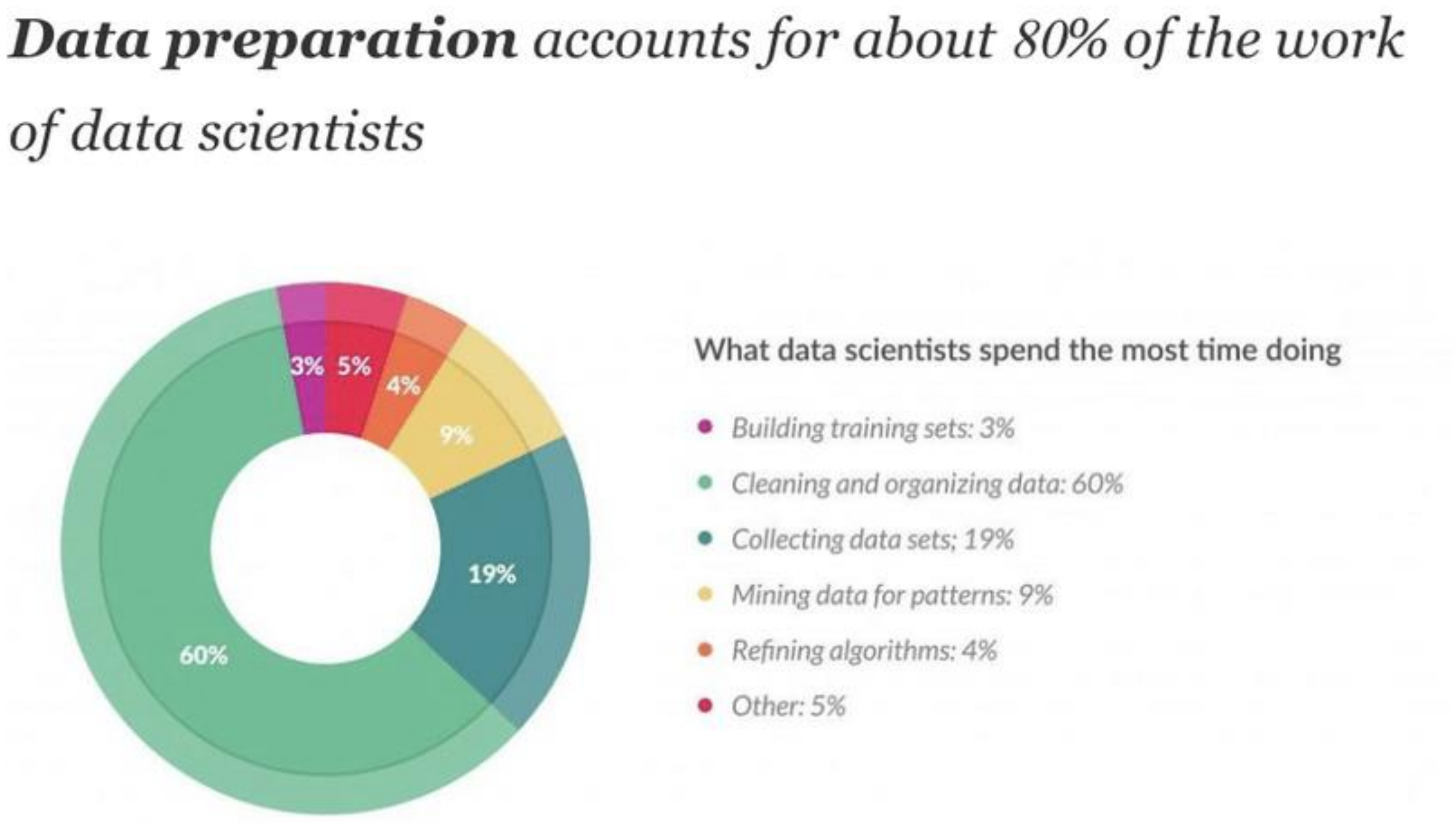
- Collecting “clean” data will
  - Minimize the time you spend wrangling
  - Easier to work with when the variable behave predictably

- Always check your work as you go along
  - Start by writing out your code once
  - Separate out the variables that will be changing
  - Move it into a function

