One Data Science Programme Week 2

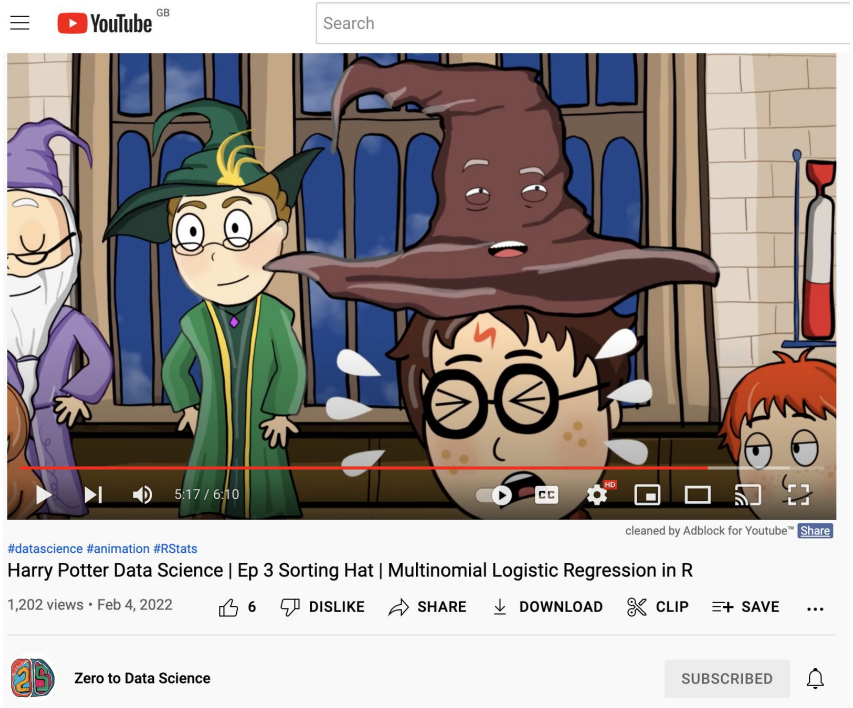
Introduction to Data Wrangling and Data Visualisation

Kai Xiang Lim

Outline

- Recap: dataset
- Introduction to data wrangling: Definition, piping, and examples
- Introduction to data visualisation: using the ggplot2 package

Dataset



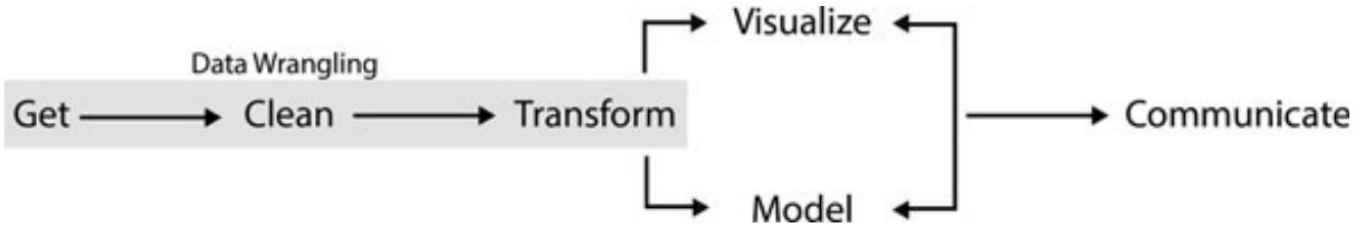
Hogwarts_enrolment_data.csv

List of variables:

- House
- Name
- Birthday
- Best hand
- Arithmancy
- Muggle Studies
- Defence Against the Dark Arts
- ……

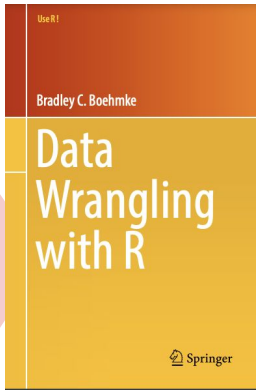
https://raw.githubusercontent.com/kai-lim/One-Data-Science/main/data/Hogwarts_enrolment_data.csv

Data wrangling



"...the art of using computer programming to **extract raw data and creating clear and actionable bits of information for your analysis**."

"...the ability to take a messy, unrefined source of data and **wrangle it into something useful**."



