Class 2: Introduction to ggplot2

Andrew Parnell
andrew.parnell@mu.ie

PRESS RECORD
https://andrewcparnell.github.io/dataviz_course

Learning outcomes

- Learn the basics of how to use `ggplot2`
- Be able to add simple features to existing ggplots
- Be able to perform basic customisation of ggplots

The philosophy behind the grammar of graphics

- The gg in `ggplot2` stands for grammar of graphics. The idea is that to build a graphic we need a good grammar, just like we need grammar to write sentences
- The graphical grammar is built from geometric objects, scales and a coordinate system which are layered on top of each other
- Extra layers can be added which might alter the scales, split the plot into multiple panels, change colours, etc
- `ggplot2` requires just a few of these components to be specified, and then cleverly works out from your data what appropriate values should be used to create the plot

Reminder: the penguin data

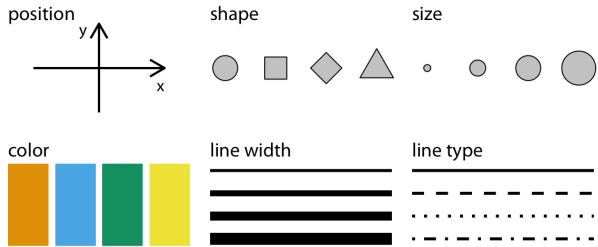
```
library(palmerpenguins)
penguins %>% glimpse
```

```
## Rows: 344
## Columns: 8
## $ species           <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adeli
## $ island            <fct> Torgersen, Torgersen, Torgersen, Torgersen, To
## $ bill_length_mm    <dbl> 39.1, 39.5, 40.3, NA, 36.7, 39.3, 38.9, 39.2,
## $ bill_depth_mm     <dbl> 18.7, 17.4, 18.0, NA, 19.3, 20.6, 17.8, 19.6,
## $ flipper_length_mm <int> 181, 186, 195, NA, 193, 190, 181, 195, 193, 19
## $ body_mass_g       <int> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675,
## $ sex               <fct> male, female, female, NA, female, male, femal
## $ year              <int> 2007, 2007, 2007, 2007, 2007, 2007, 2007, 200
```

Data formats, aesthetics, geoms

Every ggplot has:

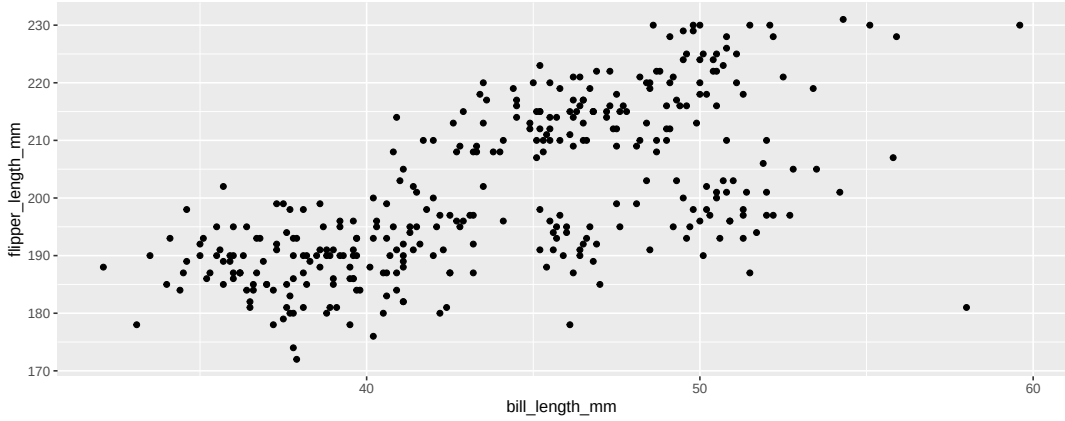
- A data set (usually a data frame or a tibble)
- An **aes(thetic)** which maps the data to graphical elements
- A **geom(etry)** which types of graphical elements to display



