15 ggplot2

Data vizualization with `ggplot2`

Read this

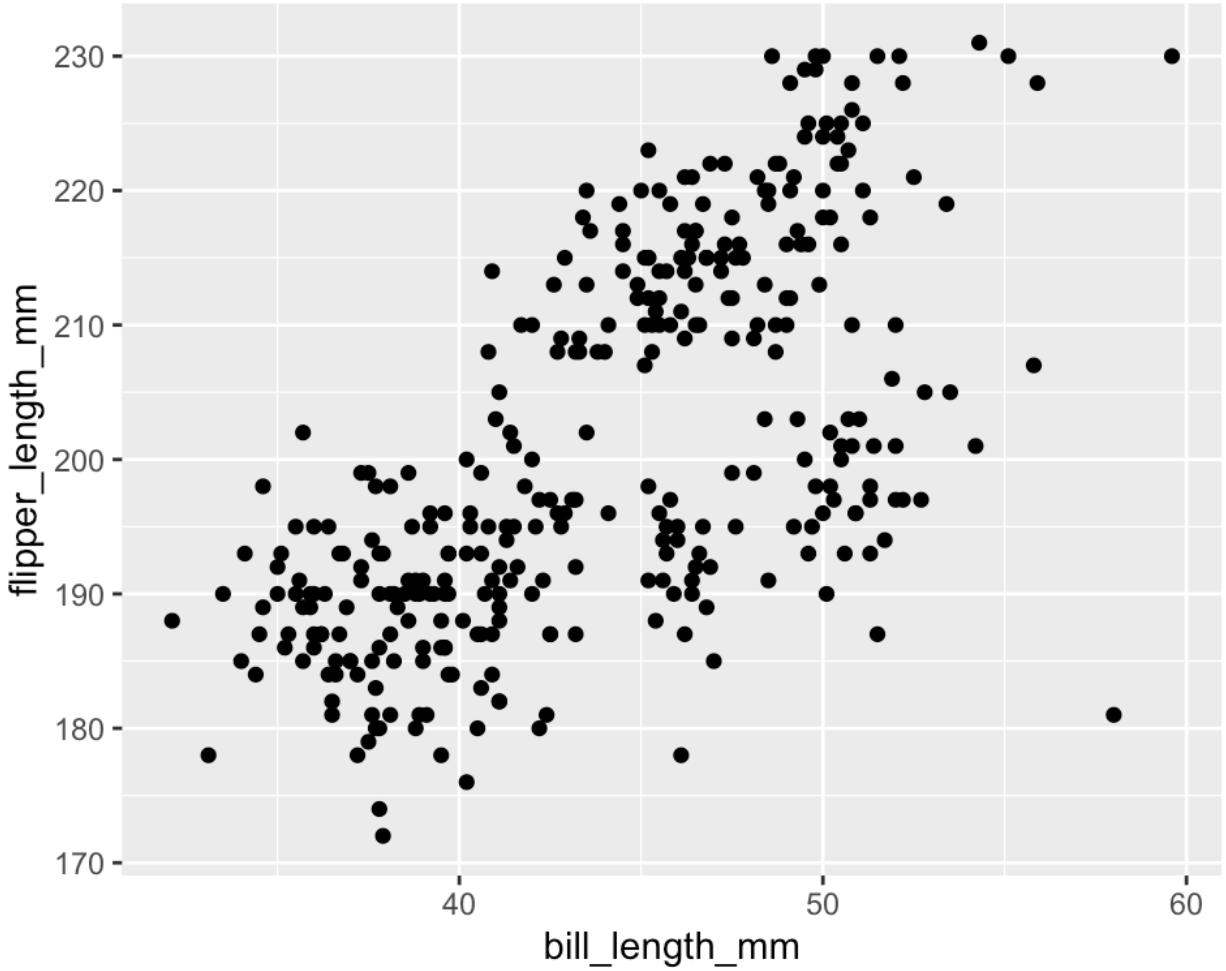
ggplot2

- `base R` plot functions require more effort and expertise to create high-quality publishable graphs
- Over the years, many R packages (e.g. `lattice`, `grid`, etc.) were introduced to overcome the limitations of `base R` plot functions
- The newest addition to R plot functions is `ggplot2` package and it can be used to produce elegant plots without much effort!

ggplot2

- `ggplot2` package implements the `grammar of graphics`, a coherent system for describing and building graphs
- To load `ggplot2` package to the current R environment

1 Scatter plot



Creating a ggplot

`ggplot(data = penguins) +`

- **A blank slate:** It creates a coordinate system to which several layers can be added
- All plot functions of `ggplot2` package begin with the `ggplot()` function
- `data` is the first argument of `ggplot()` and it specifies the data frame to be used for the plot
- One or more layers can be added to `ggplot()` using a plus (`+`) sign

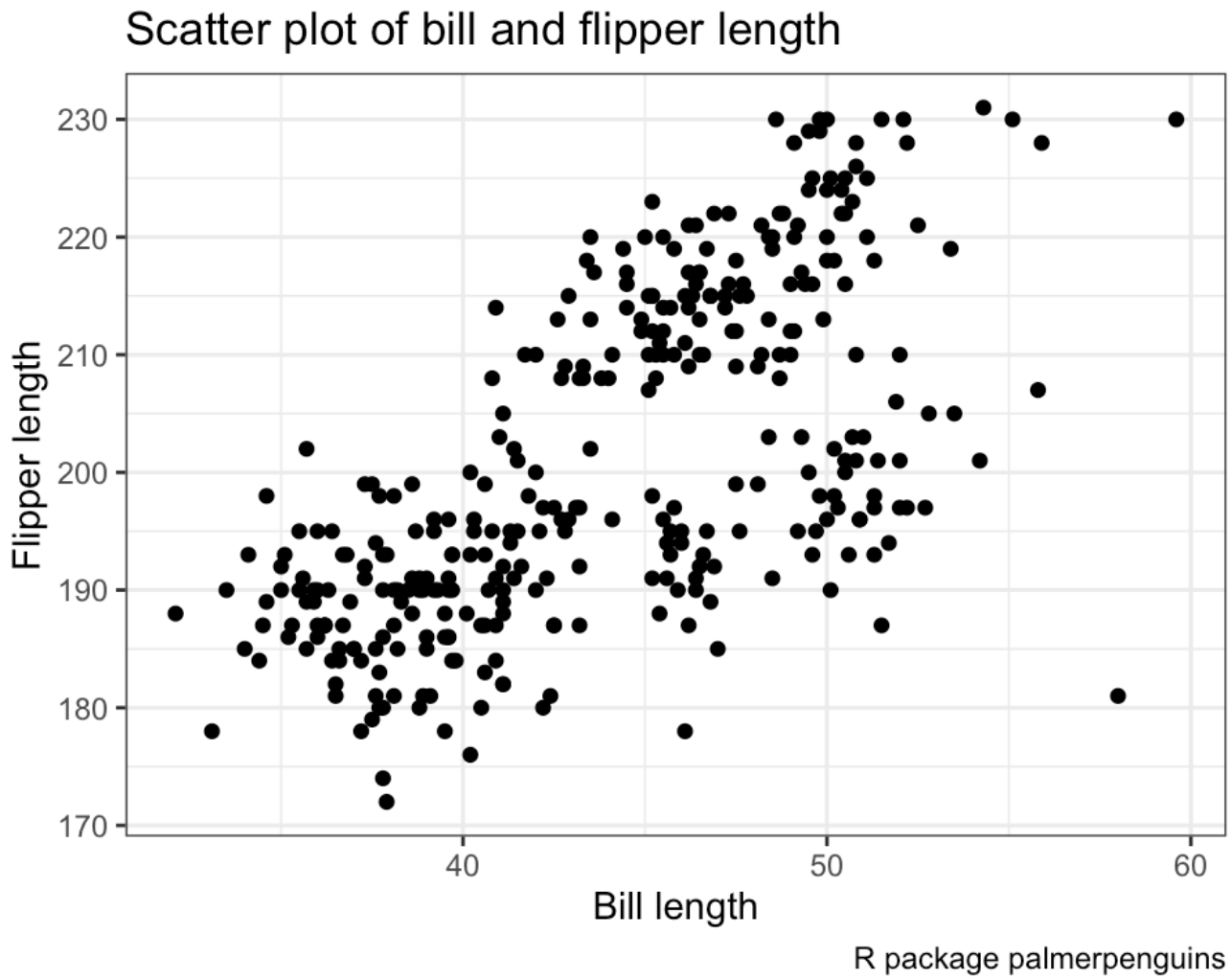
`geom_point`

- Geometric objects (called `geom`) are the shapes we put on a plot (e.g. points, bars, etc.).
- You can have an unlimited number of layers, but at a minimum a plot **must have at least one geom**
  - `geom_point()` makes a scatter plot by adding a layer of points.
  - `geom_line()` adds a layer of lines connecting data points.
  - `geom_col()` adds bars for bar charts.
  - `geom_histogram()` makes a histogram.
  - `geom_boxplot()` adds boxes for boxplots

`mapping = aes()`

- Each type of `geom` usually has a **required set of aesthetics** to be set. Aesthetic mappings are set with the `aes()` function. Examples include
  - `x` and `y` (the position on the x and y axes)
  - `color` ("outside" color, like the line around a bar)
  - `fill` ("inside" color, like the color of the bar itself)
  - `shape` (the type of point, like a dot, square, triangle, etc.)
  - `linetype` (solid, dashed, dotted etc.)
  - `size` (of geoms)

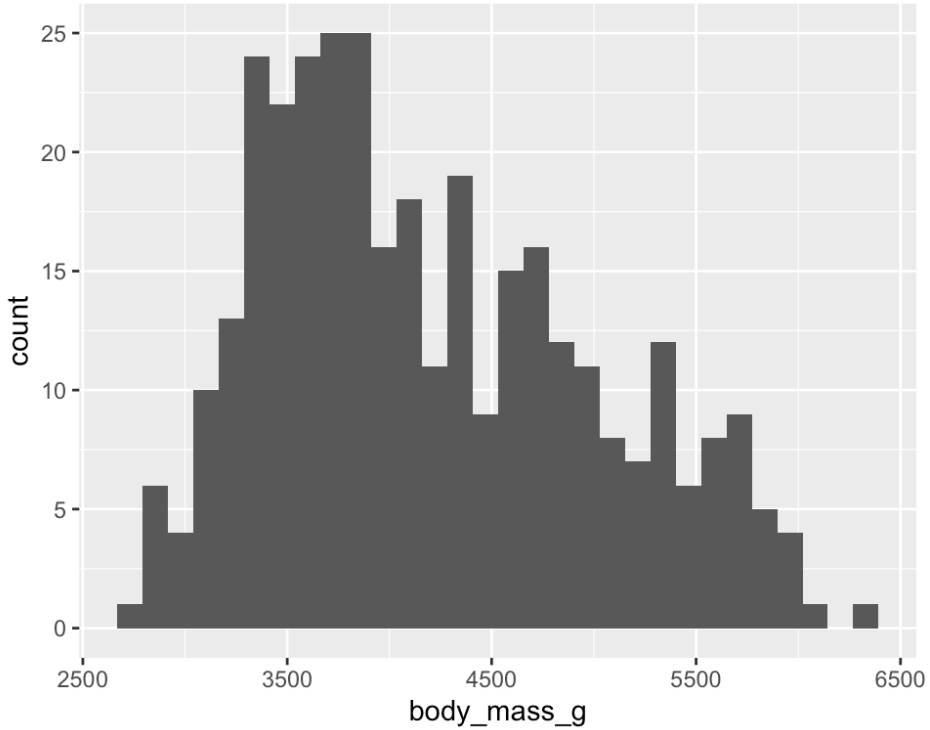
Adding labels, title, and caption to a graph



Adding labels, title, and caption to a graph

2 Histogram

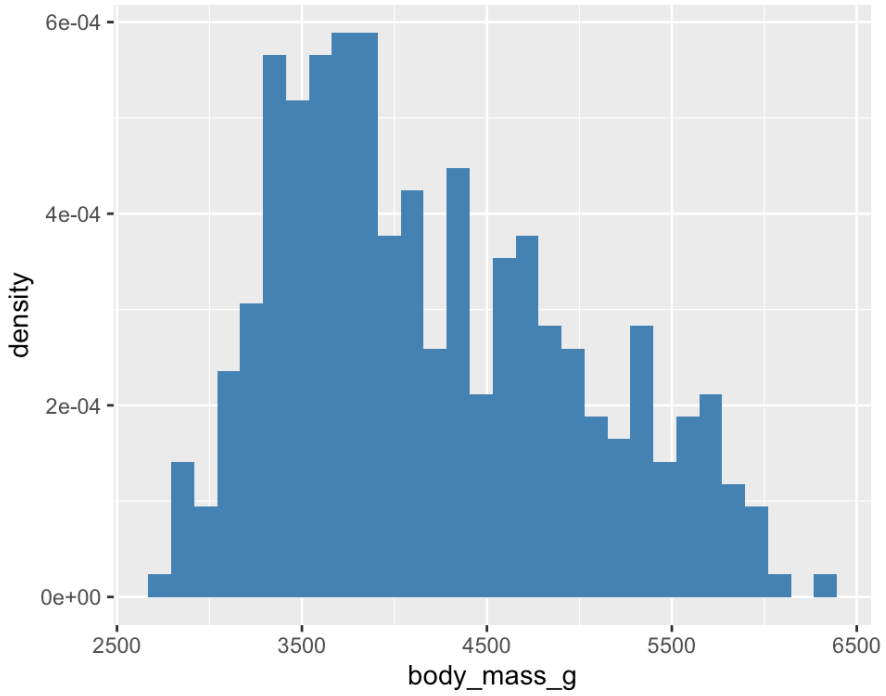
- `geom_histogram()` is for histogram
- Only `x` value is needed for its `aes()` function



