Intro to Data Visualisations in R

Taylor Blair

2022-11-10

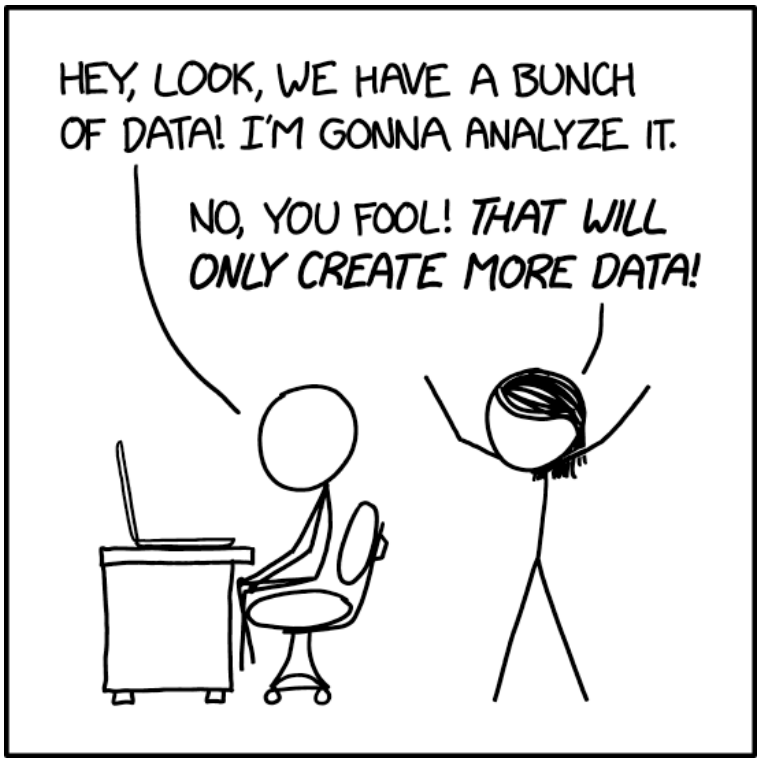
Section 1

Before we get started

A bit about me

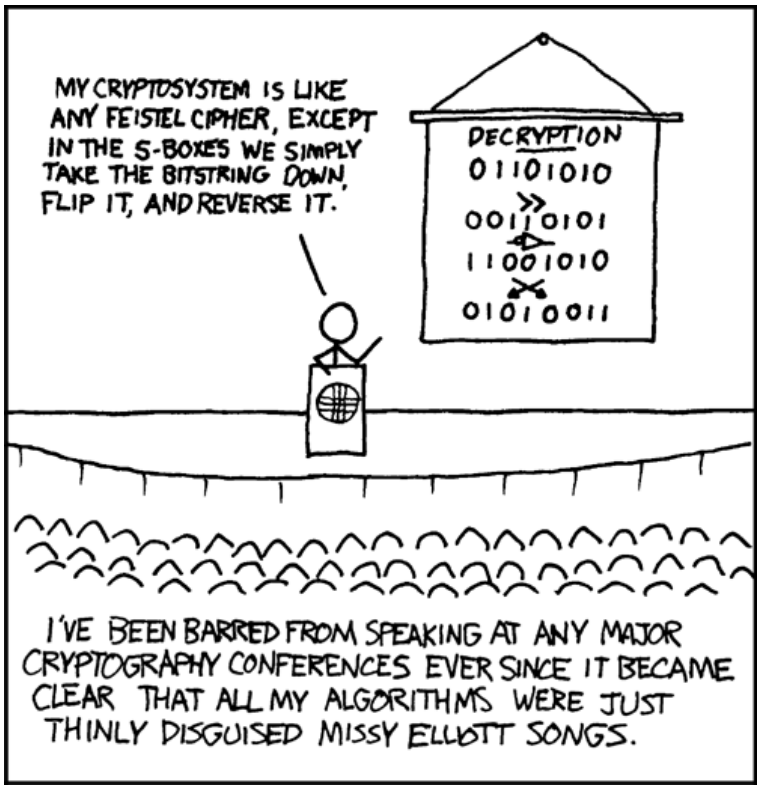
- **Name:** Taylor Blair
- **College I attend:** Reed College
- **Major:** CS/Math
- **Fun Fact:** Sea otters hold hands when they sleep

Who is this workshop for?



- Some programming experience.
- If you already know R then you won't get much out of this.

What will we be doing?



1. Going over a bit about R
2. Setting up RStudio cloud
3. Setting up a work environment
4. Learning to manipulate data
5. Making graphs!

What is R?

- ~~A language that falsely believes that lists start at 1.~~
- A programming language for statisticians, by statisticians.
- Simple language for data science.

Why are we not using Python?