(From Fundamentals of Data Visualisation)

An example ggplot

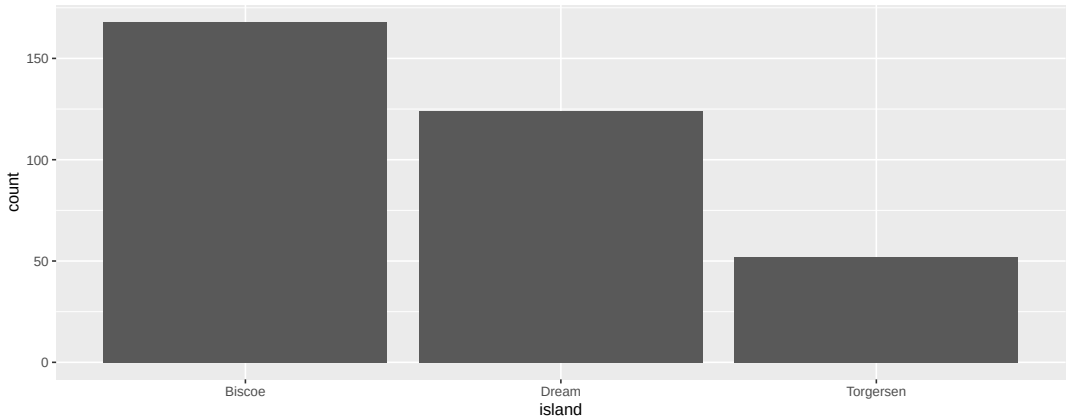
```
ggplot(data = penguins,
       aes(x = bill_length_mm, y = flipper_length_mm)) +
  geom_point()
```



(Why is this better than the plot from class 1?)

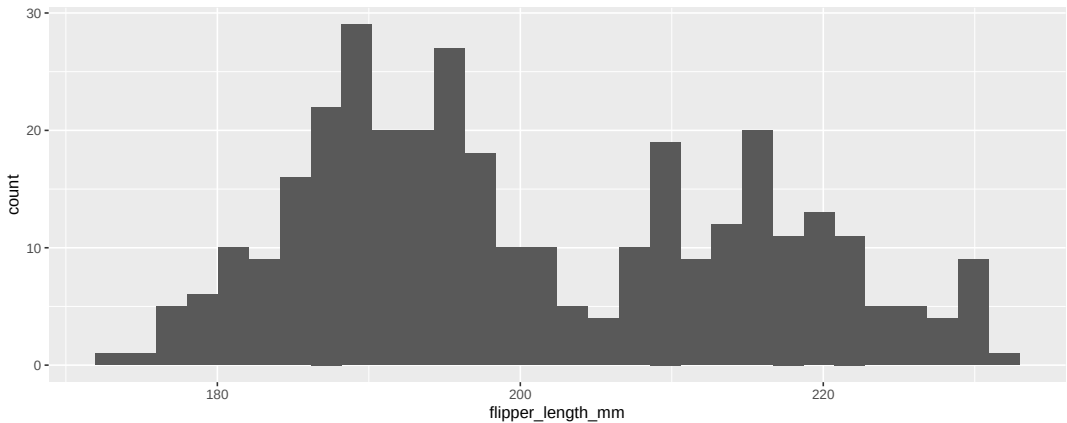
Some basic plot types: 1 Bar charts

```
ggplot(penguins, aes(x = island)) +
  geom_bar()
```