Adding labels, title, and caption to a graph

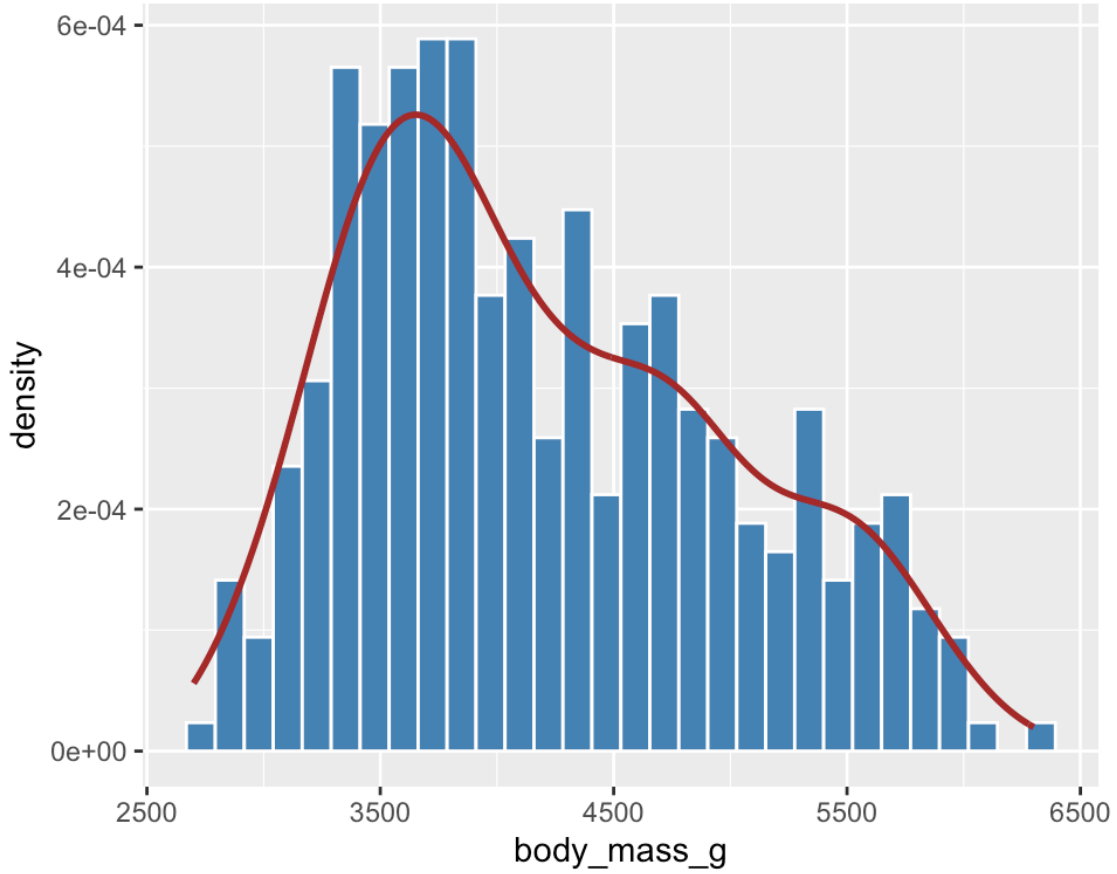
`fill` argument | `col` argument

- `fill` argument of `geom_histogram()` modifies color of the bars



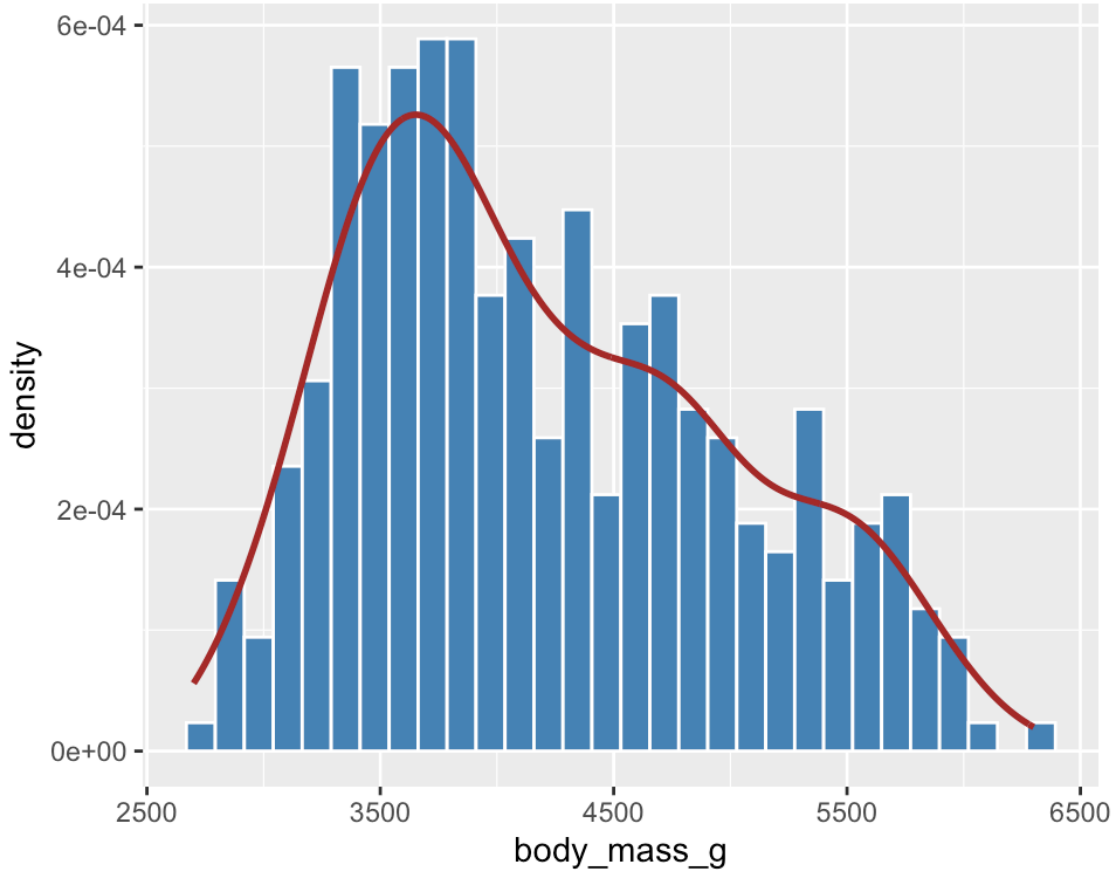
Histogram and density function

- `geom_density()` is used to obtain the density of a variable



Histogram and density function

- A common mapping function in `ggplot()` for different `geom_*()`



Exercise 3.2.1

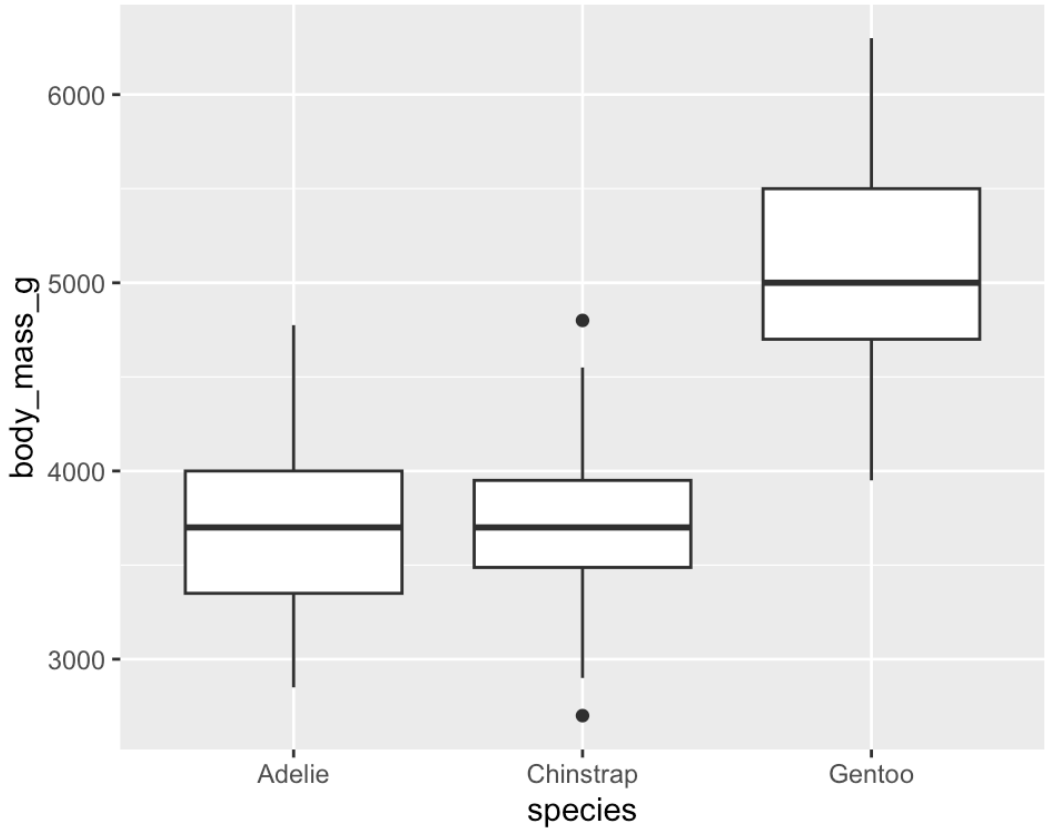
(use `diamonds` data to answer the followings)

- Create a histogram of `carat` and check the effect of `bins` on histogram
- Add a density line to the plot obtained in Question 1

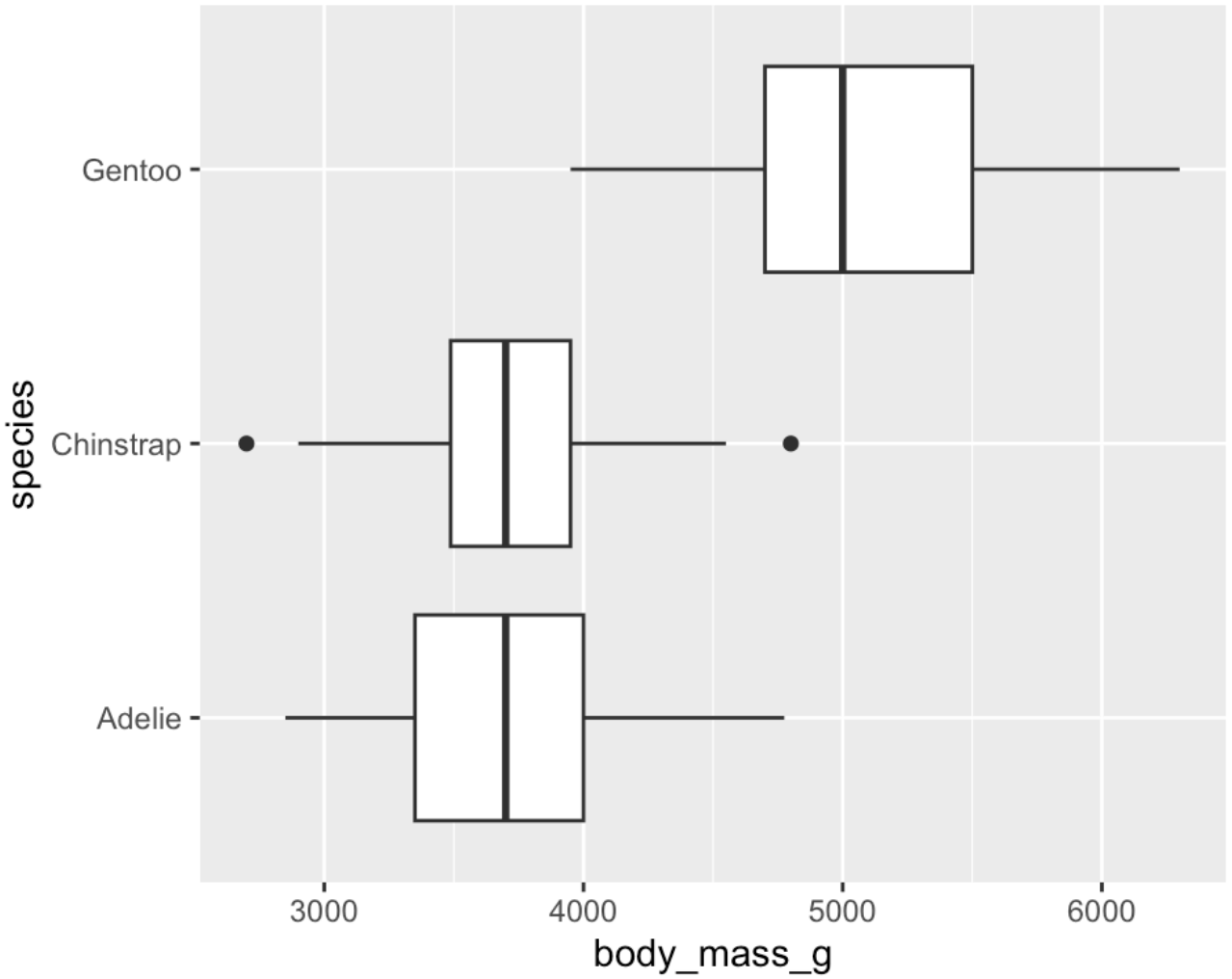
Exercise 3.2.1

3 Boxplot

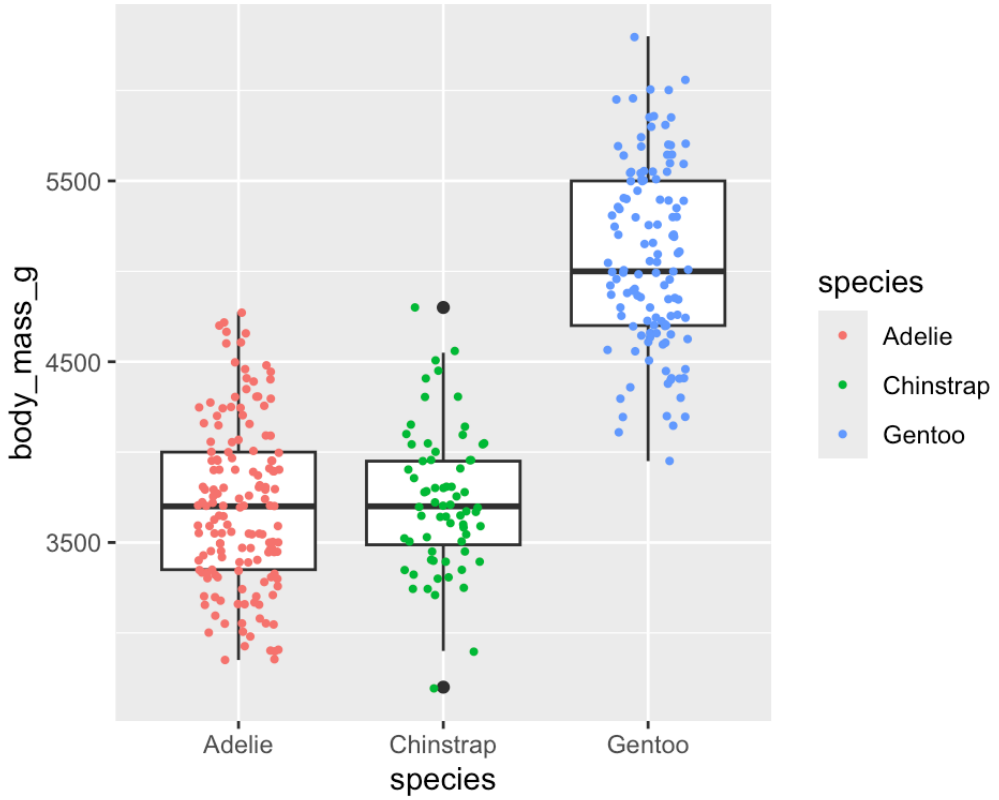
- `geom_boxplot()` is used to obtain a boxplot