Python is a useful tool for data science, but it has its own use case.

Because Python was not made designed with data science in mind it has a steeper learning curve.

Section 2

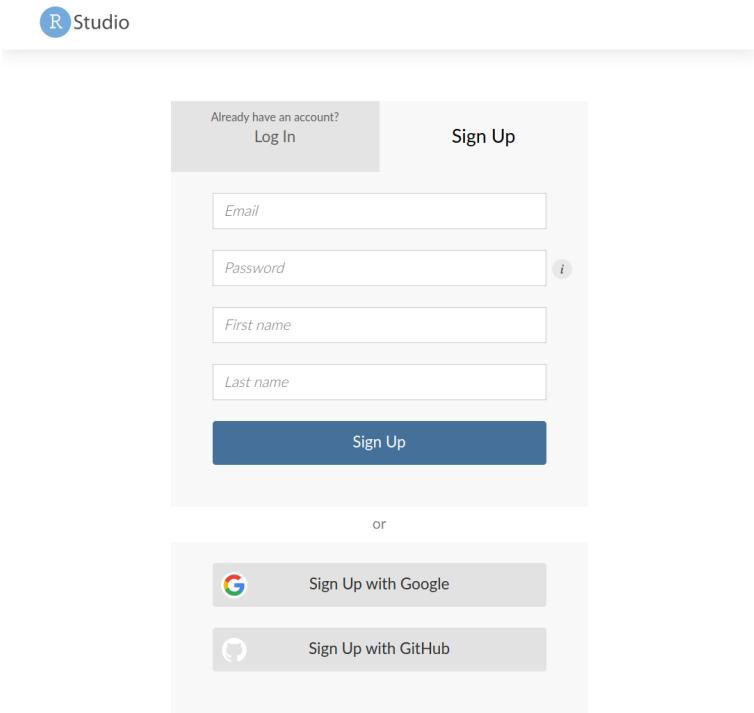
Setup

RStudio

RStudio is a common IDE for R based development.

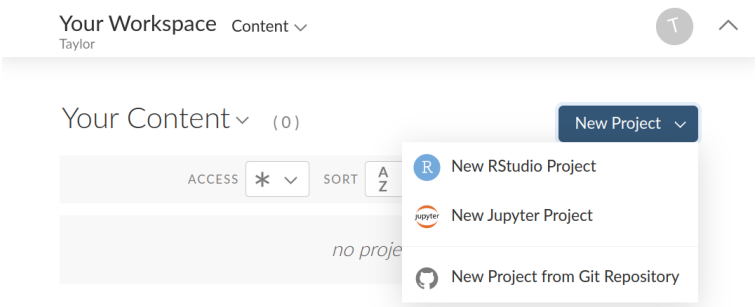
- **Pros**
  - It is laid out for R development.
- **Cons**
  - RStudio features do not translate to other languages.

Making an R Studio Server account



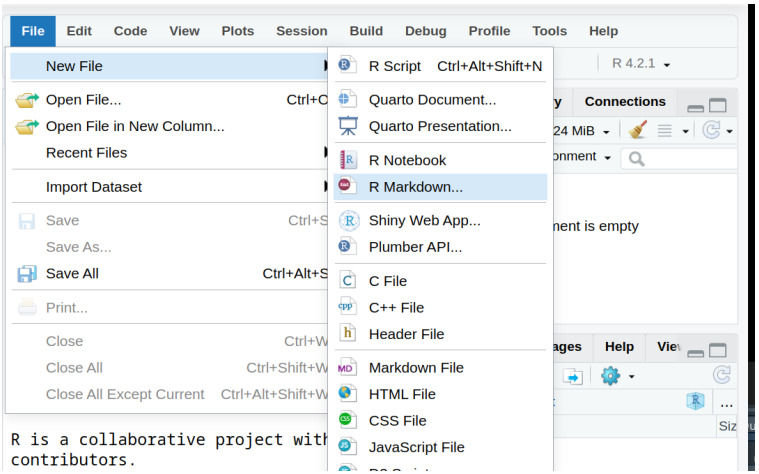
1. Go to: https://rstudio.cloud/plans/free
2. Create an account

