Introduction to Data Science

Data and Data Visualization

Nith Kosal

Future Forum

Future Forum, April 9, 2022

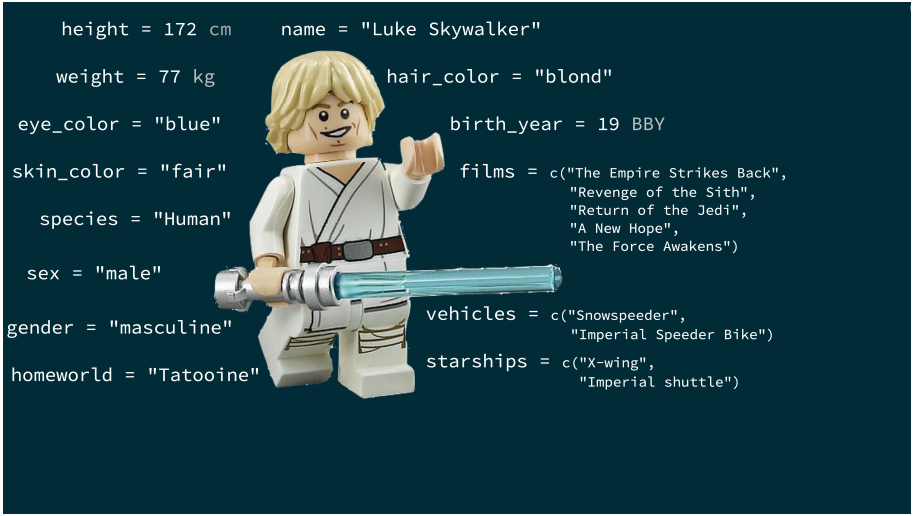
Dataset Terminology

- Each row is an **observation**
- Each column is a **variablee**, with an emphasis on statistical thinking.

```
starwars
```

```
## # A tibble: 87 × 14
##   name    height  mass hair_color skin_color eye_color birth_year
##   <chr>    <int> <dbl> <chr>      <chr>      <chr>           <dbl>
## 1 Luke S…    172    77 blond      fair       blue             19
## 2 C-3PO      167    75 <NA>       gold       yellow          112
## 3 R2-D2       96    32 <NA>       white, bl… red              33
## 4 Darth …    202   136 none       white      yellow           41.9
## 5 Leia O…    150    49 brown      light      brown            19
## 6 Owen L…    178   120 brown, gr… light      blue             52
## # … with 81 more rows, and 7 more variables: sex <chr>,
## #   gender <chr>, homeworld <chr>, species <chr>, films <list>,
## #   vehicles <list>, starships <list>
```

Luke Skywalker



What's in the Star Wars Data?

```
glimpse(starwars)
```

```
## Rows: 87
## Columns: 14
## $ name       <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth V…
## $ height     <int> 172, 167, 96, 202, 150, 178, 165, 97, 183, 1…
## $ mass       <dbl> 77.0, 75.0, 32.0, 136.0, 49.0, 120.0, 75.0, …
## $ hair_color <chr> "blond", NA, NA, "none", "brown", "brown, gr…
## $ skin_color <chr> "fair", "gold", "white, blue", "white", "lig…
## $ eye_color  <chr> "blue", "yellow", "red", "yellow", "brown", …
## $ birth_year <dbl> 19.0, 112.0, 33.0, 41.9, 19.0, 52.0, 47.0, N…
## $ sex        <chr> "male", "none", "none", "male", "female", "m…
## $ gender     <chr> "masculine", "masculine", "masculine", "masc…
## $ homeworld  <chr> "Tatooine", "Tatooine", "Naboo", "Tatooine",…
## $ species    <chr> "Human", "Droid", "Droid", "Human", "Human",…
## $ films      <list> <"The Empire Strikes Back", "Revenge of the…
## $ vehicles   <list> <"Snowspeeder", "Imperial Speeder Bike">, <…
## $ starships  <list> <"X-wing", "Imperial shuttle">, <>, <>, "TI…
```