Boxplot


Gentoo
Chinstrap
Adelie
species
3000
4000
5000
6000
body_mass_g

Boxplot with original data points



- `geom_jitter()` adds a small amount random variation to each point and it is useful to visualize points at different levels

Exercise 3.2.2

(use `diamonds` data to answer the followings)

- Create a boxplot of `carat` at different levels of `cut`
- Create a scatter plot to examine the effect of `carat` on `price`

Aesthetic mappings

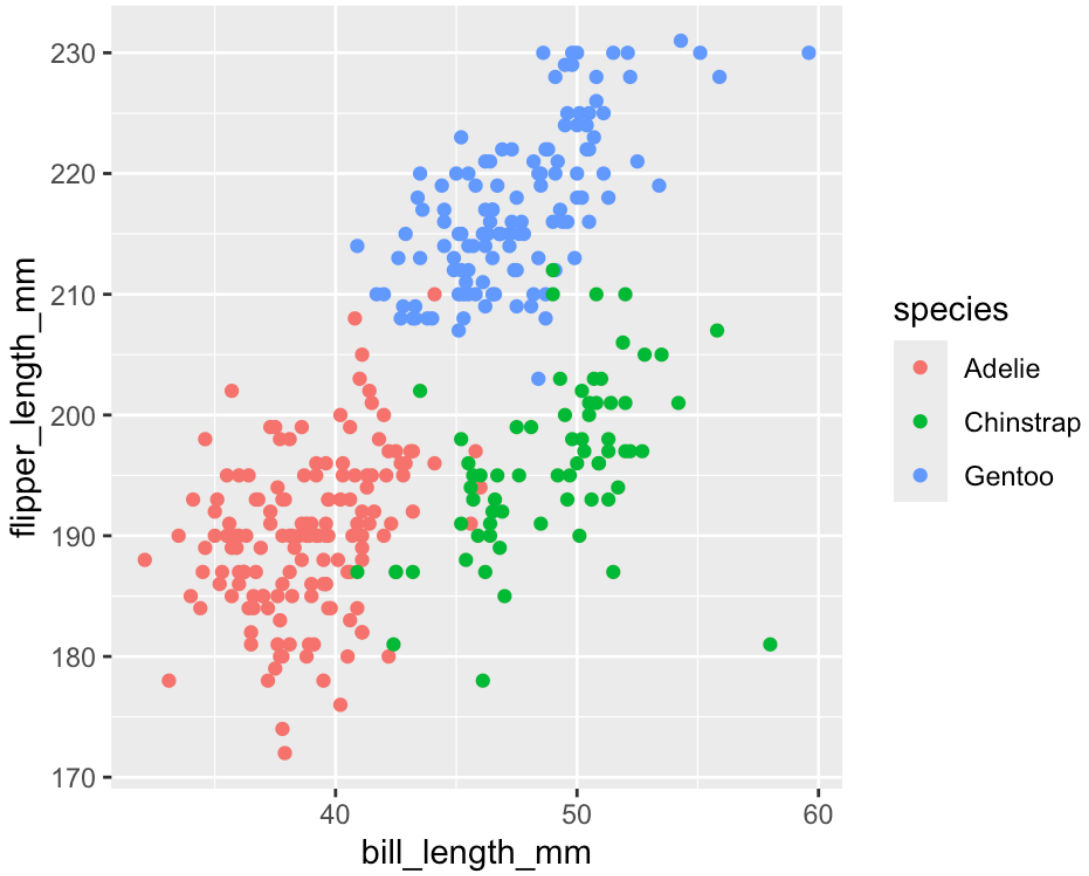
Aesthetic mappings

- A third variable can be added to a two-dimensional scatter plot by mapping it to an *aesthetic*
- A aesthetic is a visual property (such as the size, shape, and color of the points) of the plot
- Points of a plot can be displayed in different ways by changing the levels of its aesthetic properties (e.g. size, shape, or color of points can be changed)

Aesthetic mappings

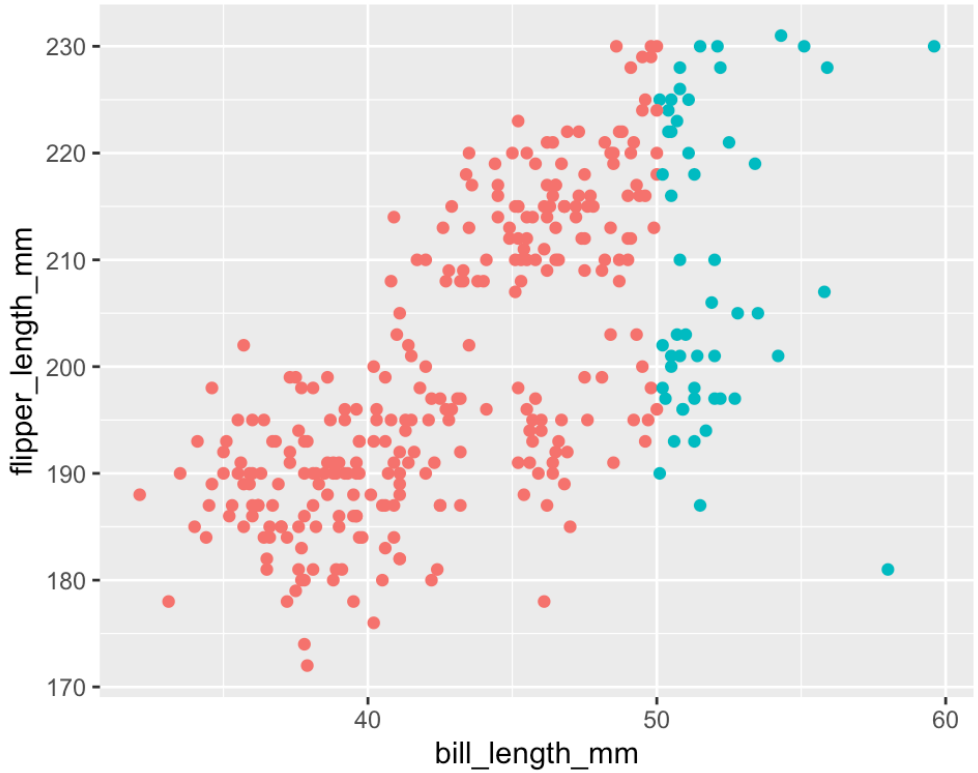
- Variables can be linked to the graph using the following properties
- positions (`x`, `y`)
- colors (`color`, `fill`)
- shapes (`shape`, `linetype`)
- size (`size`)
- transparency (`alpha`)
- groupings (`group`)

Aesthetic mappings



- `col` is specified by different levels of `species`

Aesthetic mappings

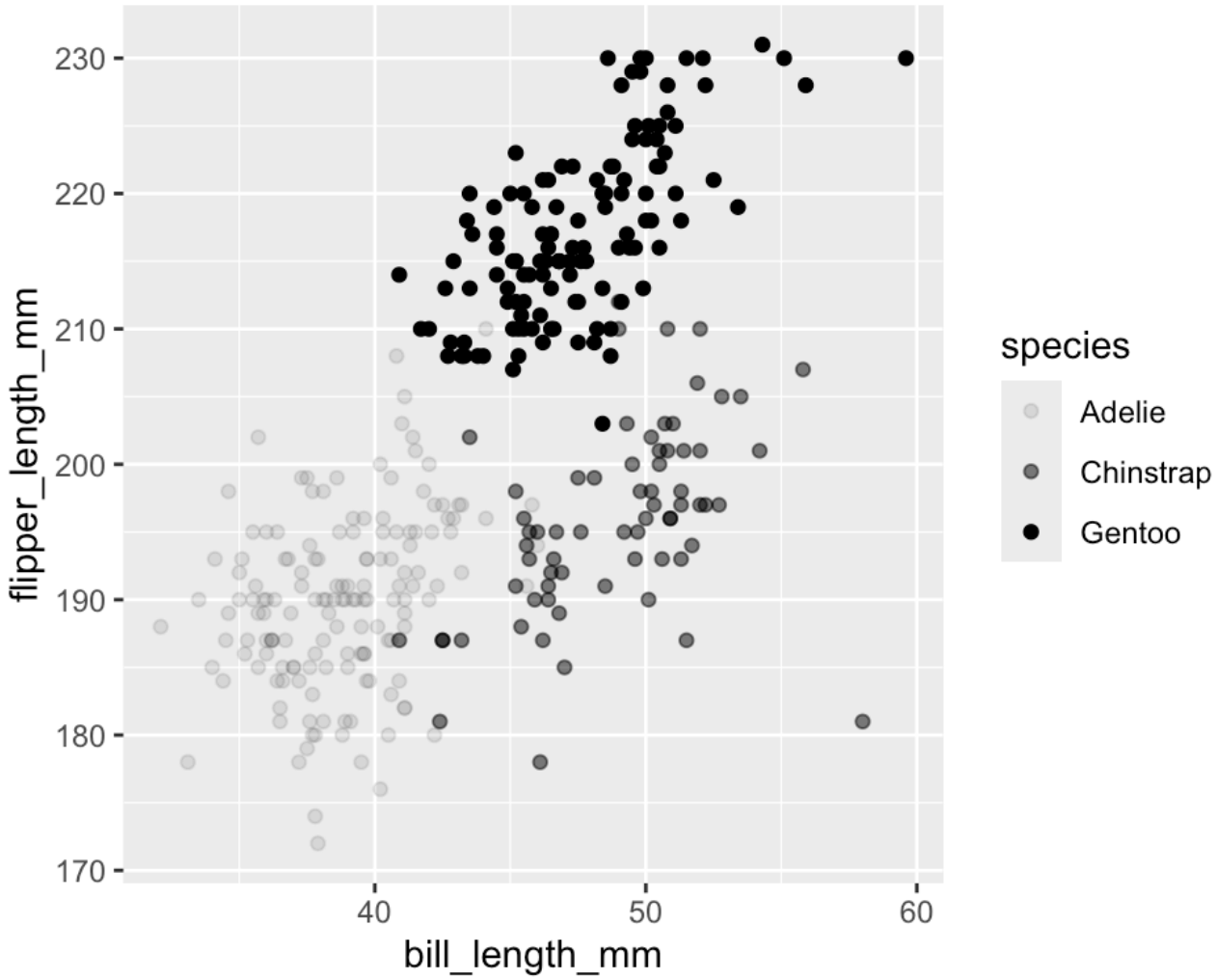


- `col` is specified by a function of `bill_length_mm`
- `show_legend` is a logical argument of `geom_*`