Making a New Project



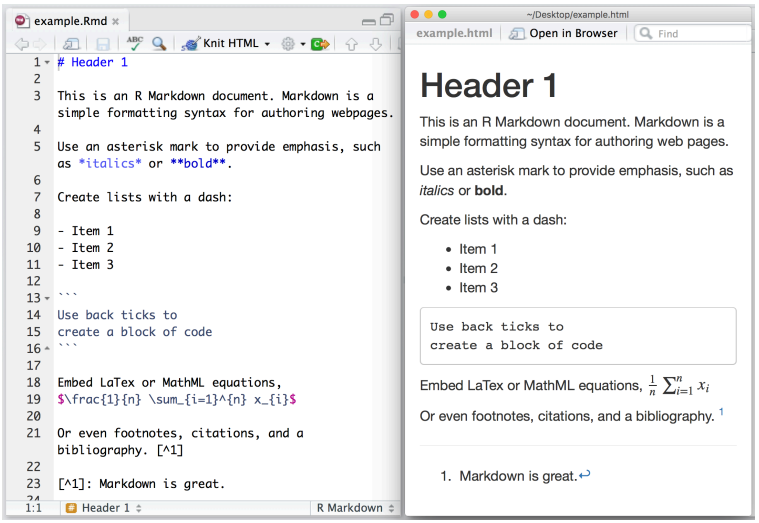
1. Go to "**Your Workspace**"
2. Click "**New Project**"
3. Click "**New RStudio Project**"

Making a new document



1. Go to the file tab
2. Select: "**New File**"
3. Select:"**RMarkdown**"
4. Let it download
5. Select the **HTML** option
6. Knit! (*Yarn ball icon*)

Layout of an RMD (R Markdown)



- YAML header: Specifies how to compile the RMD.
- Markdown text: Standard markdown formatting.
- Code chunks: Denoted by back ticks. Enable code to be run in place and compiled to a document.

tidyverse

tidyverse is a package that combines several common R packages including:

- dplyr (data manipulation)
- ggplot2 (graphing)
- And more!!!

pdxTrees

The pdxTrees package was made by a Reed professor using data from the Portland Parks & Recreation department.

Installing packages

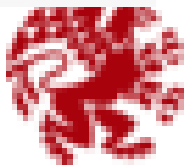
For this workshop we will use two packages: `pdxTrees` and `tidyverse`

1. To install

```
# Only need to run once
# Can be run in the console
install.packages(c("pdxTrees", "tidyverse"))
```

2. To load

```
# Place in a block at the top
library(tidyverse)
library(pdxTrees)
```



Section 3

Manipulating data

A question we can answer with `pdxTrees`

The most famous Reed alum...

- Known for something involving food
- You've probably seen their book
- Last name is a common word
- Had a temper that calmed over time
- Did not graduate from Reed

The most famous Reed alumn is James Beard!

Attended Reed from 1920 to 1922.

Importing the data

To store the `pdxTrees` data in the variable `trees`:

```
# `<-` is the assignment operator
trees <- get_pdxTrees_parks()
```

In practice, `read_csv` or a similar function from the `readr` package will be used.

```
data <- read_csv("big_data.csv")
```

Estimating the age of a tree

Tree age can be estimated using:

$$\text{Diameter at breast height (DBH)} \times \text{growth factor} = \text{Age}$$

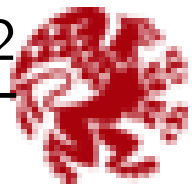
Tree Species	Growth Factor	Tree Species	Growth Factor
Red Maple	4.5	White Oak	5.0
Silver Maple	3.0	Red Oak	4.0
Sugar Maple	5.0	Pin Oak	3.0
River Birch	3.5	Linden or Basswood	3.0
White Birch	5.0	American Elm	4.0
Shagbark Hickory	7.5	Ironwood	7.0
Green Ash	4.0	Cottonwood	2.0
Black Walnut	4.5	Dogwood	7.0
Black Cherry	5.0	Redbud	7.0

Age of trees

Lets make an extra column that represents a rough estimate for tree age:

```
trees <- trees %>% # `%>%` is the pipeline operator
  mutate(growth_factor = 3, # approximate growth factor
         age = growth_factor * DBH,
         planted = 2019 - age) # surveyed between 2017-2019
```

Park	Common_Name	DBH	age	planted
Portsmouth Park	Flowering Plum	12.5	37.5	1981.5
Willamette Park	Pin Oak	31.7	95.1	1923.9
Peninsula Park	Douglas-Fir	32.5	97.5	1921.5
Willamette Park	London Plane Tree	17.3	51.9	1967.1
Kelley Point Park	Black Cottonwood	41.6	124.8	1894.2



Filtering data

Tibbles can be filtered in R using the `filter` function from the `dplyr` package

```
# a vector of parks near reed
parks <- c("Crystal Springs Rhododendron Garden",
           "Woodstock Park",
           "Kenilworth Park")

# Tibble of trees near Reed that from before 1920
james_beard <- trees %>% # `%>%` is the pipeline operator
  filter(Park %in% parks, # Check if tree is in
                          # a park near Reed
         planted < 1920) # Planted before James Beard
```