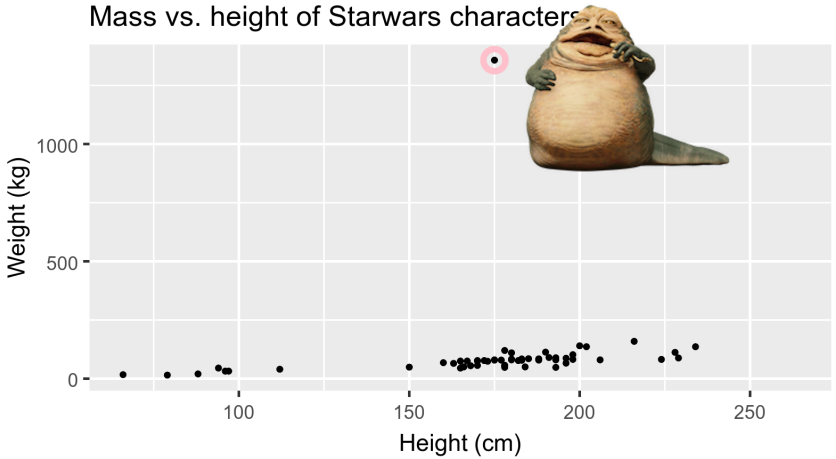
How many rows and columns does this dataset have? What does each row represent? What does each column represent?

Exploratory Data Analysis

- Exploratory data analysis (EDA) is an approach to analysing data sets to summarize its main characteristics
- Often, this is visual – this is what we'll focus on first
- But we might also calculate summary statistics and perform data wrangling/manipulation/transformation at (or before) this stage of the analysis – this is what we'll focus on next

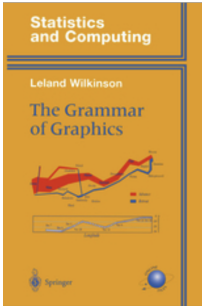
Mass vs. Height

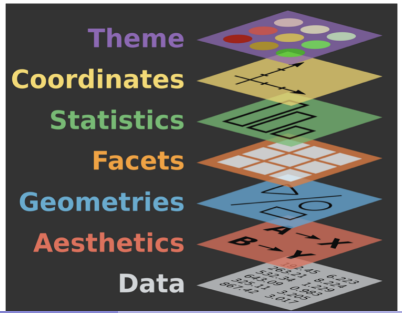
How would you describe the relationship between mass and height of Starwars characters? Who is the not so tall but really chubby character? Jabba!



Data Visualization

- Data visualization is the creation and study of the visual representation of data
- Many tools for visualizing data – R is one of them
- Many approaches/systems within R for making data visualizations – ggplot2 is one of them, and that's what we're going to use





Mass vs. Height

- What are the functions doing the plotting?
- What is the dataset being plotted?
- Which variables map to which features (aesthetics) of the plot?
- What does the warning mean?

```
ggplot(data = starwars, mapping = aes(x = height, y = mass)) +
  geom_point() +
  labs(title = "Mass vs. height of Starwars characters",
       x = "Height (cm)", y = "Weight (kg)")
```

```
## Warning: Removed 28 rows containing missing values (geom_point).
```

Visualising Data with ggplot2

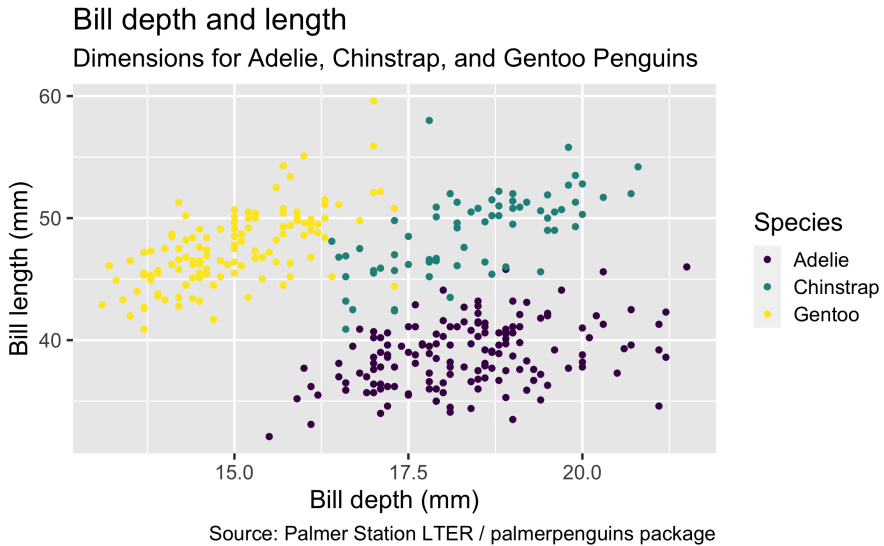
- ggplot2 is tidyverse's data visualization package
- Structure of the code for plots can be summarized as:

```
ggplot(data = [dataset],
       mapping = aes(x = [x-variable],
                     y = [y-variable])) +
   geom_xxx() +
   other options
```