(Boehmke, 2016)

Data wrangling

- Data transformation - calculating means (transform raw data into mean)
  - What is the mean of Defence Against the Dark Arts?

```
data <- read.csv("Hogwarts_enrolment_data.csv")
```

```
Dark.Arts <- data$Defense.Against.the.Dark.Arts
head(Dark.Arts, 10) #see the first 10 rows/data points
```

```
##  [1] -6.889120 -4.536762 -5.440189 -3.675312 -3.542801 -5.999016  4.261754
##  [8] -3.769207  5.077157  5.695134
```

```
mean(Dark.Arts, na.rm = TRUE) #na.rm means removing NA (aka missing data)
```

```
## [1] -0.3878635
```

Piping %>%

https://www.reddit.com/r/rstats/comments/vbd6jq/piping_in_r_is_like_baking/

Use %>% to calculate mean

```
library(magrittr) #this is needed for the "%>%" function
library(dplyr) # this is needed for functions such as select() and s
ummarise()

#Select Defence Against the Dark Arts and view the first 10
data %>%
  select(Defense.Against.the.Dark.Arts) %>%
  head(10)
```

```
##    Defense.Against.the.Dark.Arts
## 1                      -6.889120
## 2                      -4.536762
## 3                      -5.440189
## 4                      -3.675312
## 5                      -3.542801
## 6                      -5.999016
## 7                       4.261754
## 8                      -3.769207
## 9                       5.077157
## 10                      5.695134
```

The select() function literally selects one or multiple columns from the dataset

```
#Select Defence Against the Dark Arts and calculate its mean
data %>%
  select(Defense.Against.the.Dark.Arts) %>%
  summarise(Dark.Art.Mean = mean(Defense.Against.the.Dark.Arts))
```

```
##   Dark.Art.Mean
## 1    -0.3878635
```

The summarise() function provides a summary for the data, but you need to tell it what to do, such as mean() to calculate mean for this example

```
#Select Defence Against the Dark Arts and calculate its mean, whilst
 showing number of students
data %>%
  select(Defense.Against.the.Dark.Arts) %>%
  summarise(n = n(),
            Dark.Art.Mean = mean(Defense.Against.the.Dark.Arts))
```

Here, we add n() so that summarise() knows it needs to provide an output of count in addition to mean.

```
##      n Dark.Art.Mean
## 1 1600    -0.3878635
```

```r
#Add the `group_by()` function to further subset
 can calculate means for different houses
data %>%
  group_by(Hogwarts.House) %>%
  select(Defense.Against.the.Dark.Arts) %>%
  summarise(n = n(),
            Dark.Art.Mean = mean(Defense.Against.the.Dark.Arts))
```

'Group_by' adds grouping information. Together with summarise() it can compute separate summary for each group!

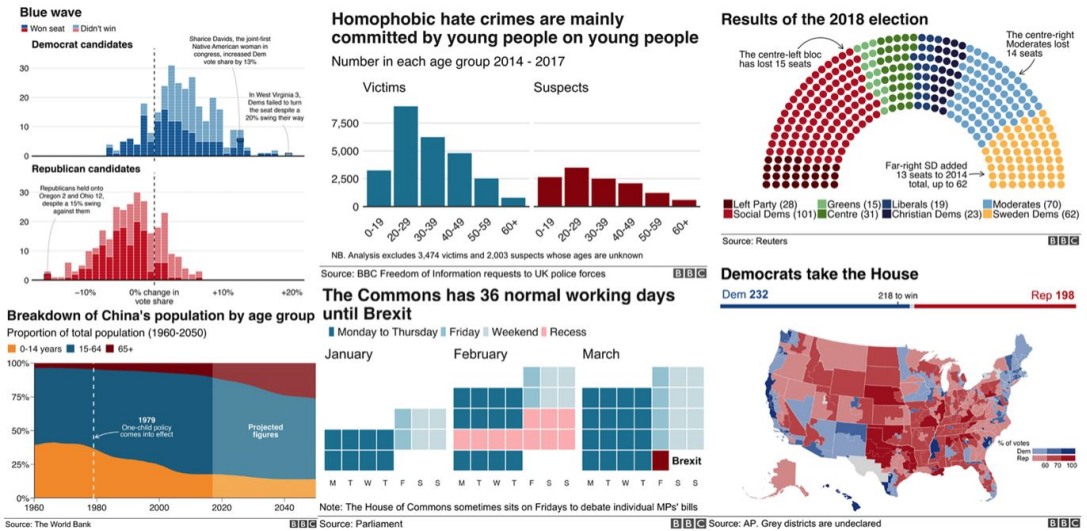
```
## # A tibble: 4 × 3
##   Hogwarts.House     n Dark.Art.Mean
##   <fct>          <int>         <dbl>
## 1 Gryffindor       327         -4.86
## 2 Hufflepuff       529         -4.89
## 3 Ravenclaw        443          4.72
## 4 Slytherin        301          4.86
```