A few trees James Beard might have seen

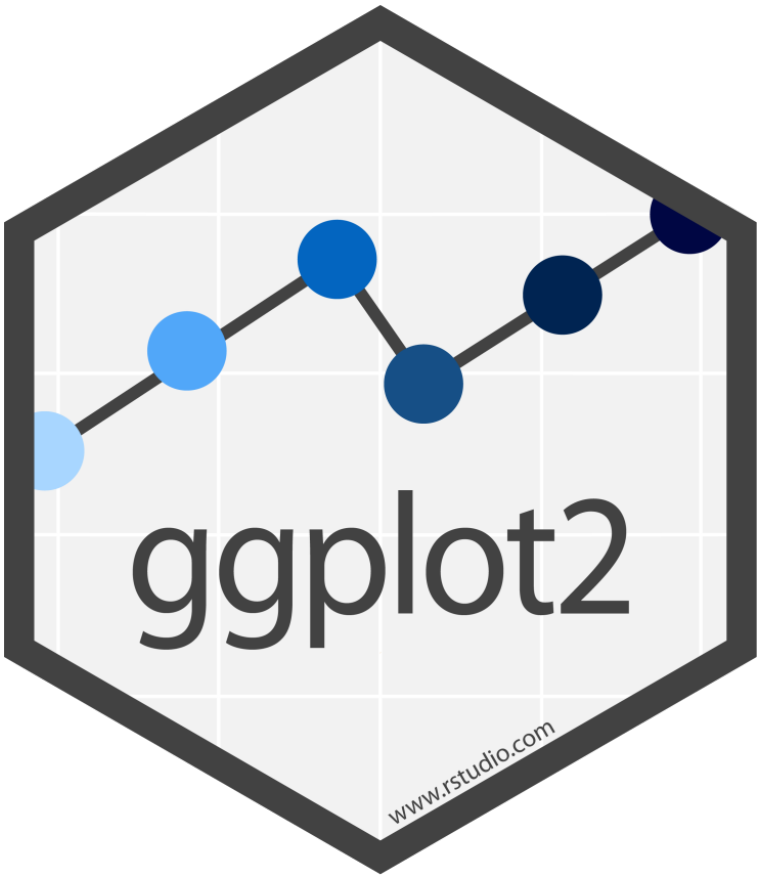
Longitude	Latitude	Common_Name	DBH	age	planted
-122.6142	45.48282	Northern Red Oak	64.7	194.1	1824.9
-122.6308	45.49214	American Sycamore	62.3	186.9	1832.1
-122.6315	45.49127	Douglas-Fir	59.9	179.7	1839.3
-122.6121	45.48323	Douglas-Fir	58.6	175.8	1843.2
-122.6121	45.48355	Douglas-Fir	56.5	169.5	1849.5

Me next to a tree James Beard might have seen

Section 4

Creating Visualizations

ggplot2



ggplot2 is a common graphing library for R. The ‘ggplot‘ function takes an aesthetic mapping and chart type and returns a pretty graph!

aes

aes stands for aesthetic. It takes the mappings for a graph and passes it to the geom layers.

The layout format for aes is:

```
ggplot(data = dataset,
       mapping = aes(x = col_x,
                     y = col_y,
                     color = ...,
                     ...))
```

For the next few graphs we will be using:

```
age_graph <- trees %>%
  ggplot(mapping = aes(x=age))
```

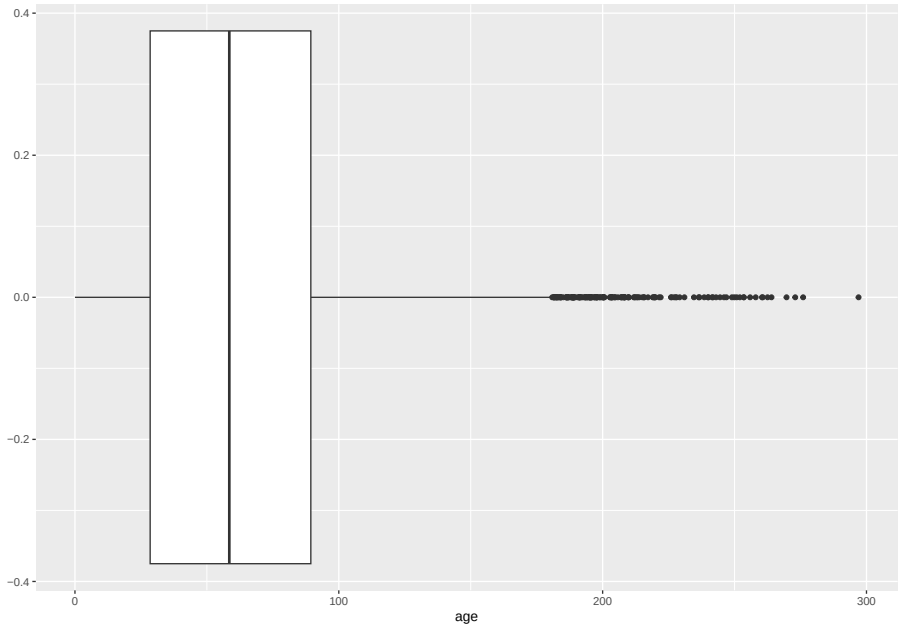
geom

geom specifies the shape for a given graph. For example, fixing an aesthetic and changing the geom to: `geom_hist()`, `geom_density()`, and `geom_boxplot()` results in different types of graphs.

```
# Boxplot
age_graph + geom_boxplot()

# Histogram
age_graph + geom_hist()

# Density graph
age_graph + geom_density()
```