Data: Palmer Penguins. Measurements for penguin species, island in Palmer Archipelago, size (flipper length, body mass, bill dimensions), and sex.

```
library(palmerpenguins)
glimpse(penguins)
```

Visualising Data with ggplot2



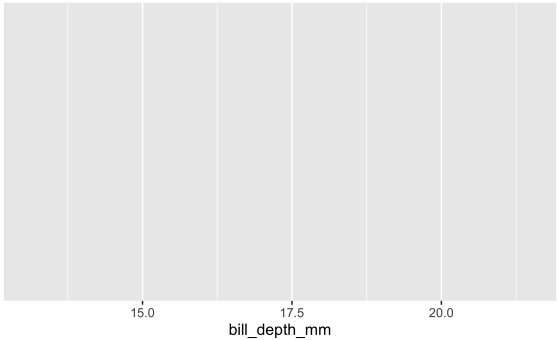
Coding Out Loud

Start with the penguins data frame

```
ggplot(data = penguins)
```

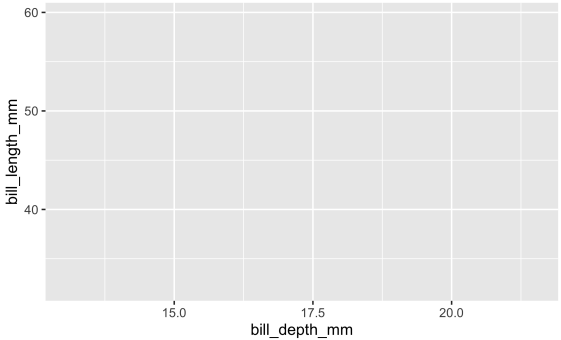
Start with the penguins data frame, map bill depth to the x-axis

```
ggplot(data = penguins,
       mapping = aes(x = bill_depth_mm))
```



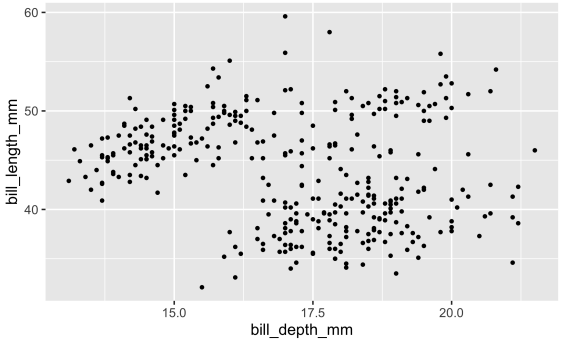
Start with the penguins data frame, map bill depth to the x-axis and map bill length to the y-axis.

```
ggplot(data = penguins,
       mapping = aes(x = bill_depth_mm,
                     y = bill_length_mm))
```



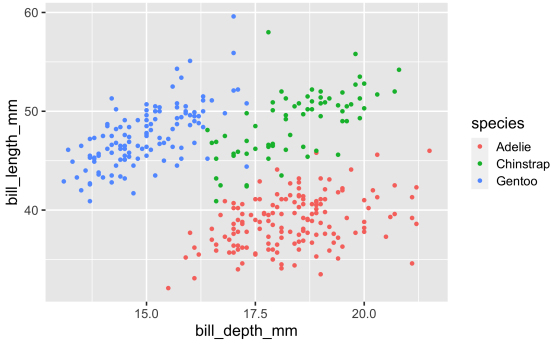
Start with the penguins data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point

```
ggplot(data = penguins,
       mapping = aes(x = bill_depth_mm,
                     y = bill_length_mm))
  geom_point()
```