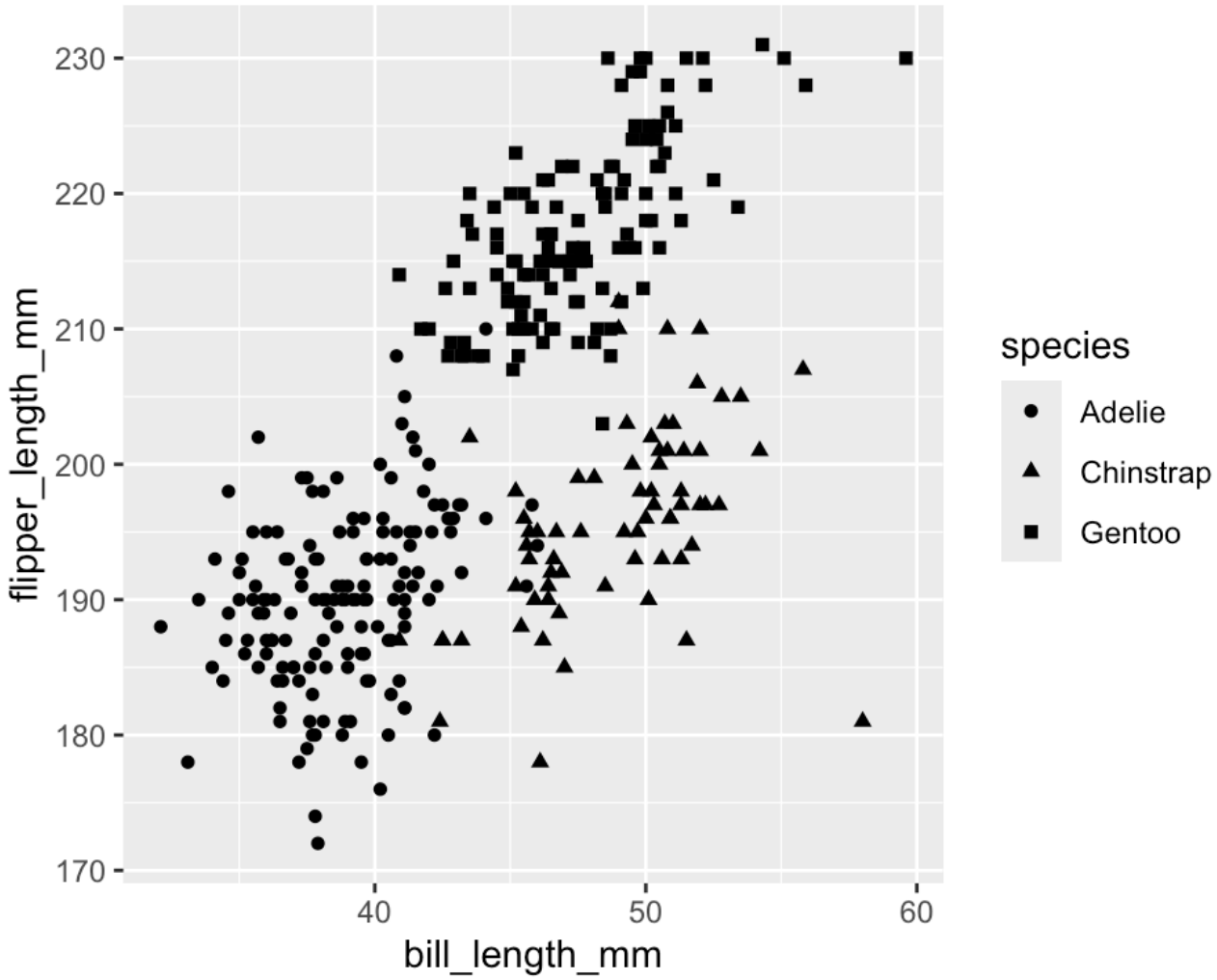
Aesthetic mappings

- Besides `col`, some other aesthetic types are useful in `ggplot2`
  - `size` → assigns different sizes of the points to different values of the variable
  - `alpha` → controls the transparency of the point
  - `shape` → assigns different (at most six) shapes to different values of the variable
- `ggplot2` creates a legend for the variables used in the arguments of `aes()` except for `x` and `y`

Aesthetic mappings



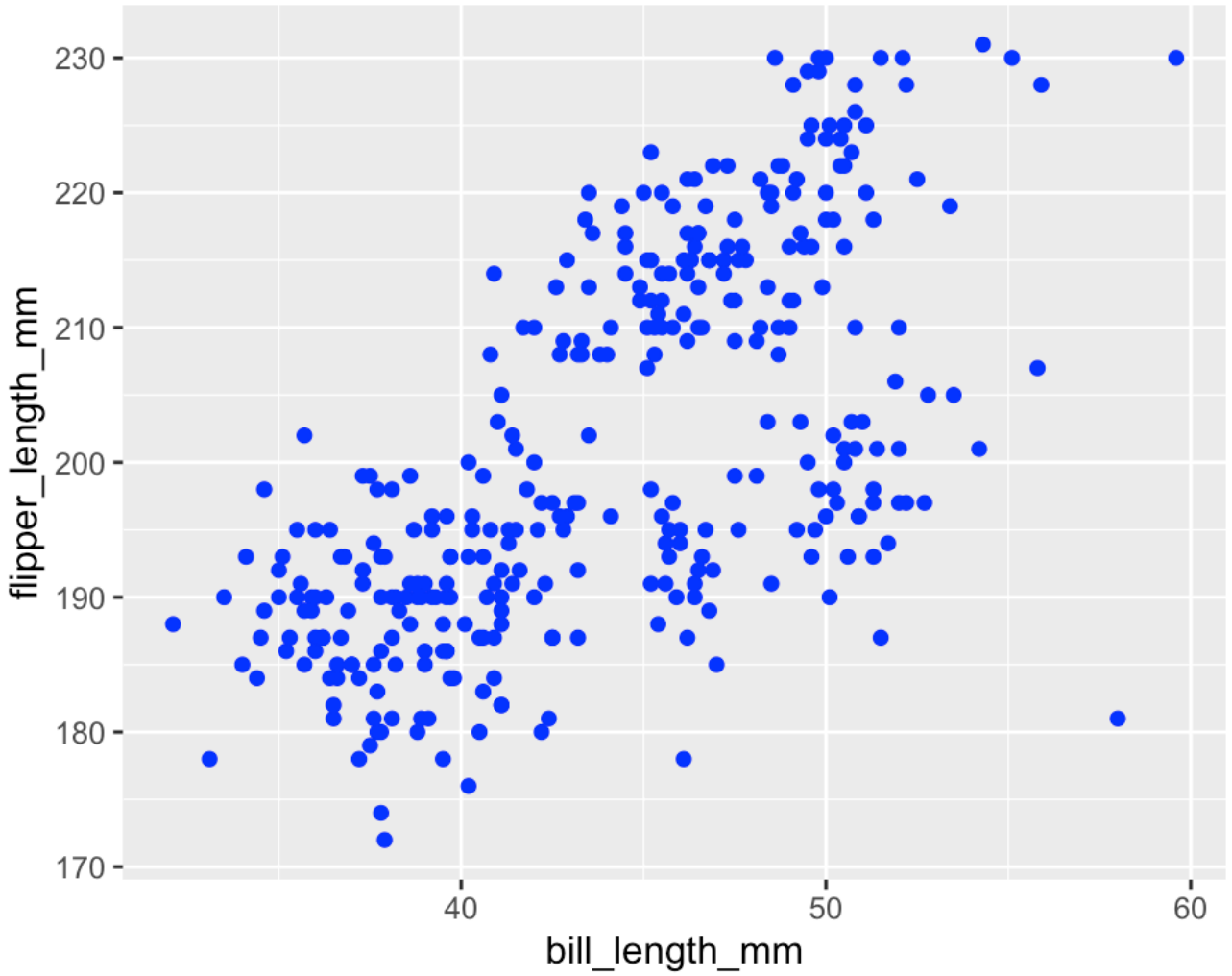
Aesthetic mappings



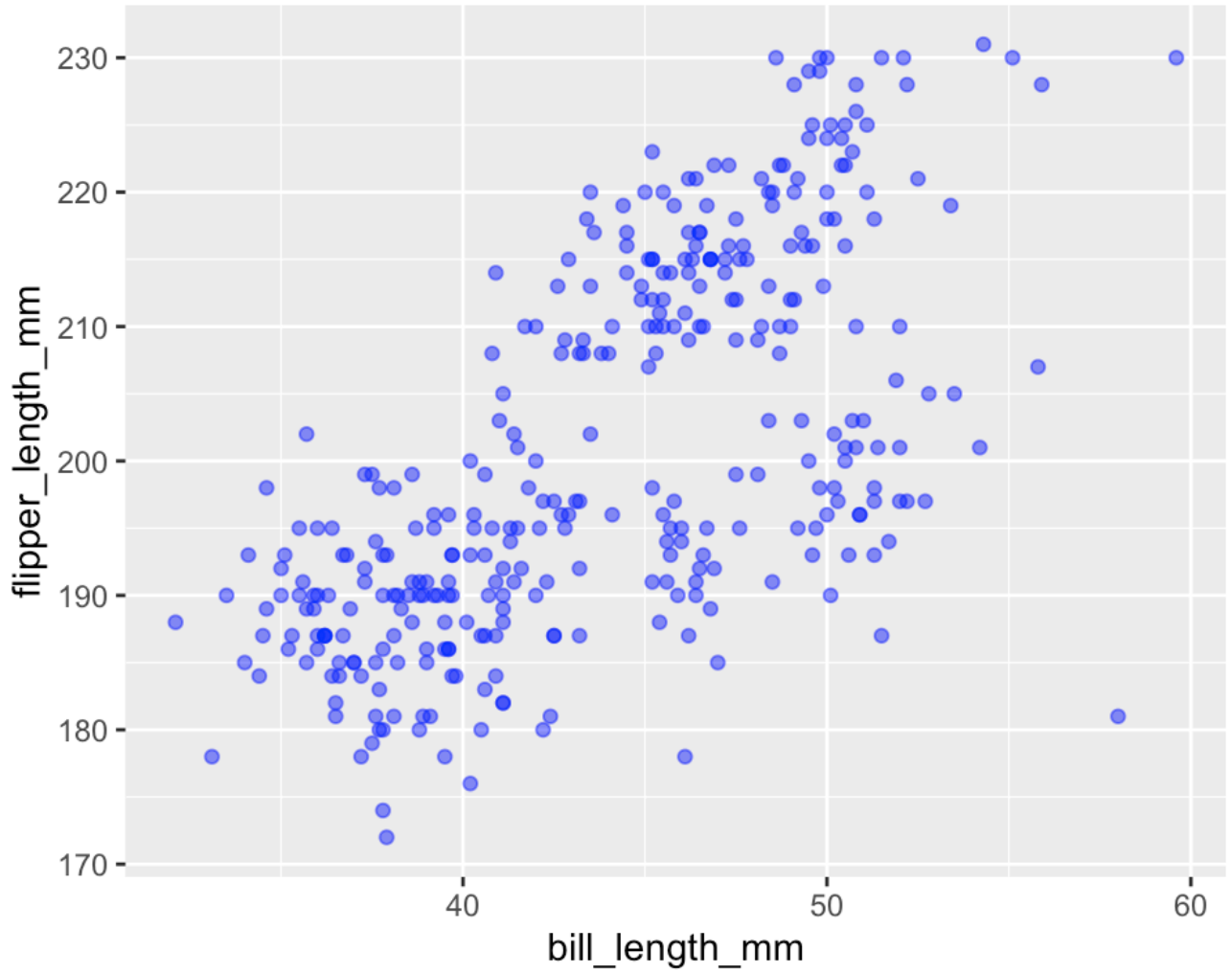
Aesthetic mappings

- Aesthetic properties can also be set manually, e.g. `col = "blue"` will make all the points blue, which does not convey any information about a variable but only changes the appearance of the plot
- To set an aesthetic manually, the aesthetic type needs to be defined outside of `aes()` as an argument of `geom_??` function