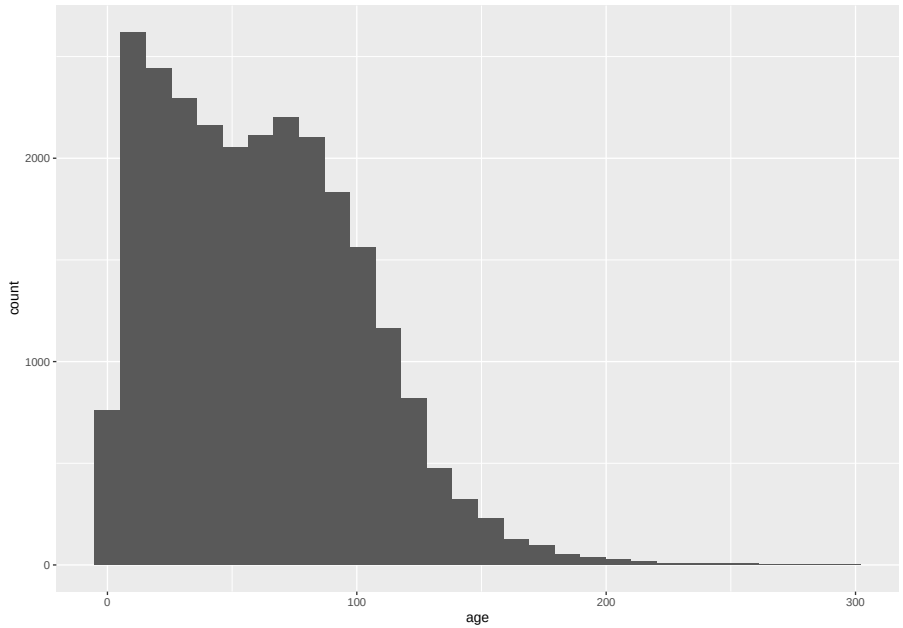
geom_boxplot()

```
# `age_graph` is:
# age_graph <- trees %>%
#   ggplot(mapping = aes(x=age))

age_graph +
  geom_boxplot()
```



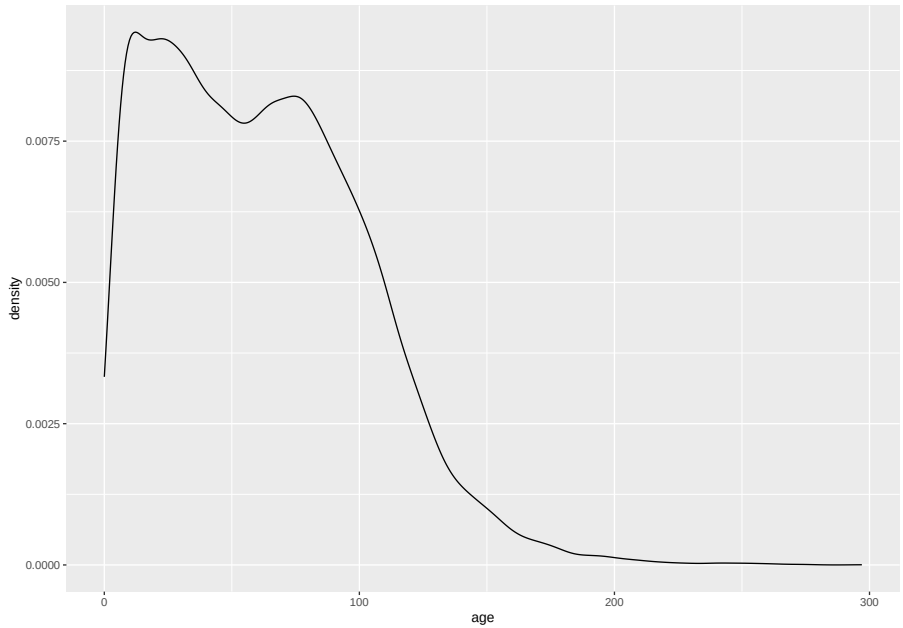
geom_histogram()

```
age_graph +
  geom_histogram()
```