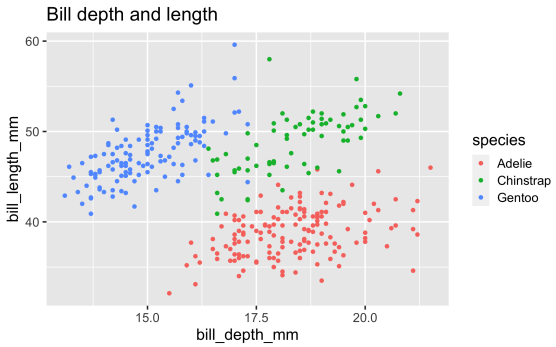
Start with the penguins data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the colour of each point.

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



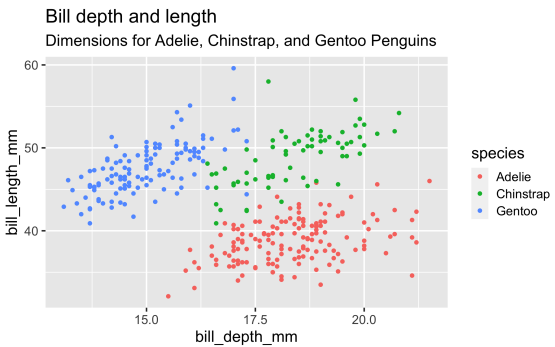
Start with the penguins data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the colour of each point. Title the plot "Bill depth and length"

```
ggplot(data = penguins,
       mapping = aes(x = bill_depth_mm,
                     y = bill_length_mm,
                     colour = species)) +
  geom_point() +
  labs(title = "Bill depth and length")
```



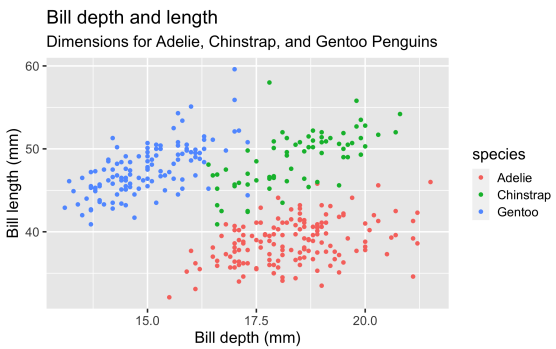
Start with the penguins data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the colour of each point. Title the plot "Bill depth and length", add the subtitle "Dimensions for Adelie, Chinstrap, and Gentoo Penguins"

```
ggplot(data = penguins,
       mapping = aes(x = bill_depth_mm,
                     y = bill_length_mm,
                     colour = species)) +
  geom_point() +
  labs(title = "Bill depth and length",
       subtitle = "Dimensions for Adelie,
```



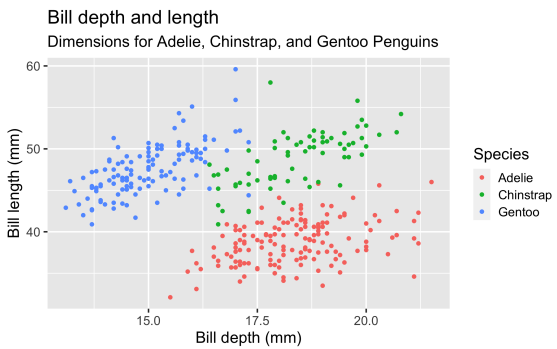
Start with the penguins data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the colour of each point. Title the plot "Bill depth and length", add the subtitle "Dimensions for Adelie, Chinstrap, and Gentoo Penguins", label the x and y axes as "Bill depth (mm)" and "Bill length (mm)", respectively

```
ggplot(data = penguins,
       mapping = aes(x = bill_depth_mm,
                     y = bill_length_mm,
                     colour = species)) +
  geom_point() +
  labs(title = "Bill depth and length",
       subtitle = "Dimensions for Adelie,
       x = "Bill depth (mm)", y = "Bill l
```