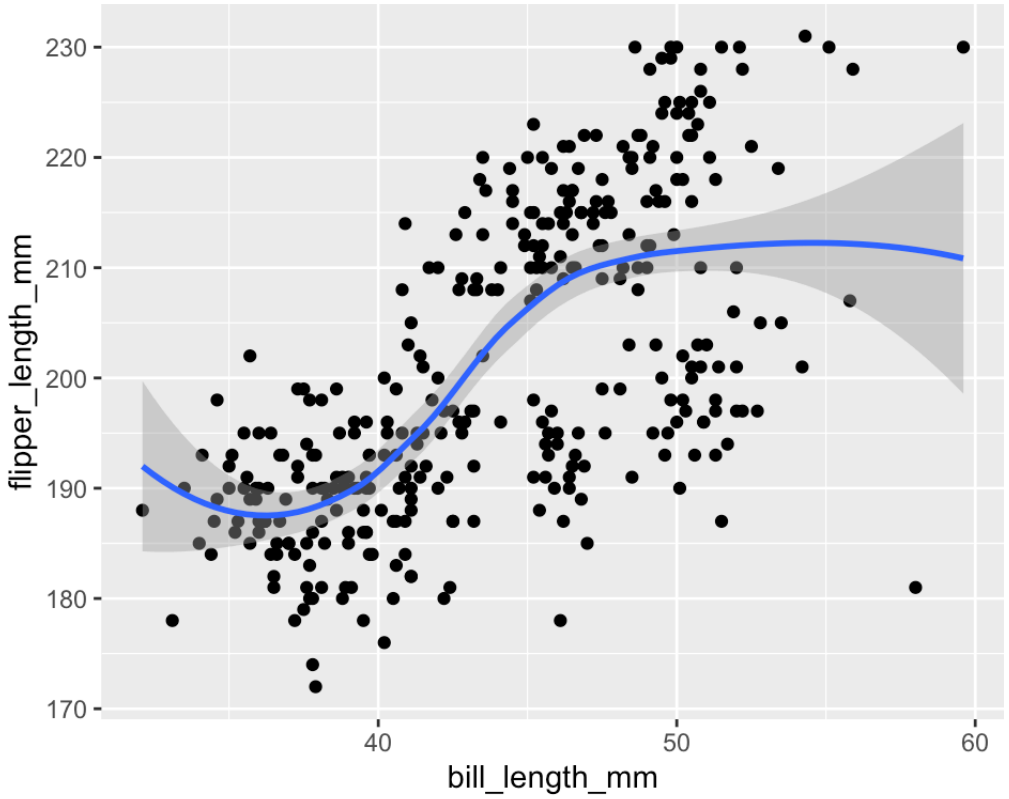
Aesthetic mappings



Aesthetic mappings

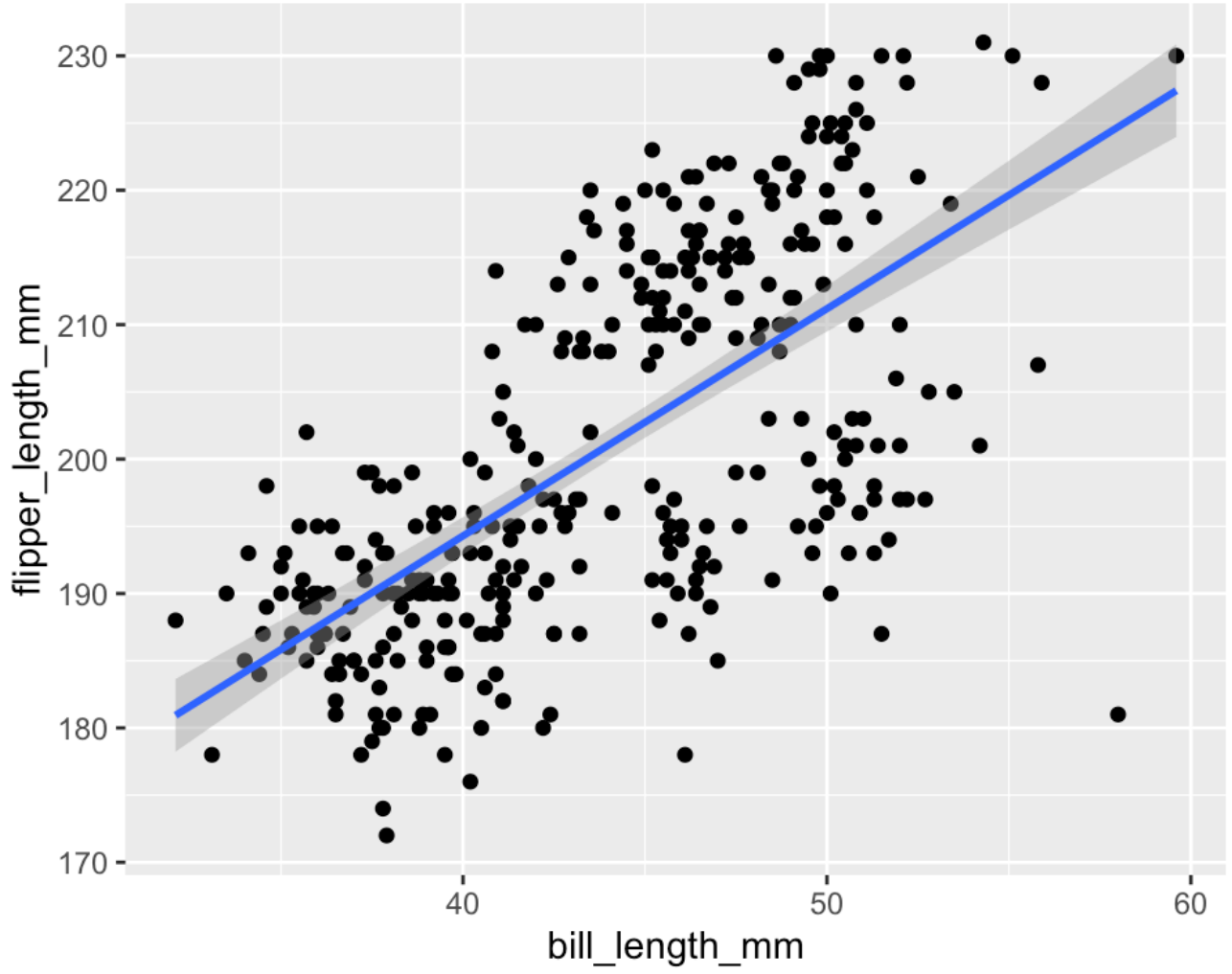


`geom_smooth()`

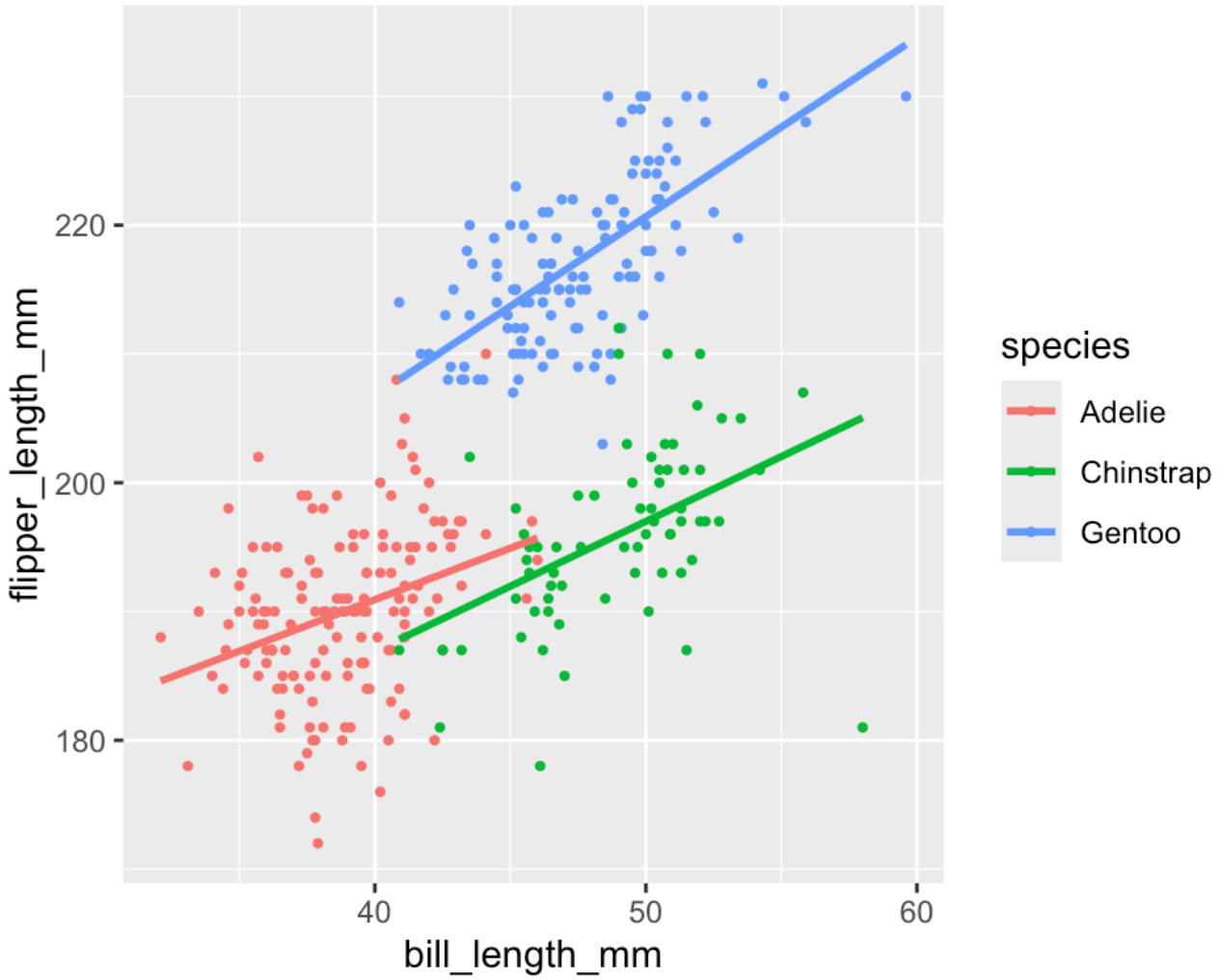


- `geom_smooth()` fits the relationship between two quantitative using a smoothing method

geom_smooth()


230
220
210
200
190
180
170
flipper_length_mm
40
50
60
bill_length_mm

geom_smooth()


flipper_length_mm
220
200
180
40
50
60
bill_length_mm
species
Adelie
Chinstrap
Gentoo

Exercise 3.2.3

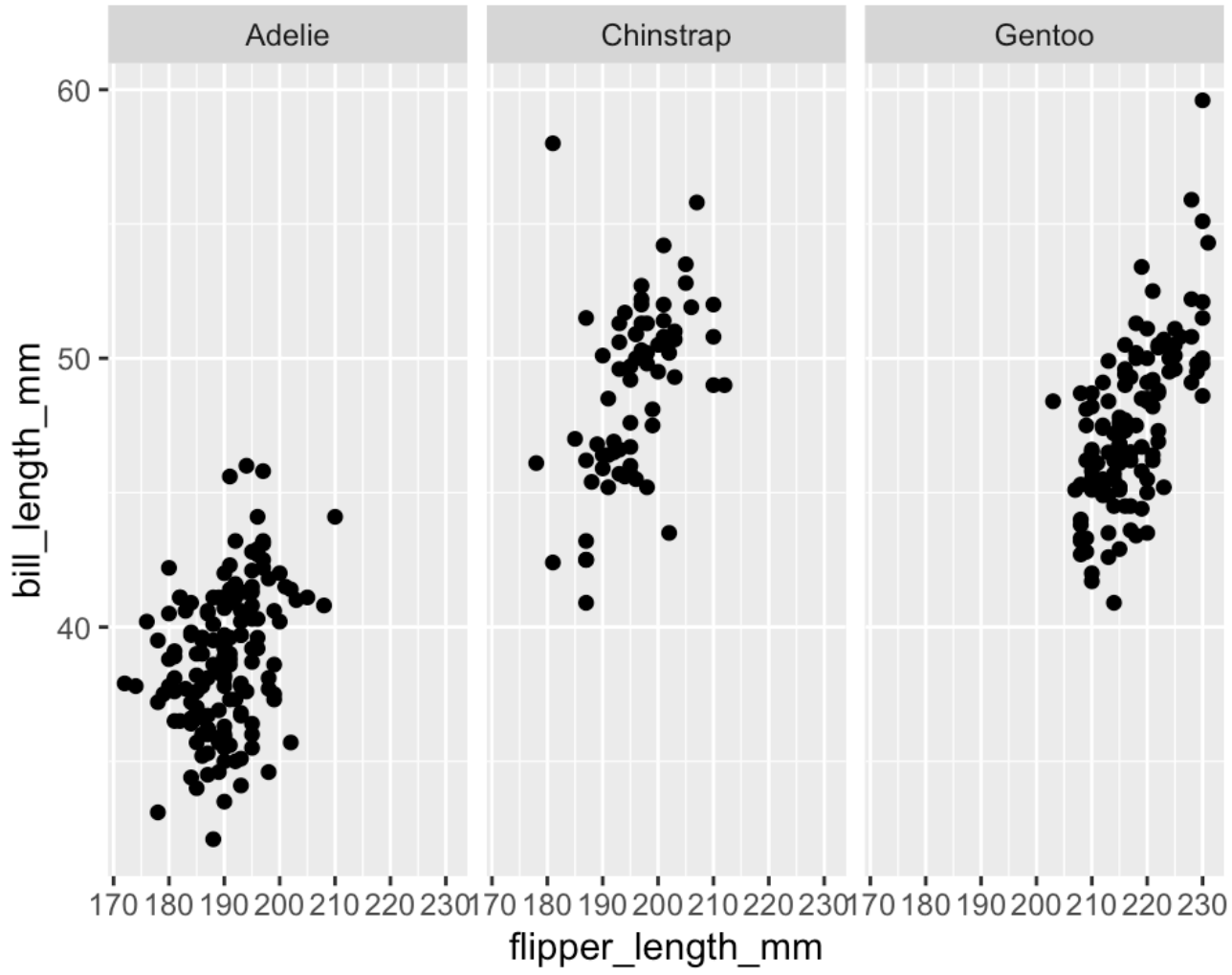
(use `diamonds` data to answer the followings)

- Create a scatter plot to examine the effect of `price` on `carat` and assign different colors to different levels of `cut`
- Show a fit of a linear model on the scatter plot of `carat` and `price`
- Show different fits of linear models (`price` on `carat`) corresponding to different levels of `cut` on the scatter plot of `price` and `carat`

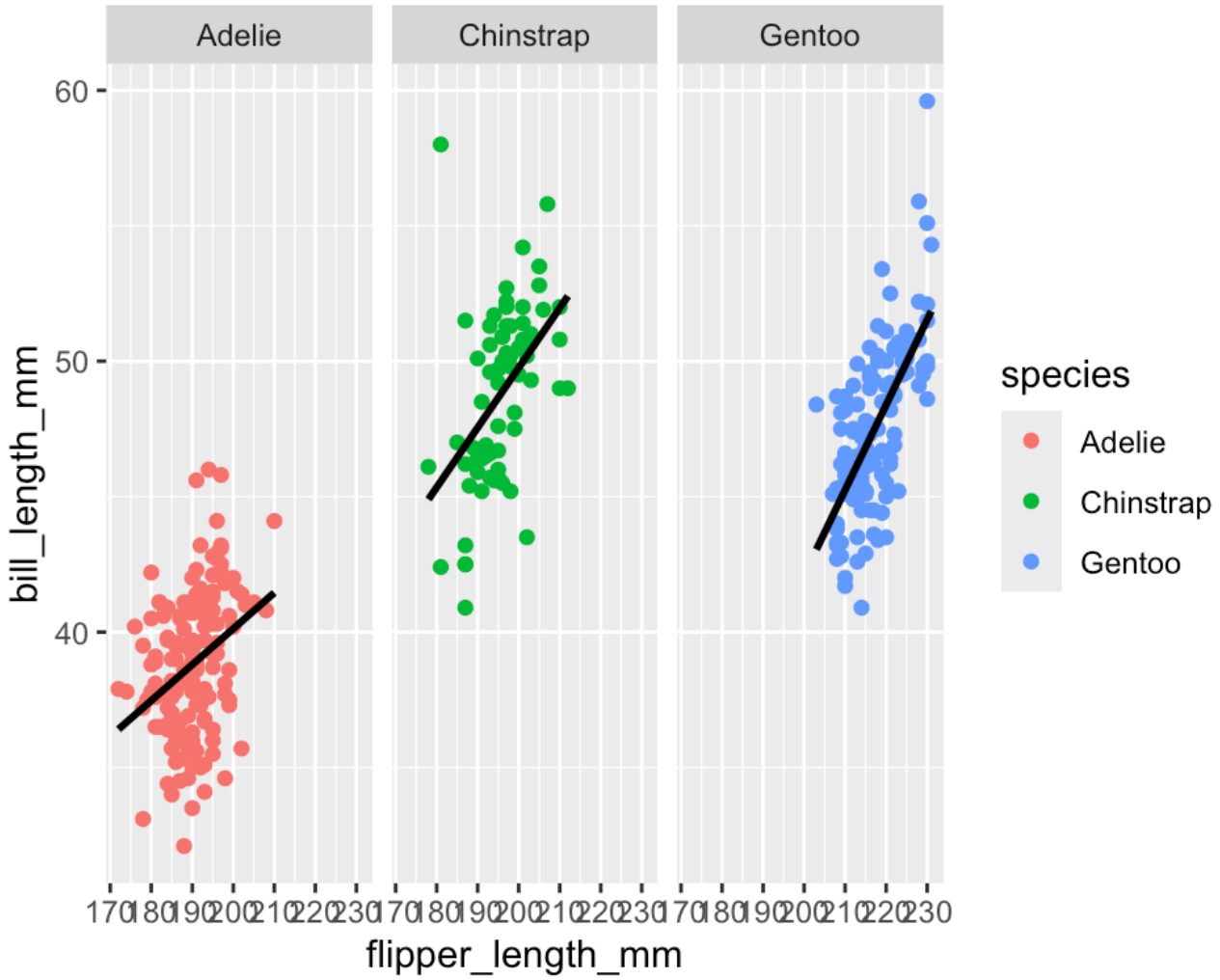
Facets

- Adding information about a new variable to an existing plot could be helpful for data analysis (e.g. aesthetic)
- `facets` can add information about a categorical variable to an existing plot by splitting the plot according to the levels of the categorical variable
  - `facet_wrap()` → splits the plot by a single variable
  - `facet_grid()` → splits the plot by the combination of two variables