Data visualisation

How the BBC Visual and Data Journalism team works with graphics in R

BBC Visual and Data Journalism Follow

Feb 1 · 8 min read



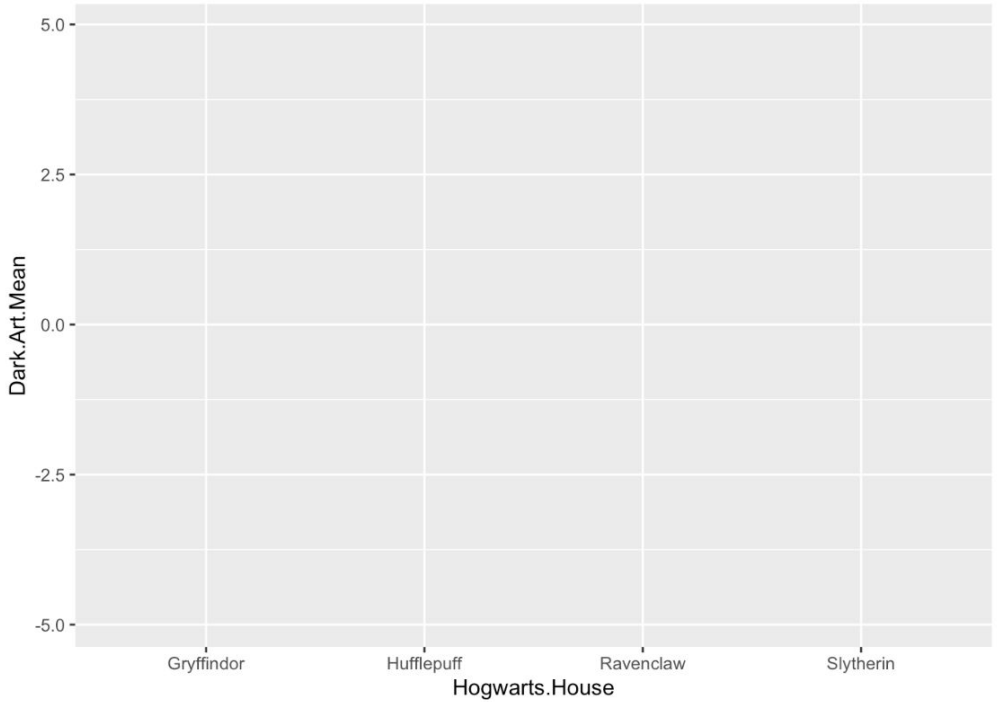
https://bookdown.org/animestina/bookdown-demo/week-4.html

```r
# Save the output as a new object - data.for.plotting
  data %>%
  group_by(Hogwarts.House) %>%
  select(Defense.Against.the.Dark.Arts) %>%
  summarise(n = n(),
            Dark.Art.Mean = mean(Defense.Against.the.Dark.Arts)) ->
  data.for.plotting

# show data.for.plotting
data.for.plotting
```

```
## # A tibble: 4 × 3
##   Hogwarts.House     n Dark.Art.Mean
##   <fct>          <int>         <dbl>
## 1 Gryffindor       327         -4.86
## 2 Hufflepuff       529         -4.89
## 3 Ravenclaw        443          4.72
## 4 Slytherin        301          4.86
```