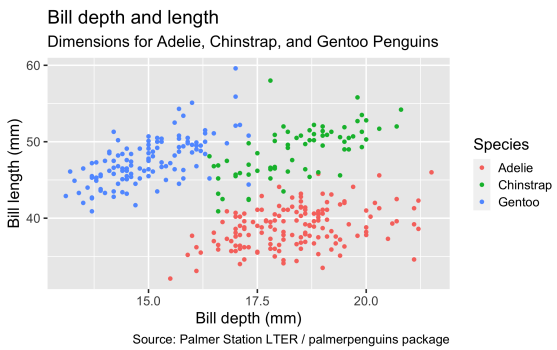
Start with the penguins data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the colour of each point. Title the plot "Bill depth and length", add the subtitle "Dimensions for Adelie, Chinstrap, and Gentoo Penguins", label the x and y axes as "Bill depth (mm)" and "Bill length (mm)", respectively, label the legend "Species"

```
ggplot(data = penguins,
       mapping = aes(x = bill_depth_mm,
                     y = bill_length_mm,
                     colour = species)) +
  geom_point() +
  labs(title = "Bill depth and length",
       subtitle = "Dimensions for Adelie,
       x = "Bill depth (mm)", y = "Bill l
       colour = "Species")
```



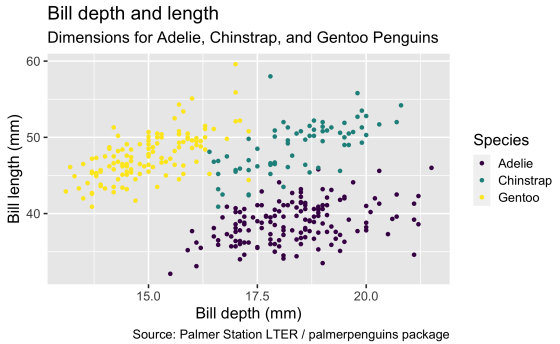
Start with the penguins data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the colour of each point. Title the plot "Bill depth and length", add the subtitle "Dimensions for Adelie, Chinstrap, and Gentoo Penguins", label the x and y axes as "Bill depth (mm)" and "Bill length (mm)", respectively, label the legend "Species", and add a caption for the data source.

```
ggplot(data = penguins,
       mapping = aes(x = bill_depth_mm,
                     y = bill_length_mm,
                     colour = species)) +
  geom_point() +
  labs(title = "Bill depth and length",
       subtitle = "Dimensions for Adelie,
       x = "Bill depth (mm)", y = "Bill l
       colour = "Species",
       caption = "Source: Palmer Station
```



Start with the penguins data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the colour of each point. Title the plot "Bill depth and length", add the subtitle "Dimensions for Adelie, Chinstrap, and Gentoo Penguins", label the x and y axes as "Bill depth (mm)" and "Bill length (mm)", respectively, label the legend "Species", and add a caption for the data source. Finally, use a discrete colour scale that is designed to be perceived by viewers with common forms of colour blindness.

```
ggplot(data = penguins,
       mapping = aes(x = bill_depth_mm,
                     y = bill_length_mm,
                     colour = species)) +
  geom_point() +
  labs(title = "Bill depth and length",
       subtitle = "Dimensions for Adelie,
       x = "Bill depth (mm)", y = "Bill le
       colour = "Species",
       caption = "Source: Palmer Station
  scale_colour_viridis_d()
```



Argument Names

You can omit the names of first two arguments when building plots with `ggplot()`.