Data Wrangling Workflow

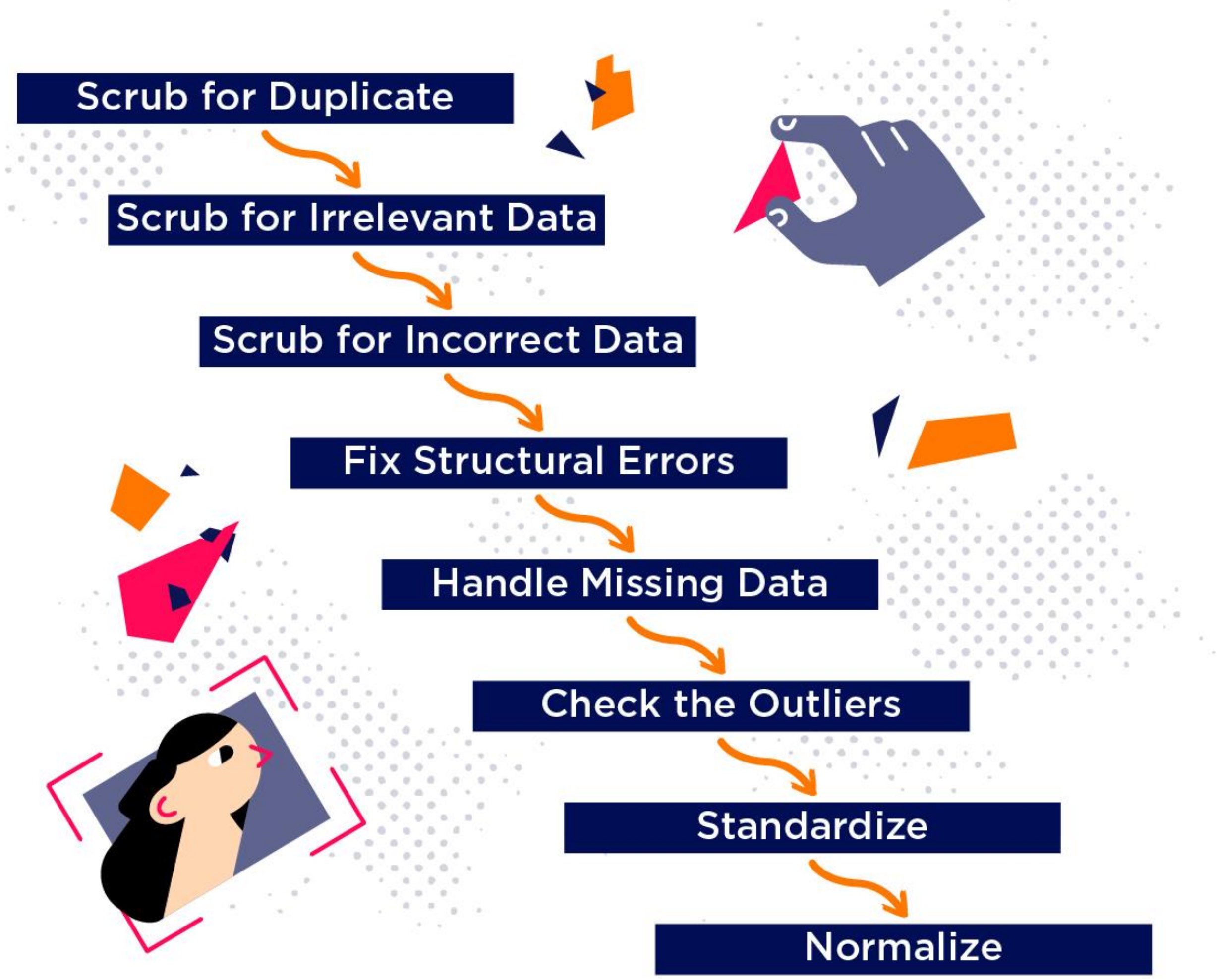


Data wrangling can often be a large component of the total analysis



"Playing the whole game": A data collection and analysis exercise with Google Calendar

What to look for in your dataset



https://www.iteratorshq.com/blog/data-cleaning-in-5-easy-steps/

Data wrangling can be frustrating

- There is no set formula for data wrangling
  - Depends on how the data is collect, what tools you want to use for the analysis

- Often is a very time intensive and iterative process

- Much of the work will not go into the “final product”
  - E.g., often in the supplemental figures rather than main figures of publications

Wrap-up

- Anatomy of a “good” figure
  - Choose the right format for your data, never use 3D plots
- When you need to do something 3 or more times, automate the task
  - Write generalizable code that is easy to edit
  - Custom functions are robust
- Stitch figures together into a panel in R for consistency and ease of formatting
- Garbage in, garbage out – plan your analysis before you start collecting data
  - Data wrangling is reshaping and cleaning the data to prepare it for analysis