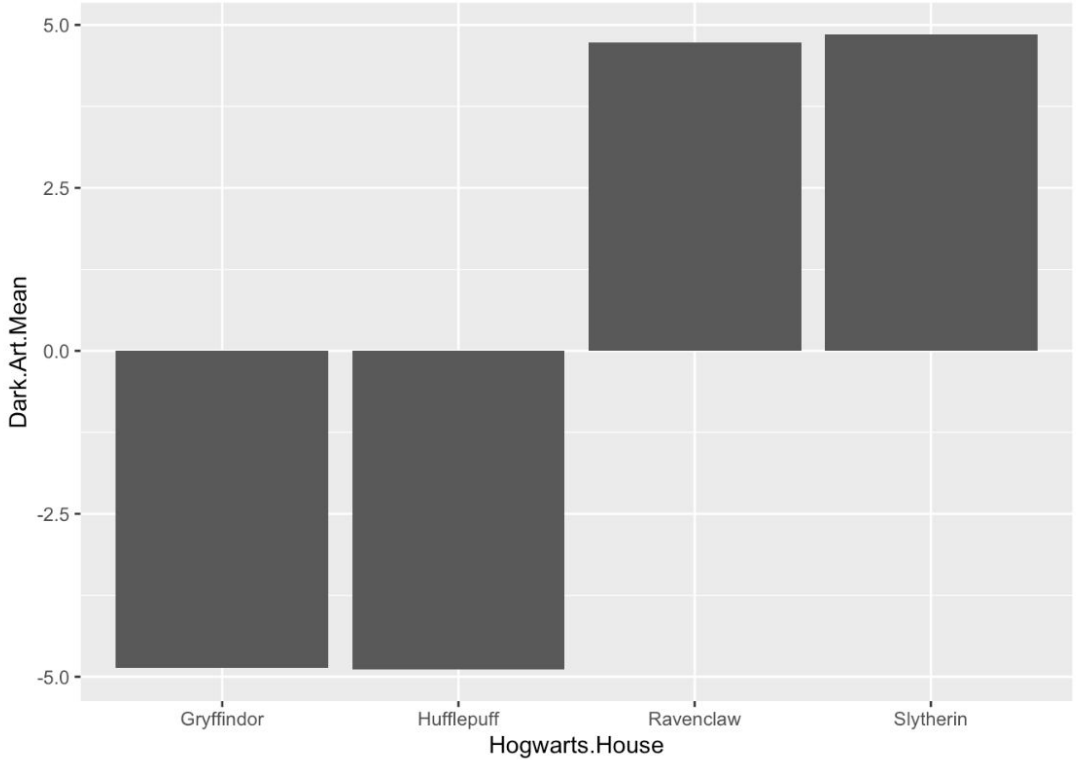
Let's plot a bar chart using this table!

```r
# load another library called ggplot2, which is used for data visual
isation.
library(ggplot2)

# This will only show an empty plot with the x and y axes.
data.for.plotting %>%
  ggplot(aes(x = Hogwarts.House, y = Dark.Art.Mean))
```