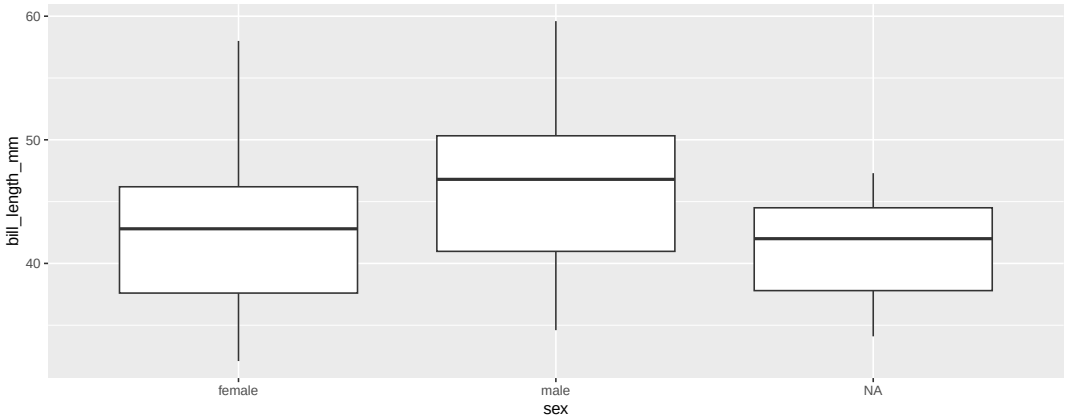
Some basic plot types: 2 Histograms

```
ggplot(penguins, aes(x = flipper_length_mm)) +
  geom_histogram(bins = 30)
```



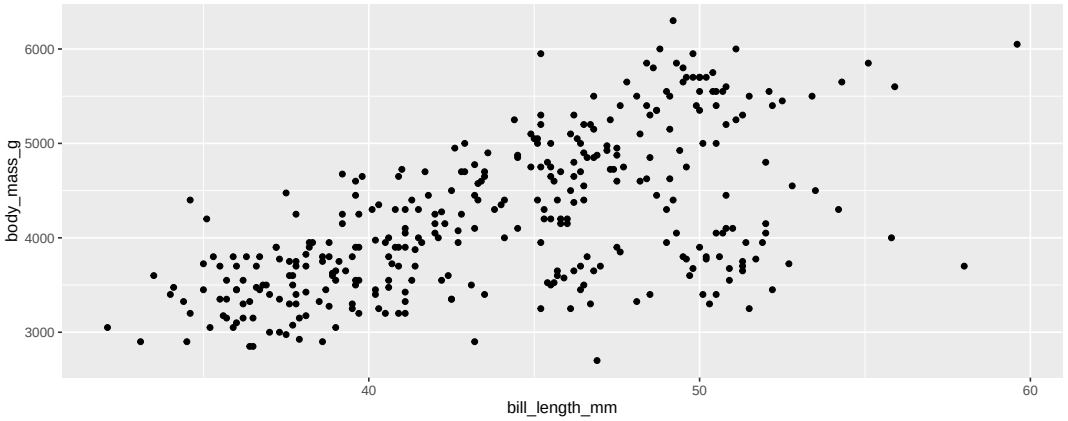
Some basic plot types: 3 Boxplots

```
ggplot(penguins, aes(x = sex, y = bill_length_mm)) +
  geom_boxplot()
```



Some basic plot types: 4 Scatter plots

```
ggplot(data = penguins,
       aes(x = bill_length_mm, y = body_mass_g)) +
  geom_point()
```



Basic plot customisation and themes

Labels and titles

Use `labs`:

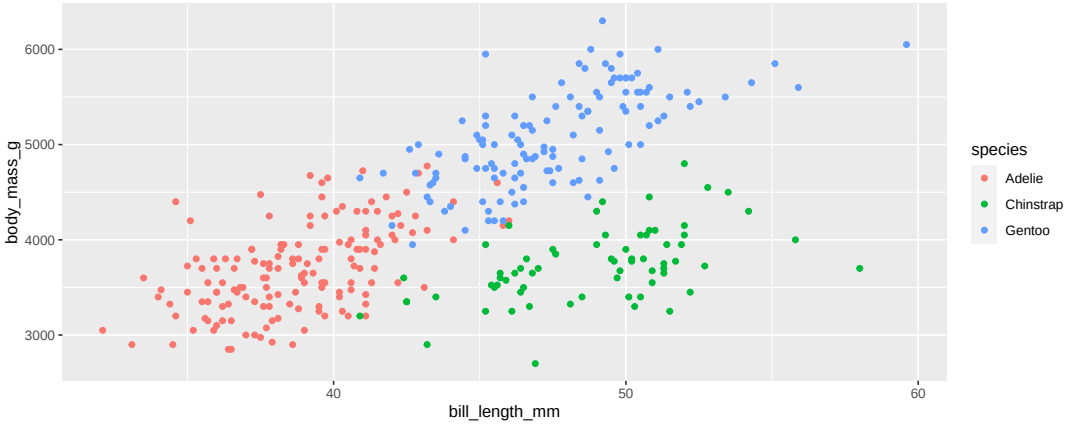
```
ggplot(data = penguins,
       aes(x = bill_length_mm, y = body_mass_g)) +
  geom_point() +
  labs(
    title = "Bill length vs body mass",
    subtitle = "Penguin data set",
    x = "Bill length (mm)",
    y = "Body mass (g)",
  )
```

or you can specify these individually with, e.g. `+ xlab("Bill length")`

Adding colour

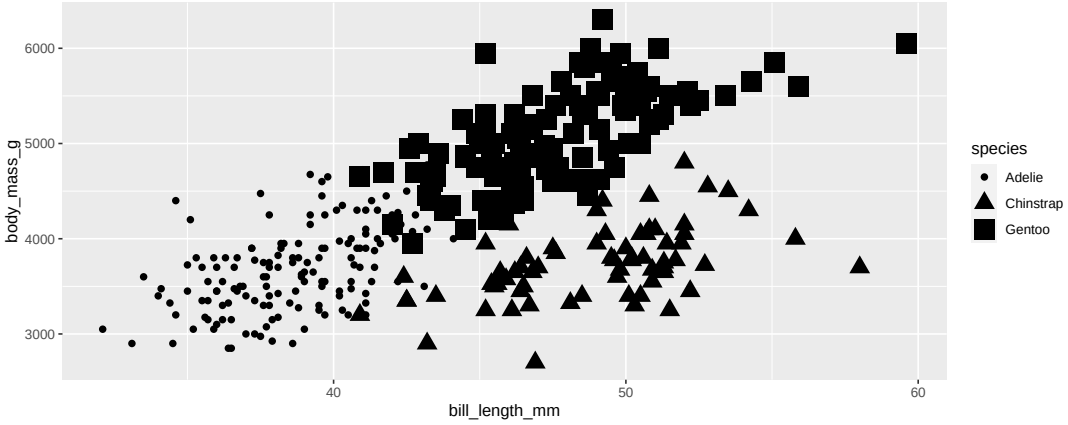
Add colour as another aesthetic:

```
ggplot(data = penguins,
       aes(x = bill_length_mm, y = body_mass_g, colour = species)) +
  geom_point()
```



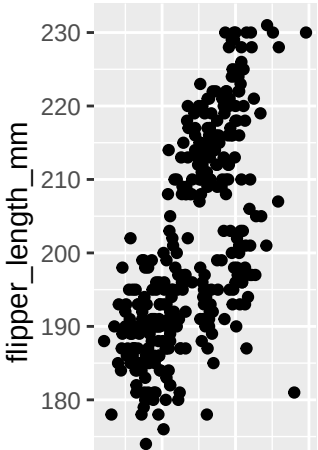
Changing point types

```
ggplot(data = penguins,
       aes(x = bill_length_mm, y = body_mass_g,
           shape = species, size = species)) +
  geom_point()
```