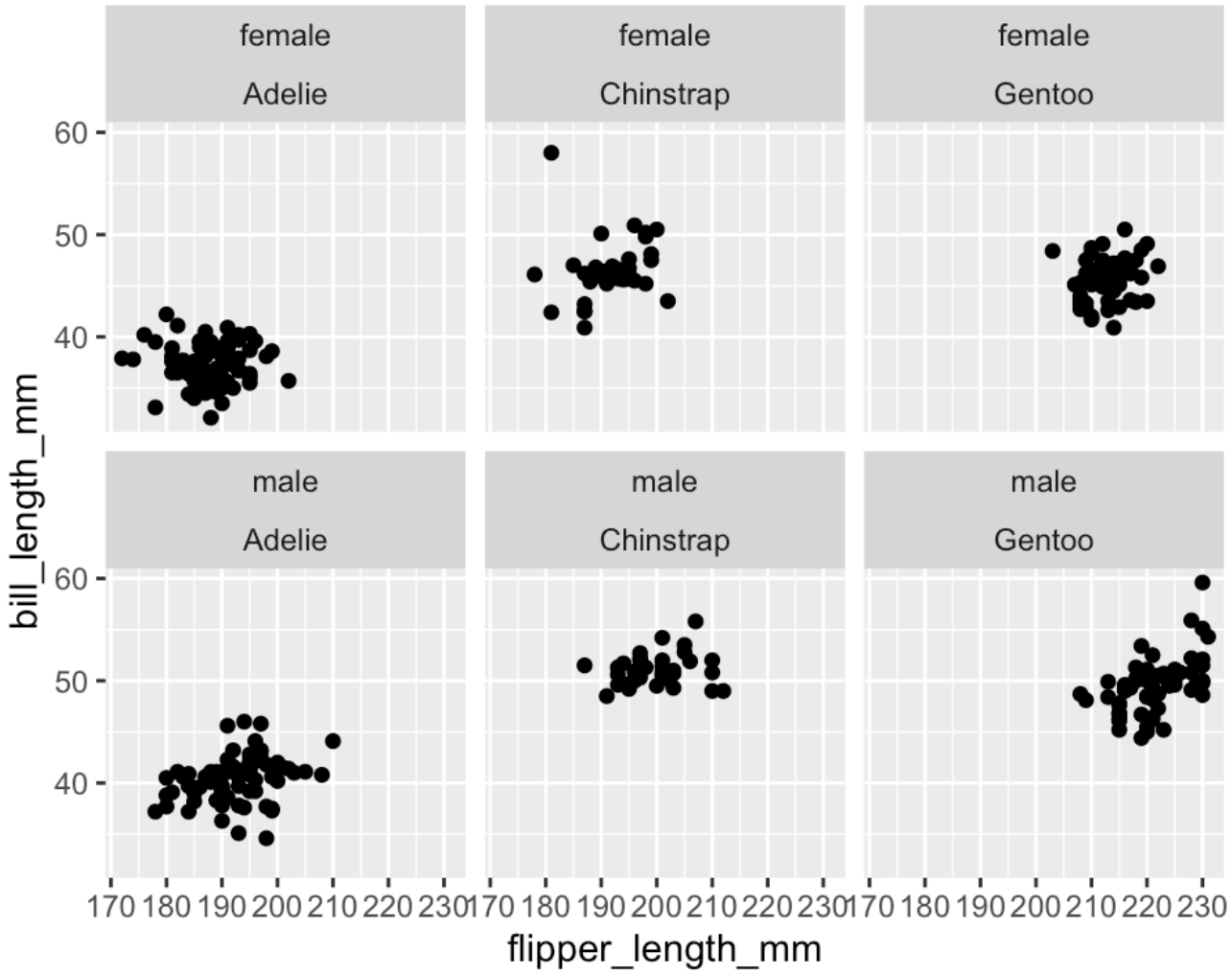
Facets


Adelie
Chinstrap
Gentoo
bill_length_mm
60
50
40
170 180 190 200 210 220 230
flipper_length_mm

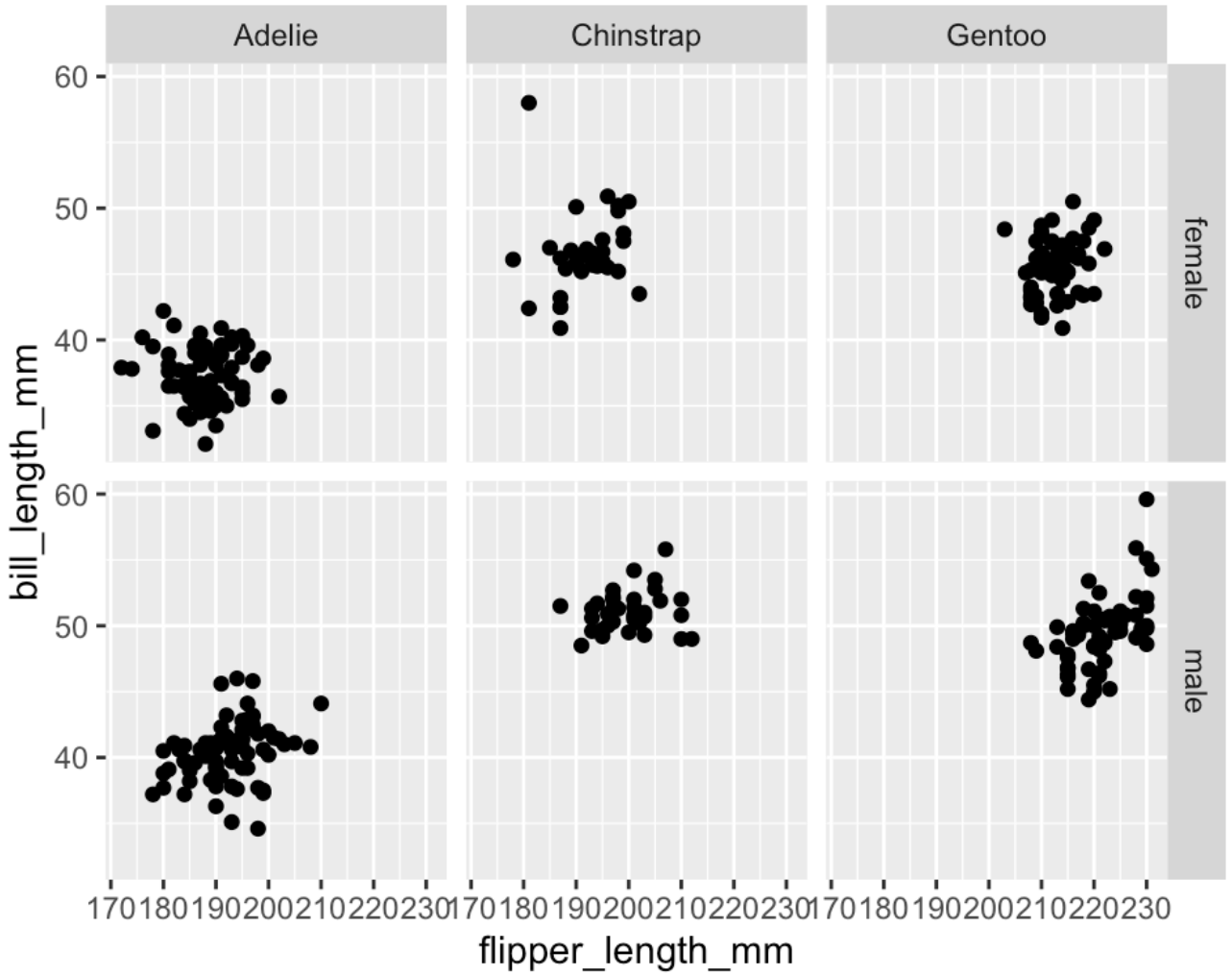
Facets


Adelie
Chinstrap
Gentoo
bill_length_mm
60
50
40
170
180
190
200
210
220
230
flipper_length_mm
species
Adelie
Chinstrap
Gentoo

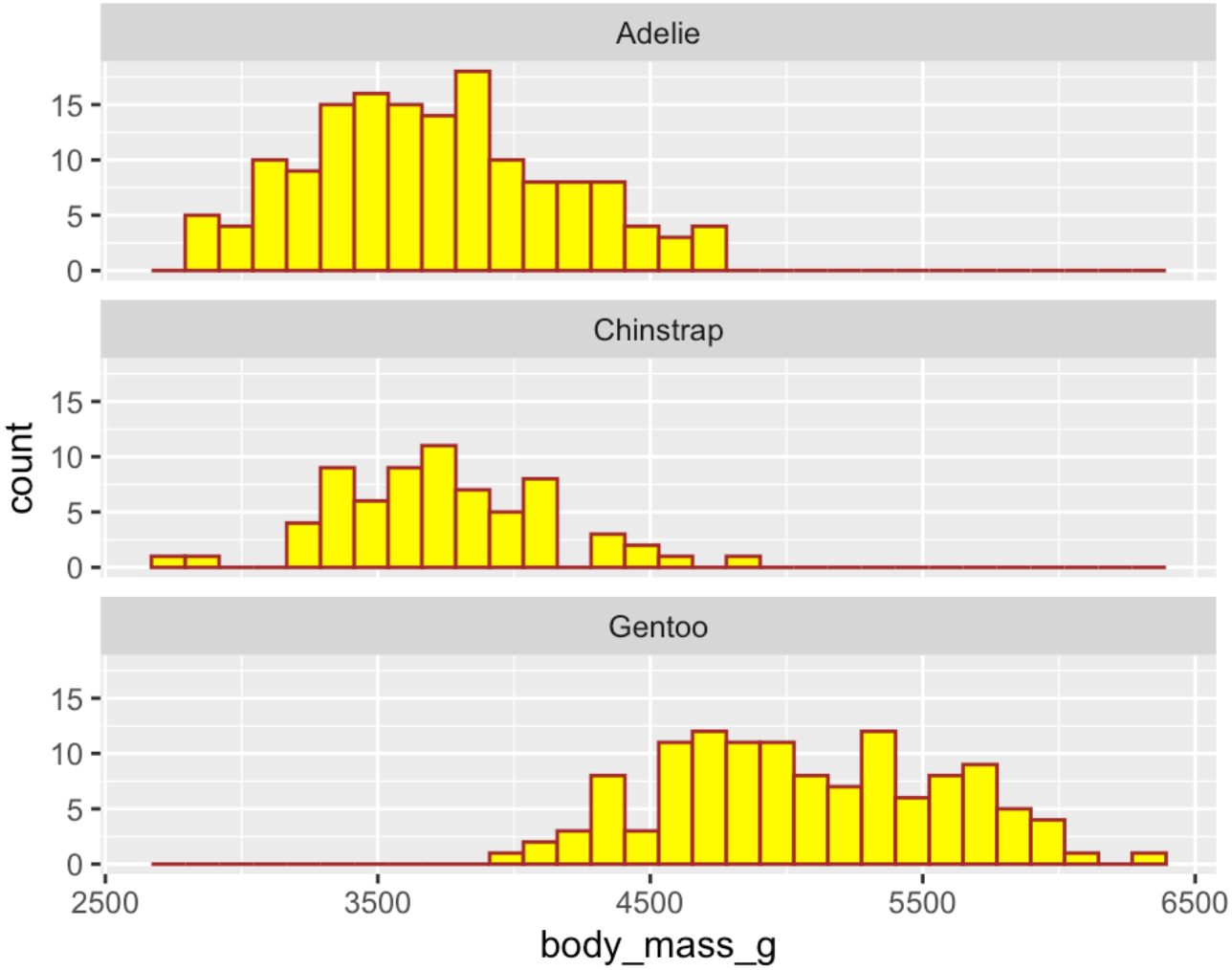
Facets


female
Adelie
female
Chinstrap
female
Gentoo
male
Adelie
male
Chinstrap
male
Gentoo
bill_length_mm
60
50
40
170 180 190 200 210 220 230
flipper_length_mm