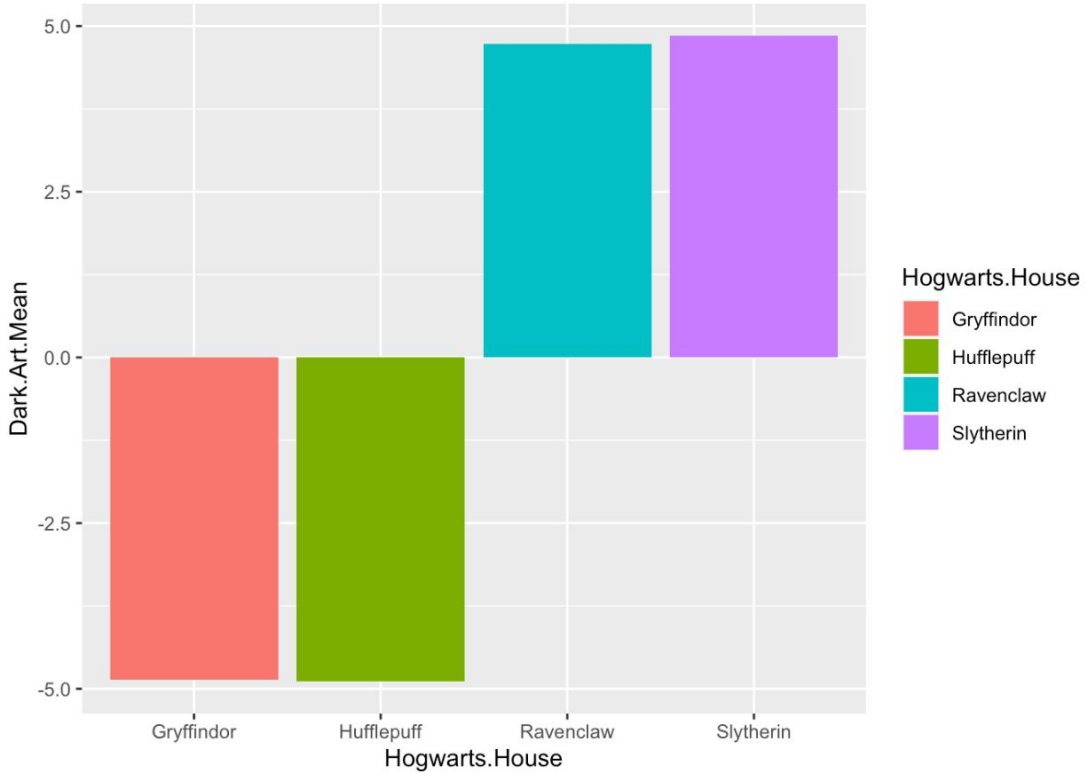


Only x and y axes plotted, where are the bars?

```r
# To add the barplots, we need to add the geom_bar() function.
data.for.plotting %>%
  ggplot(aes(x = Hogwarts.House, y = Dark.Art.Mean)) +
  geom_bar(stat = "identity")
```

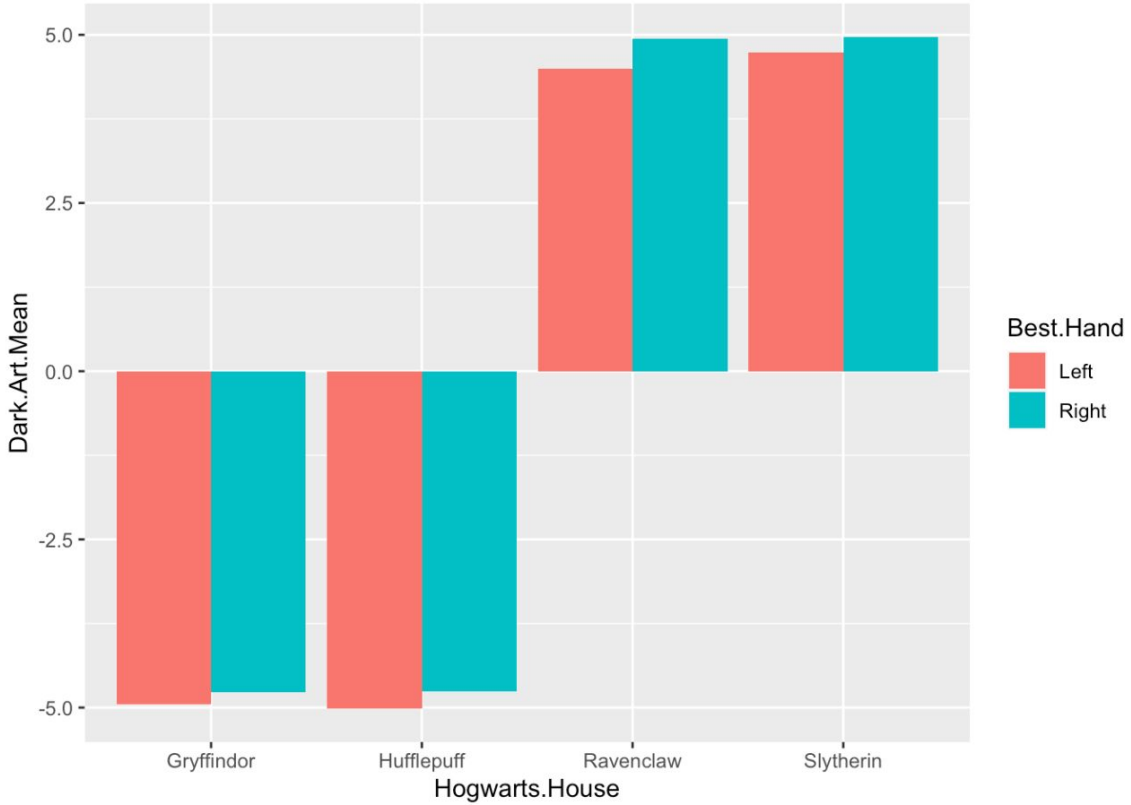


```
# Add fill colour by using 'fill=Hogwarts.House' such that each House has a d
ifferent colour.
data.for.plotting %>%
  ggplot(aes(x = Hogwarts.House, y = Dark.Art.Mean, fill = Hogwarts.House)) +
  geom_bar(stat = "identity")
```