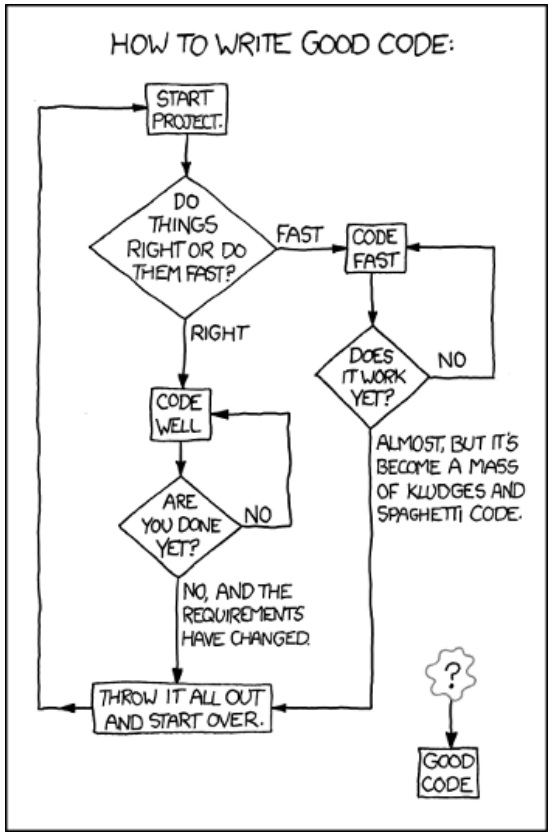


geom_density()

```
age_graph +
  geom_density()
```

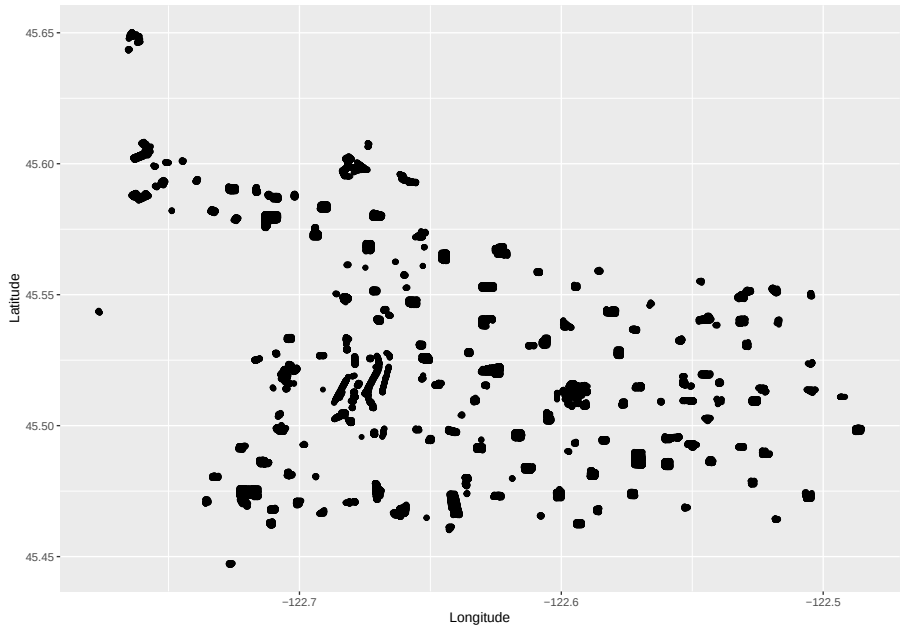


Building out a graph



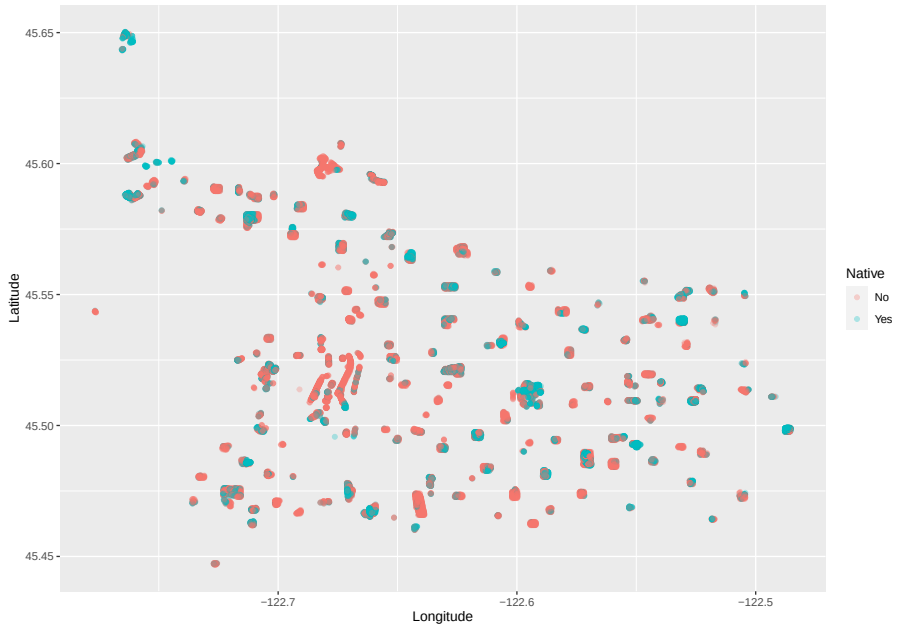
geom_point() of trees in Portland

```
trees %>%
  ggplot(aes(x = Longitude, y = Latitude)) +
  geom_point()
```