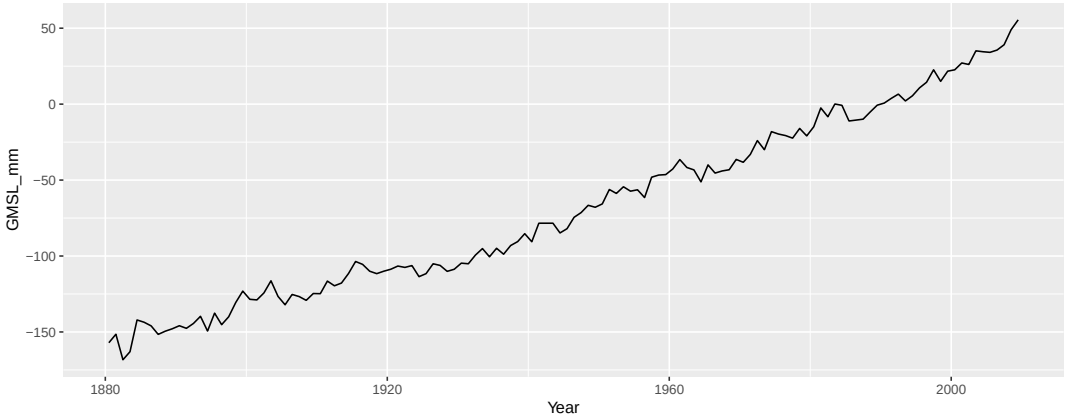
Changing the coordinates

```
ggplot(data = penguins,
       aes(x = bill_length_mm, y = flipper_length_mm)) +
  geom_point() +
  coord_fixed()
```



Adding lines

```
sl <- read.csv('../data/sea_level.csv')
ggplot(sl, aes(x = Year, y = GMSL_mm)) +
  geom_line()
```



Some notes

- The aesthetic can go inside the `ggplot` or inside the `geom`
- If you put it inside the `ggplot` function it will persist across later layers
- By contrast if you put it inside the geom it only works for that layer

ggplots as objects

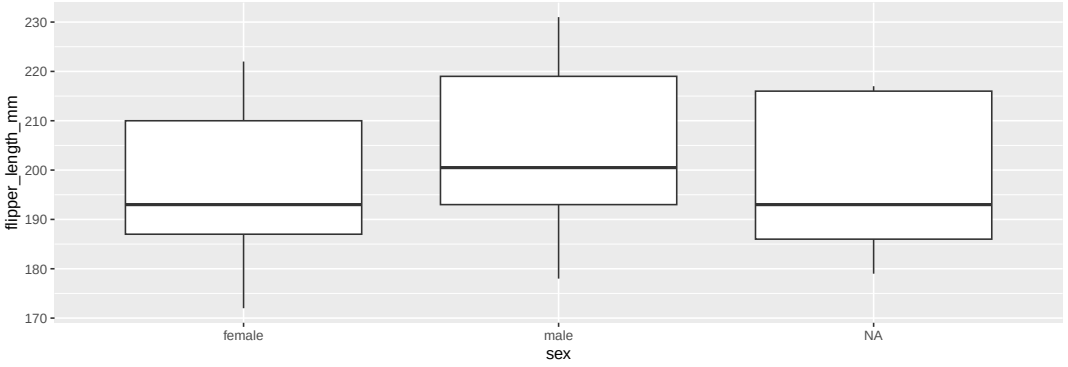
- You can save a ggplot as an object and then repeatedly update it:

```
p <- ggplot(data = penguins,
       aes(x = bill_length_mm, y = flipper_length_mm))
p + geom_point()
p + geom_line()
```

Exercise

Every body spend 5 minutes taking this plot command and adding basic customisations to it. Post your better versions to Slack

```
ggplot(data = penguins,
       aes(x = sex, y = flipper_length_mm)) +
  geom_boxplot()
```



Summary

- Most of the hard work with `ggplot2` is getting your data into the right format (see the next practical)
- Then it's a question of adding the right layers to get the plot you want. More layers discussed in next section
- Always go back to class 1 to check that you have satisfied the golden rules!