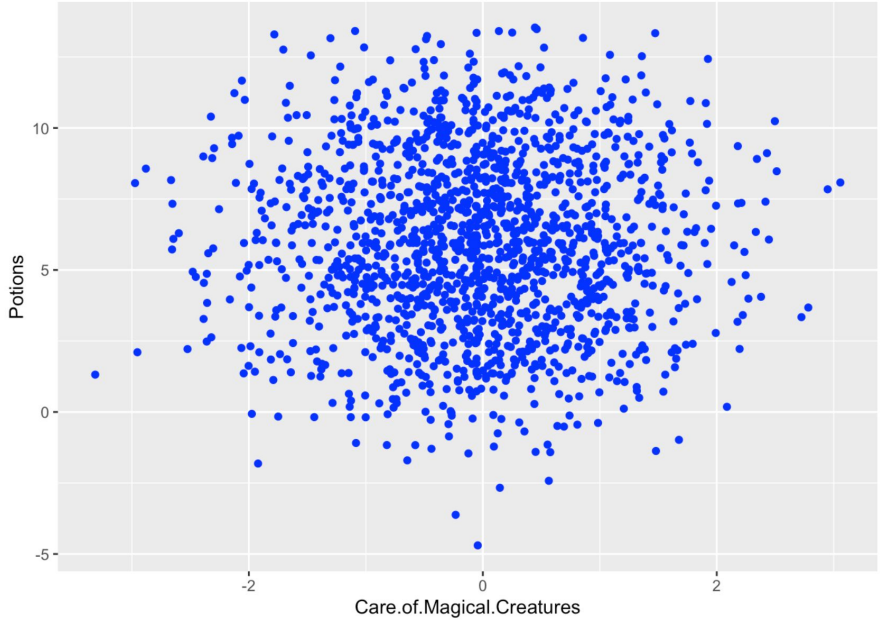
```
# We add another "group_by" variable: Best.Hand together with Hogwar
ds.House. This will give us means for 8 groups (4 Houses and within
 each house there are two means for the best hand, one for left and
 one for right)
  data %>%
  group_by(Hogwarts.House, Best.Hand) %>%
  select(Defense.Against.the.Dark.Arts) %>%
  summarise(n = n(),
            Dark.Art.Mean = mean(Defense.Against.the.Dark.Arts)) ->
  data.for.plotting.2
```

```r
# Plot the second graph
data.for.plotting.2 %>%
  ggplot(aes(x = Hogwarts.House, y = Dark.Art.Mean, fill = Best.Hand)) +
  geom_bar(stat = "identity", position = position_dodge())
```

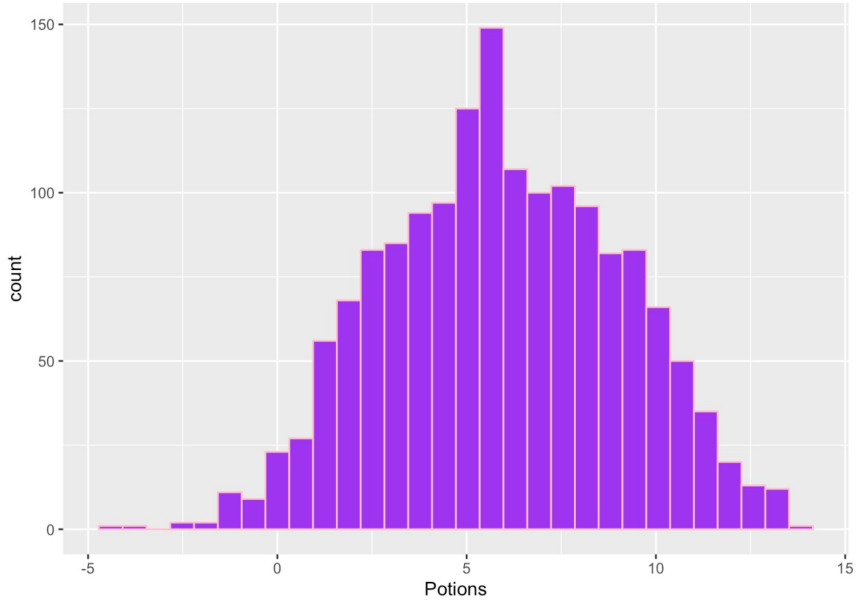


There are many other ways to visualise data

```
ggplot(data,aes(x=Care.of.Magical.Creatures,y=Potions))+
  geom_point(color="blue")
```




```
ggplot(data,aes(x=Potions))+
  geom_histogram(fill="purple", color="pink")
```