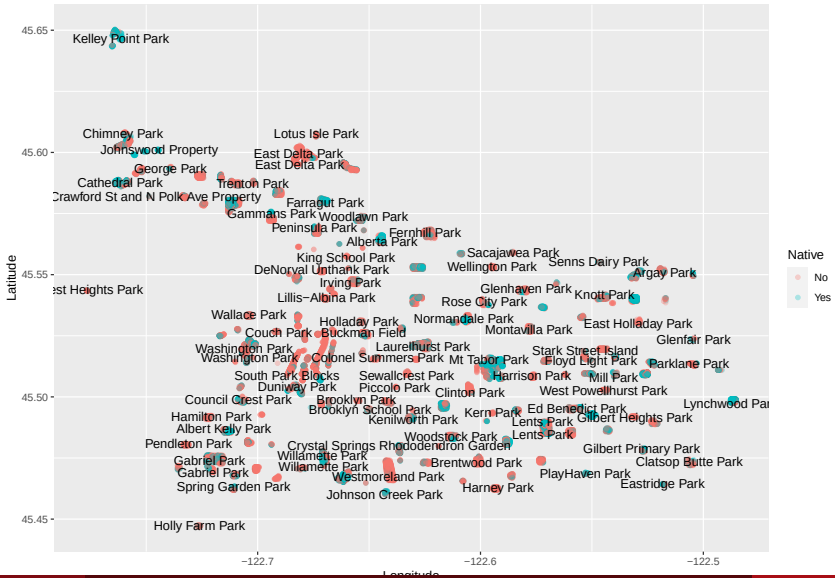
geom_point() of trees in Portland, colored by nativity

```
trees %>%
  drop_na(Native) %>%
  ggplot(aes(x = Longitude, y = Latitude, color = Native)) +
  geom_point(alpha=0.3)
```



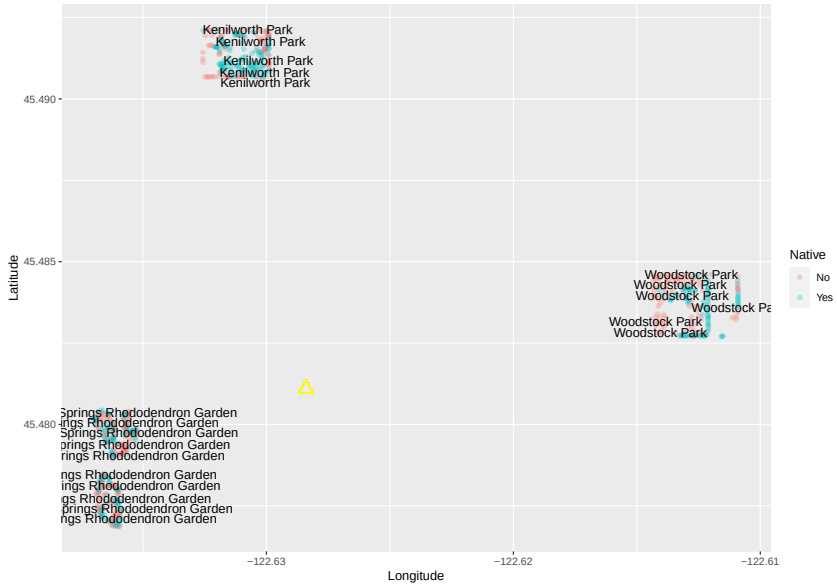
geom_point() of trees in Portland, colored by nativity with park labels

```
trees %>%
  drop_na(Native) %>%
  ggplot(aes(x = Longitude, y = Latitude, color = Native,
             label = Park)) +
  geom_point(alpha=0.3) +
  geom_text(check_overlap = TRUE, colour = "black")
```