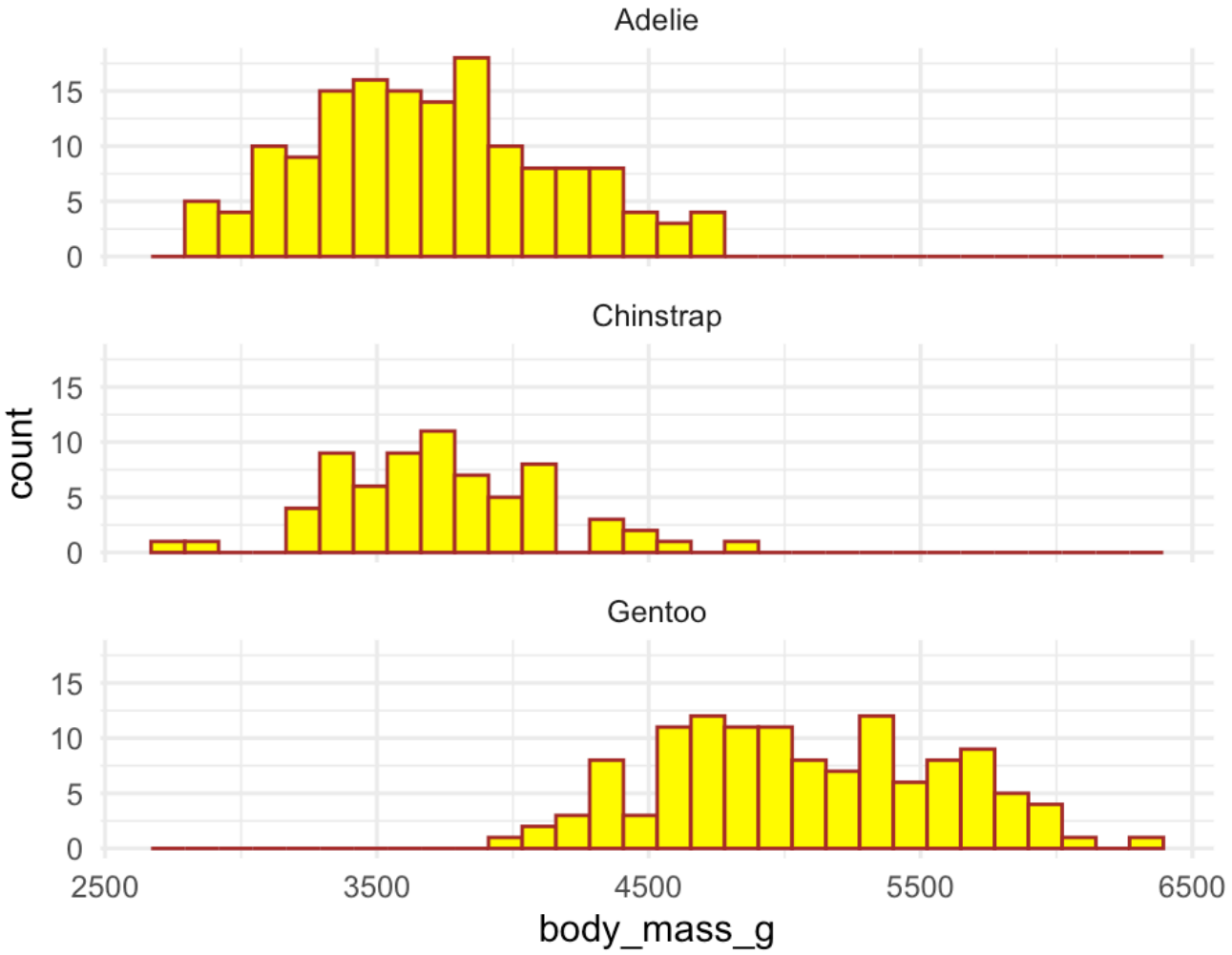
Facets


Adelie
Chinstrap
Gentoo
female
male
bill_length_mm
60
50
40
170
180
190
200
210
220
230
flipper_length_mm

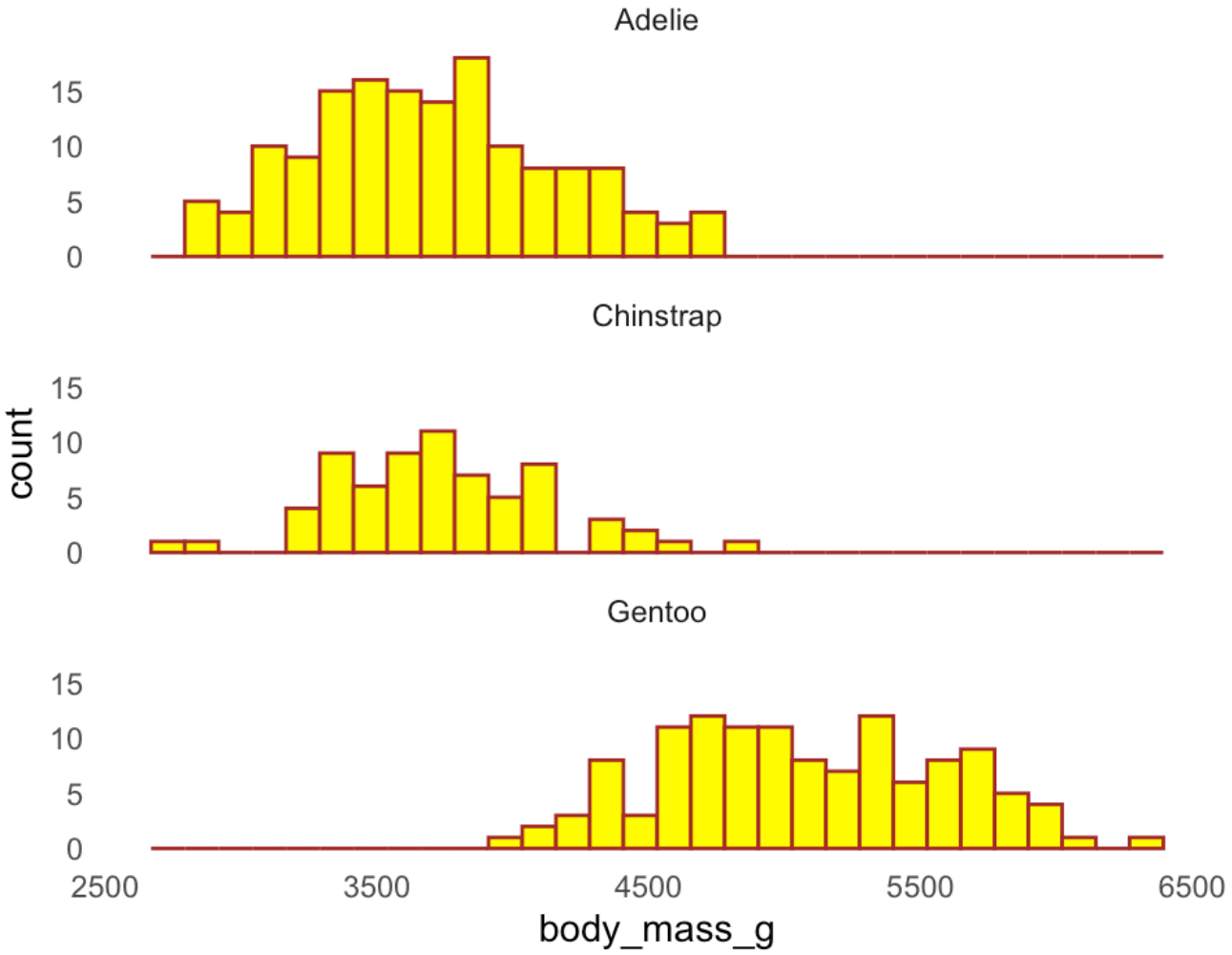
Facets


Adelie
Chinstrap
Gentoo
count
15
10
5
0
2500
3500
4500
5500
6500
body_mass_g

Facets


Adelie
Chinstrap
Gentoo
count
15
10
5
0
2500
3500
4500
5500
6500
body_mass_g

Facets


Adelie
Chinstrap
Gentoo
count
15
10
5
0
2500
3500
4500
5500
6500
body_mass_g

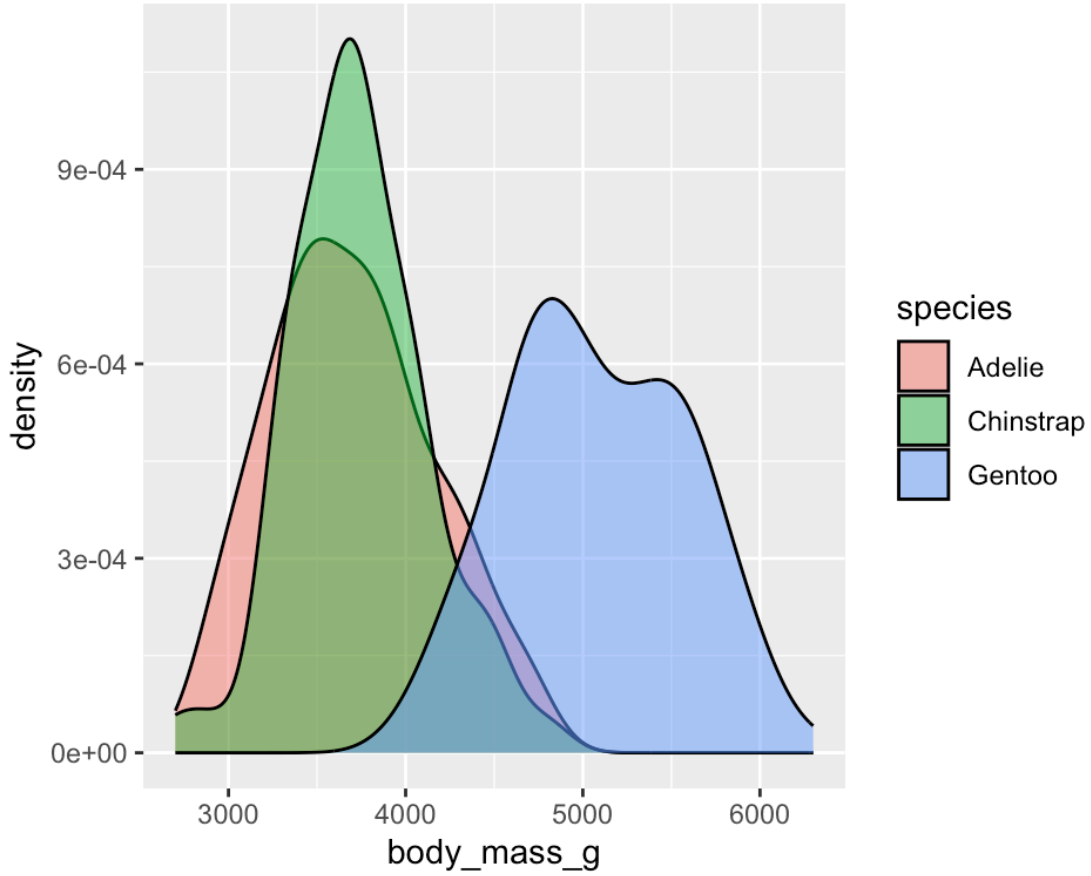
Exercise 3.2.4

(use `diamonds` data to answer the followings)