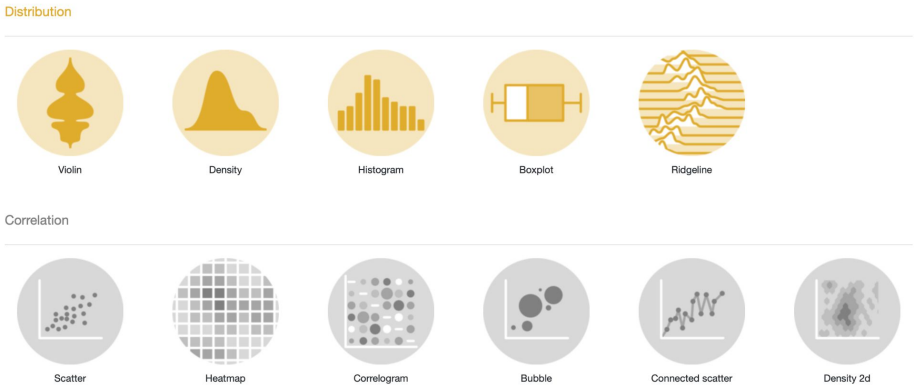


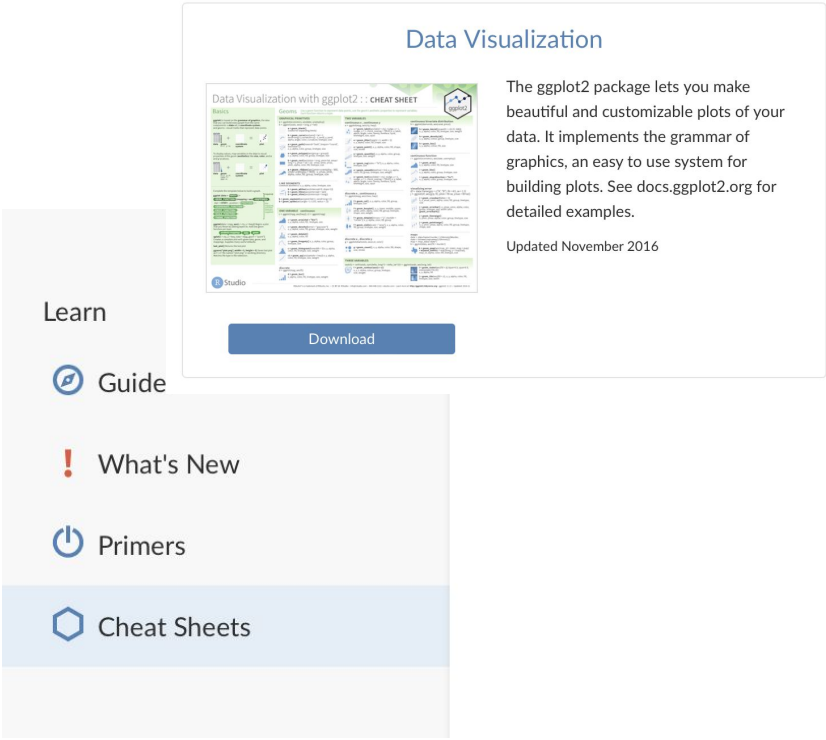
```
long_data <- tidyr::pivot_longer(data, cols=colnames(data)[7:18], names_to="subject",values_to="score")

ggplot(long_data,aes(x=score,fill=subject))+
  geom_histogram()+
  facet_wrap(~subject, scale="free_x")+
  theme(legend.position = "none")
```

Useful resources

Ggplot2 cheatsheet

https://r-graph-gallery.com





Send us an email if you have any questions:

one.data.science.program@gmail.com