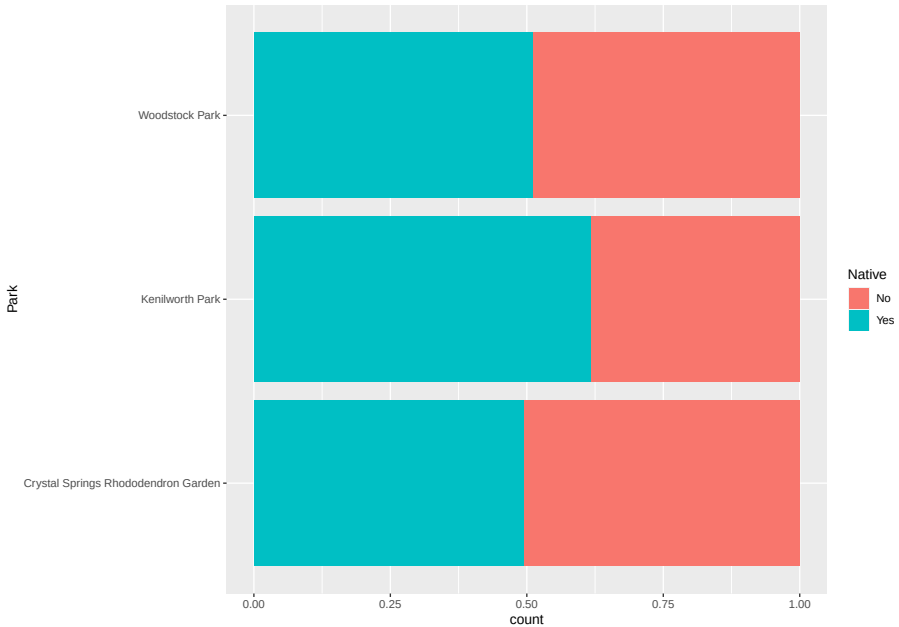
geom_point() of trees near Reed

```
trees %>%
  filter(Park %in% parks) %>%
  drop_na(Native) %>%
  ggplot(aes(x = Longitude, y = Latitude, color = Native, labe
  geom_point(alpha = 0.3) +
  geom_point(x = -122.6284, y = 45.4811, shape = 2, size = 4,
  geom_text(check_overlap = TRUE, colour = "black")
```



geom_bar near Reed

```
trees %>%
  filter(Park %in% parks) %>%
  drop_na(Native) %>%
  ggplot(aes(y=Park, fill = Native)) +
  geom_bar(position = "fill")
```



Go try things!

Make a chart using `pdxTrees` and `ggplot` provided. Here are some commands to get you started:

General `R`	`dplyr`	`ggplot2`
Assignment: `<-`	`filter()`	`ggplot(aes()) +`
Pipeline: `%>%`	`mutate()`	`aes(x =..., y=...)`
Comparison: `%in%`, `>`	`drop_na()`	`geom_bar()`, `geom_point`

Section 5

Closing

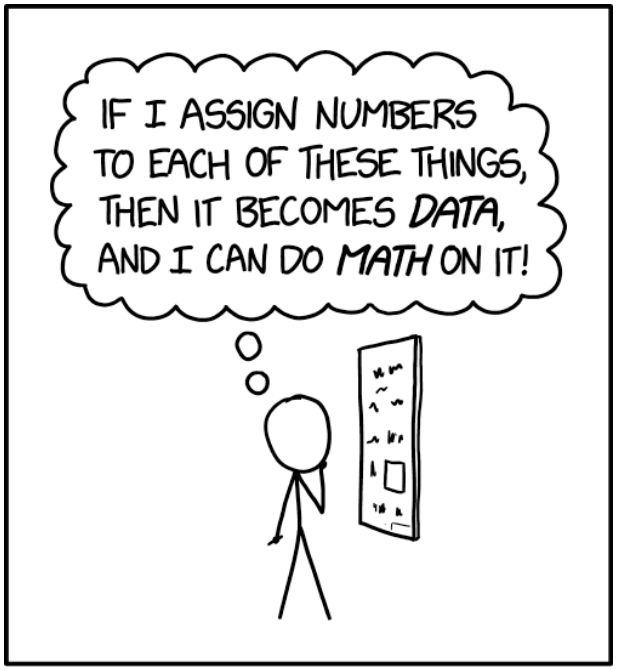
If you want to continue

Center for
Data Science
College of Information & Computer Sciences

Special Thanks

- Becca Elenzil
- Charles McGuffey
- Christine Miller
- Evan Sieden
- Joyce Levine
- Kelly McConville
- Nate Wells
- Workshop Staff

That's all!


IF I ASSIGN NUMBERS TO EACH OF THESE THINGS, THEN IT BECOMES DATA, AND I CAN DO MATH ON IT!
THE SAME BASIC IDEA UNDERLIES GÖDEL'S INCOMPLETENESS THEOREM AND ALL BAD DATA SCIENCE.