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

```
ggplot(penguins,
       aes(x = bill_depth_mm,
           y = bill_length_mm,
           colour = species)) +
  geom_point() +
  scale_colour_viridis_d()
```

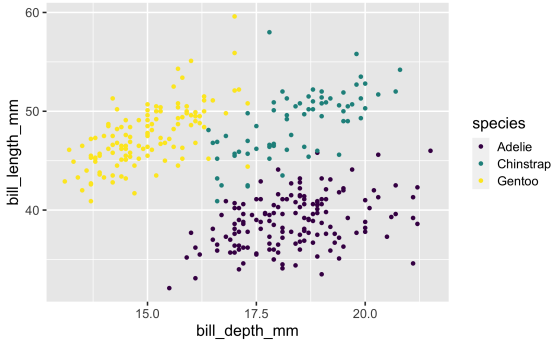
Aesthetics Options

Commonly used characteristics of plotting characters that can be *mapped to a specific variable* in the data are

- colour
- shape
- size
- alpha (transparency)

Colour

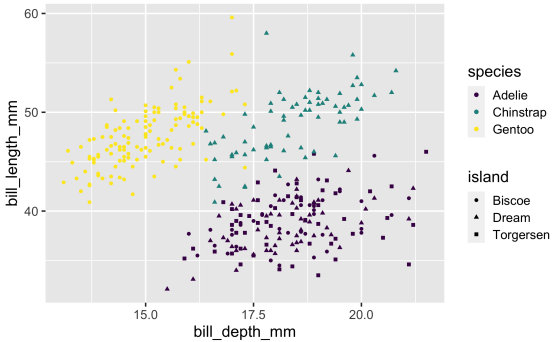
```
ggplot(penguins,
       aes(x = bill_depth_mm,
           y = bill_length_mm,
           colour = species)) +
  geom_point() +
  scale_colour_viridis_d()
```



Shape

Mapped to a different variable than colour

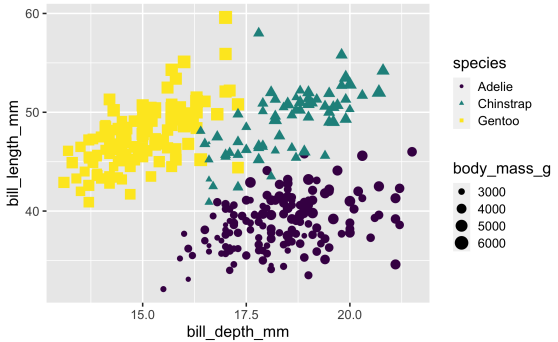
```
ggplot(penguins,
       aes(x = bill_depth_mm,
           y = bill_length_mm,
           colour = species,
           shape = island)) +
  geom_point() +
  scale_colour_viridis_d()
```



Size

Mapped to a different variable than colour

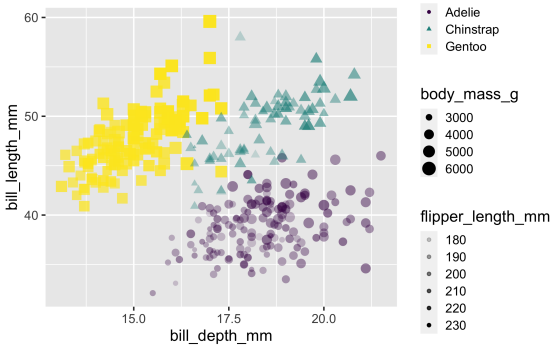
```
ggplot(penguins,
       aes(x = bill_depth_mm,
           y = bill_length_mm,
           colour = species,
           shape = species,
           size = body_mass_g)) +
  geom_point() +
  scale_colour_viridis_d()
```



Alpha

Mapped to a different variable than colour

```
ggplot(penguins,
       aes(x = bill_depth_mm,
           y = bill_length_mm,
           colour = species,
           shape = species,
           size = body_mass_g,
           alpha = flipper_length_mm)) +
  geom_point() +
  scale_colour_viridis_d()
```



Mapping vs. Setting

Mapped to a different variable than colour

- Mapping: Determine the size, alpha, etc. of points based on the values of a variable in the data. Goes into aes()
- Setting: Determine the size, alpha, etc. of points not based on the values of a variable in the data. Goes into geom_*()

Alpha

Mapped to a different variable than colour

```
ggplot(penguins,
       aes(x = bill_depth_mm,
           y = bill_length_mm,
           colour = species,
           shape = species,
           size = body_mass_g,
           alpha = flipper_length_mm)) +
  geom_point() +
  scale_colour_viridis_d()
```