- Create histogram of `x` at different levels of `cut`

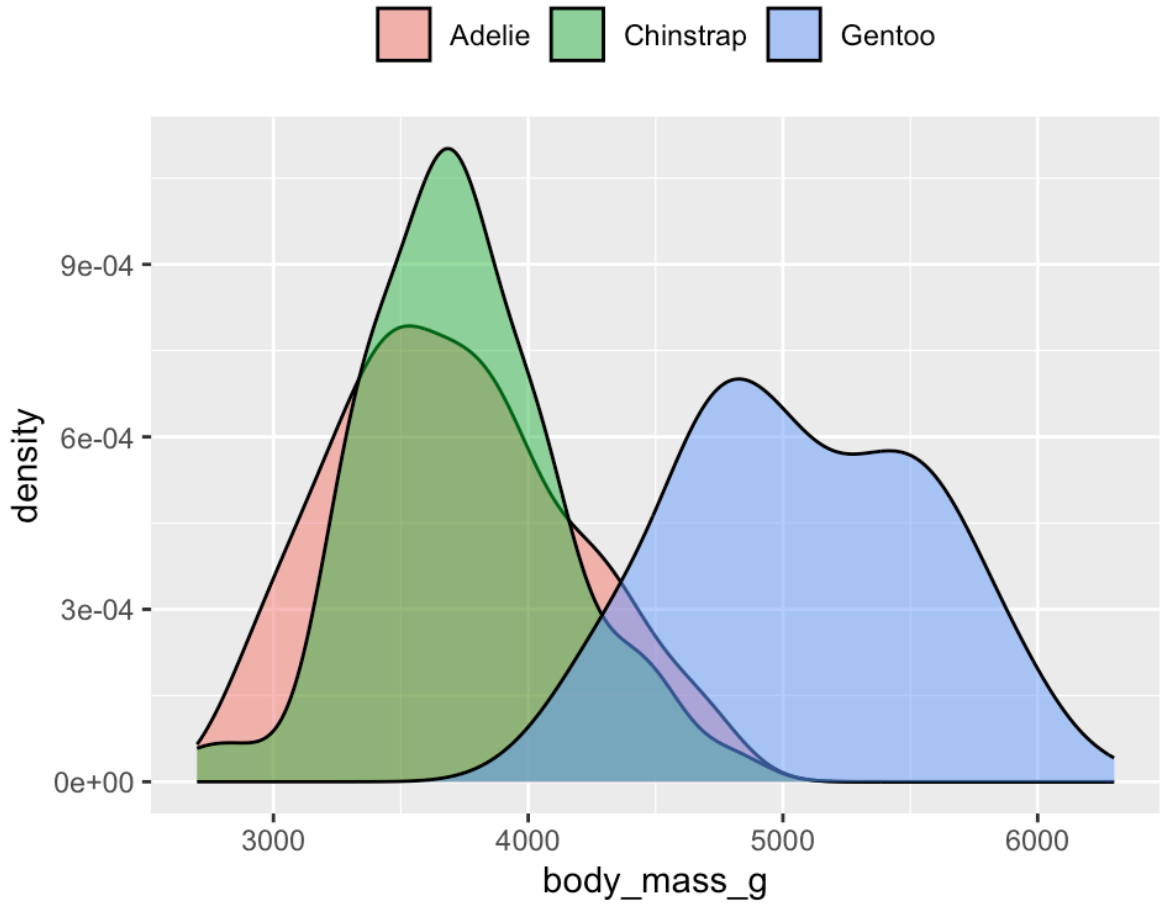
4 Density plot

Distribution of penguins' body mass



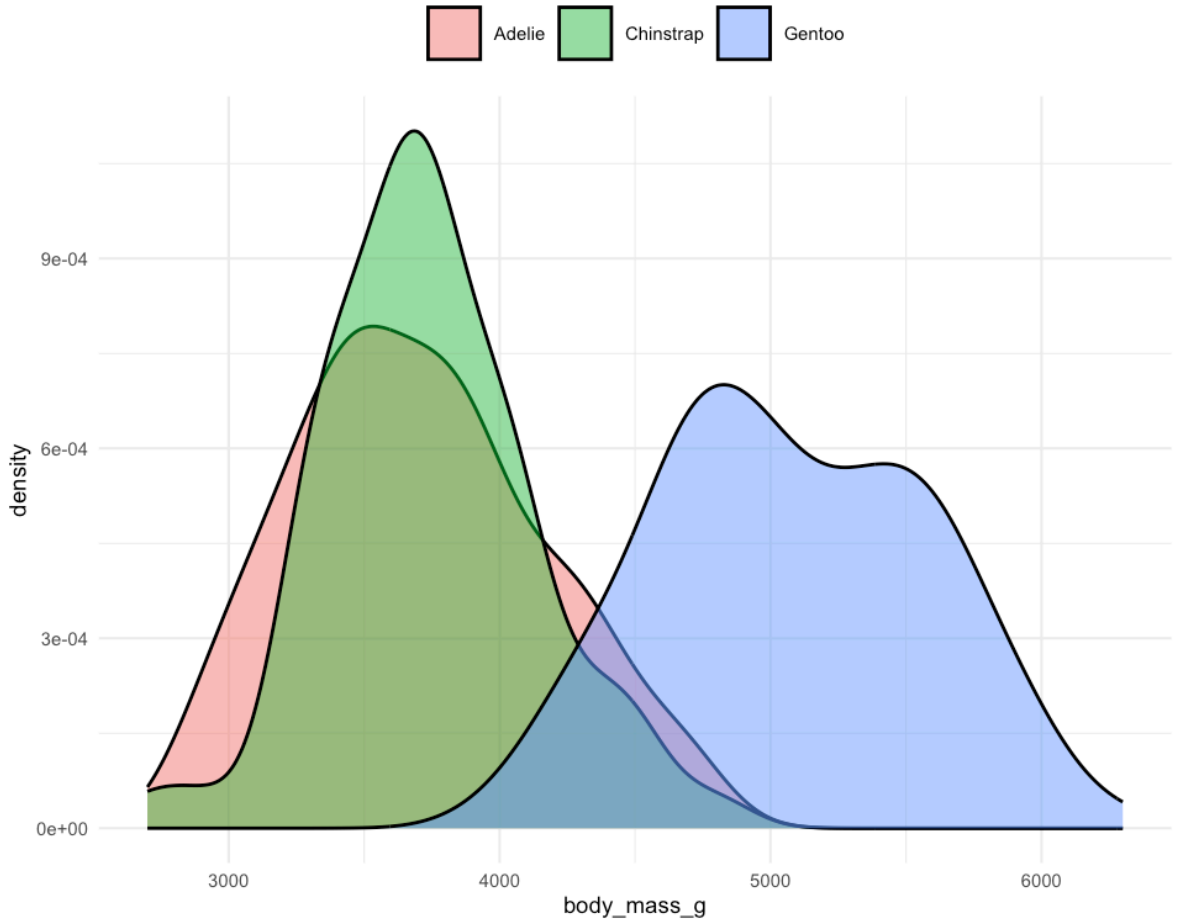
Density plot

Distribution of penguins' body mass



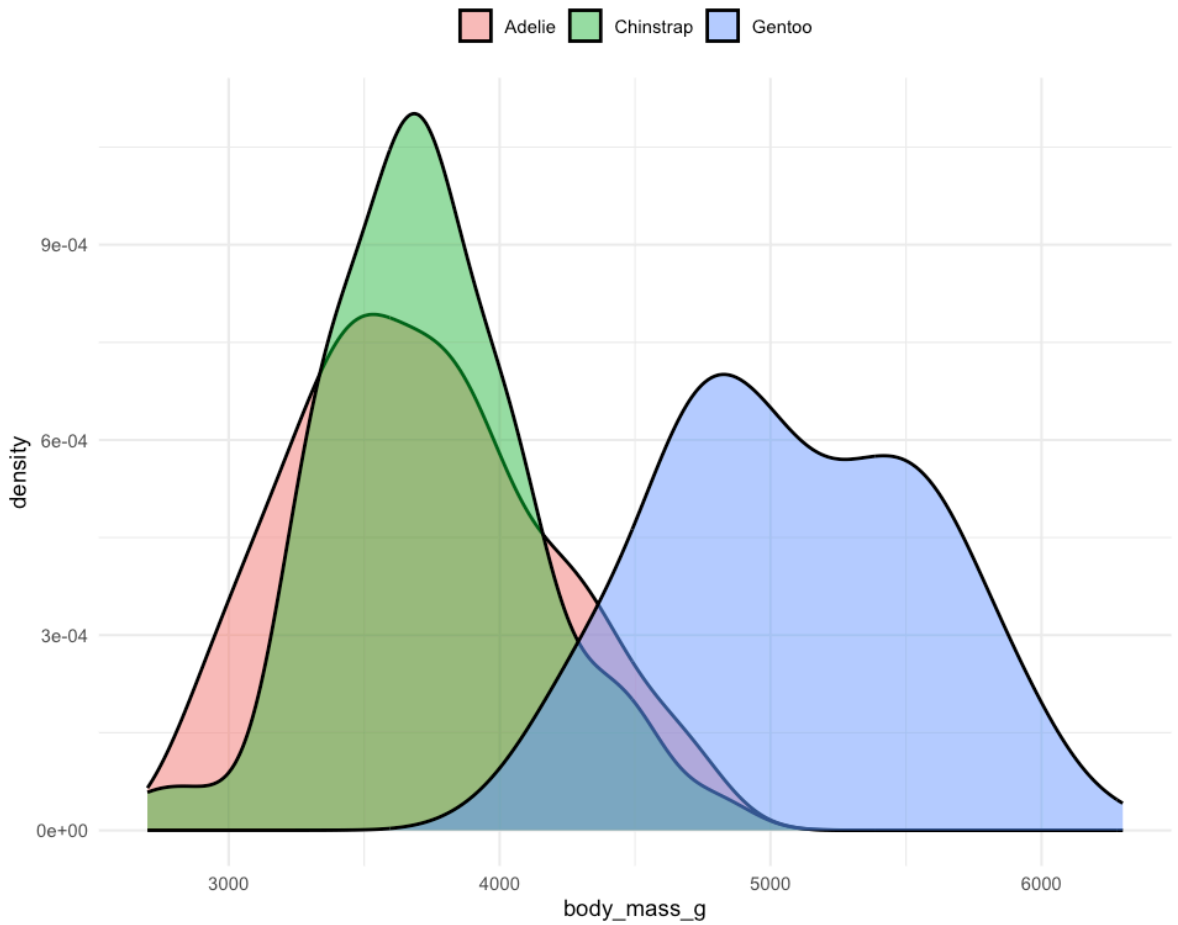
Density plot

Distribution of penguins' body mass



Density plot

Distribution of penguins' body mass



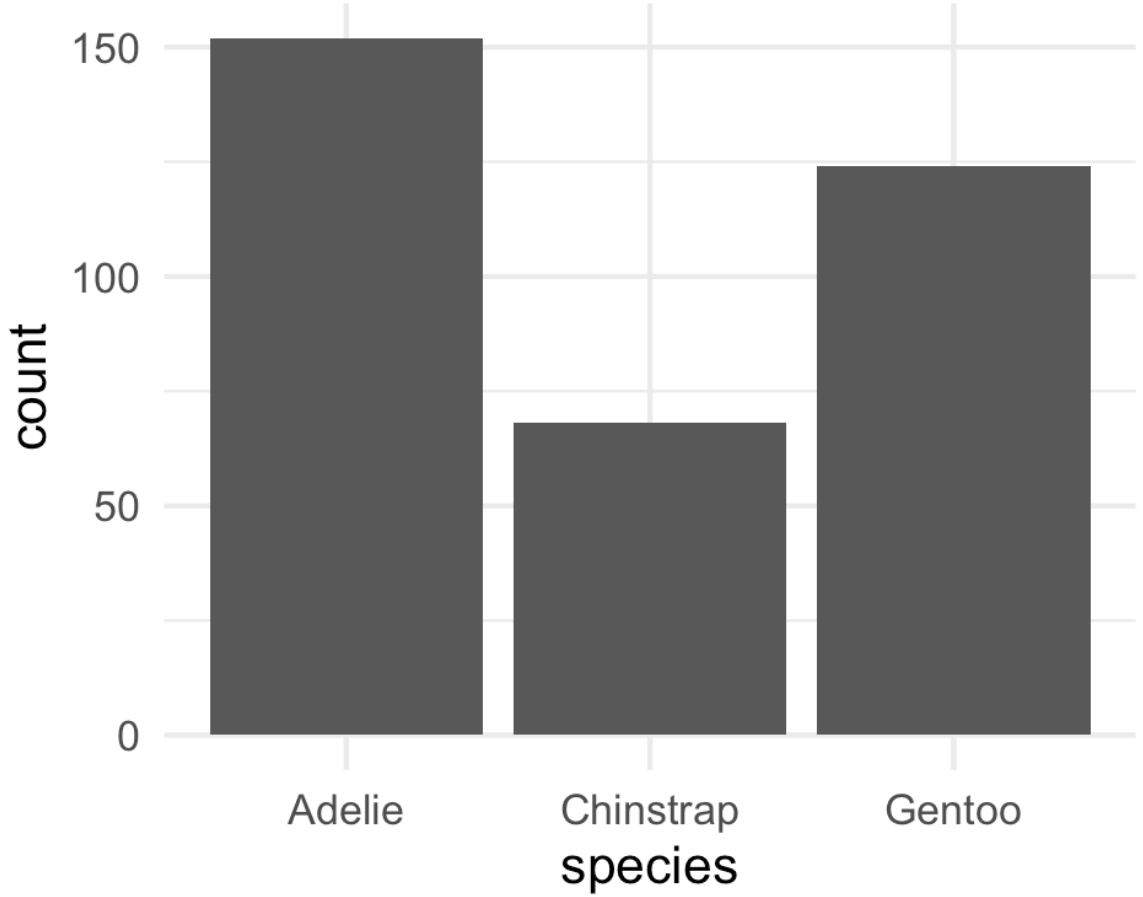
5 Barchart

Frequency distribution of `species`

```
ggplot(data = penguins) +
  geom_bar(aes(x = species))
```

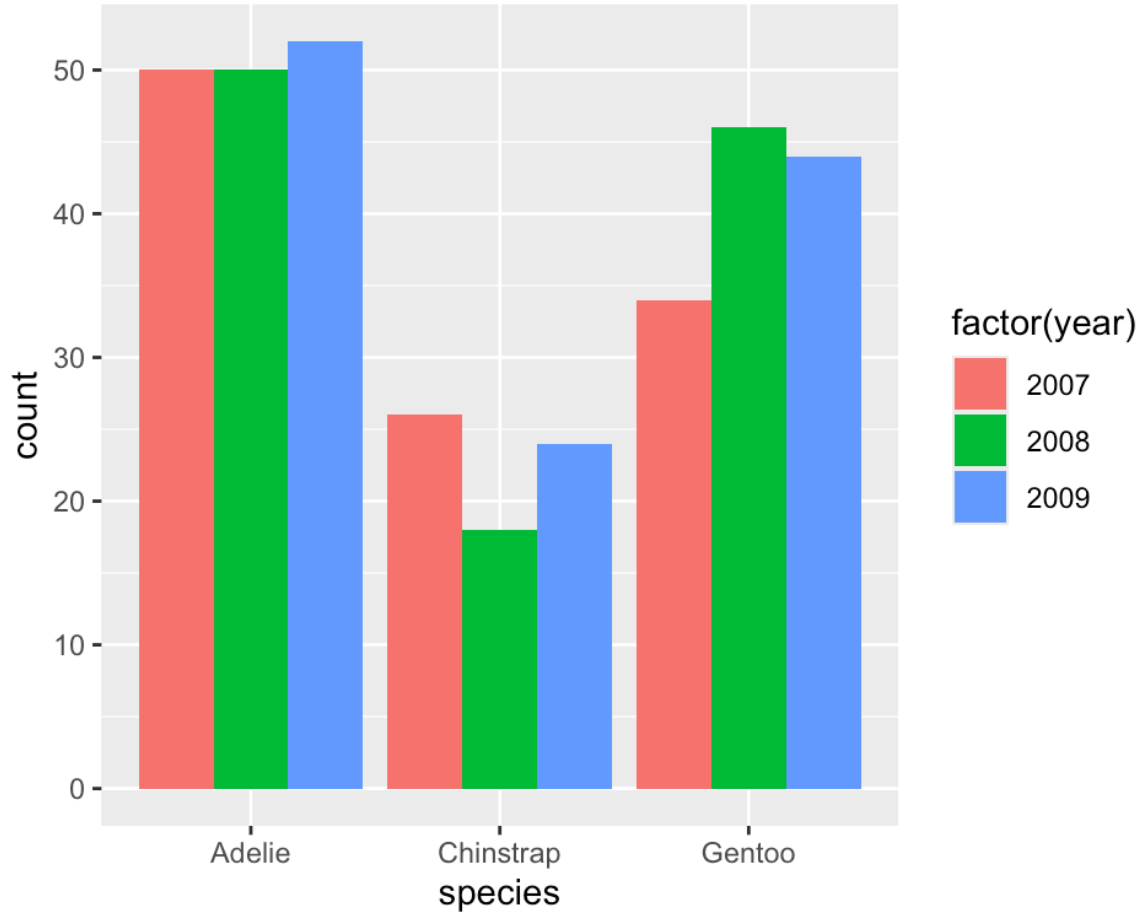
Density plot

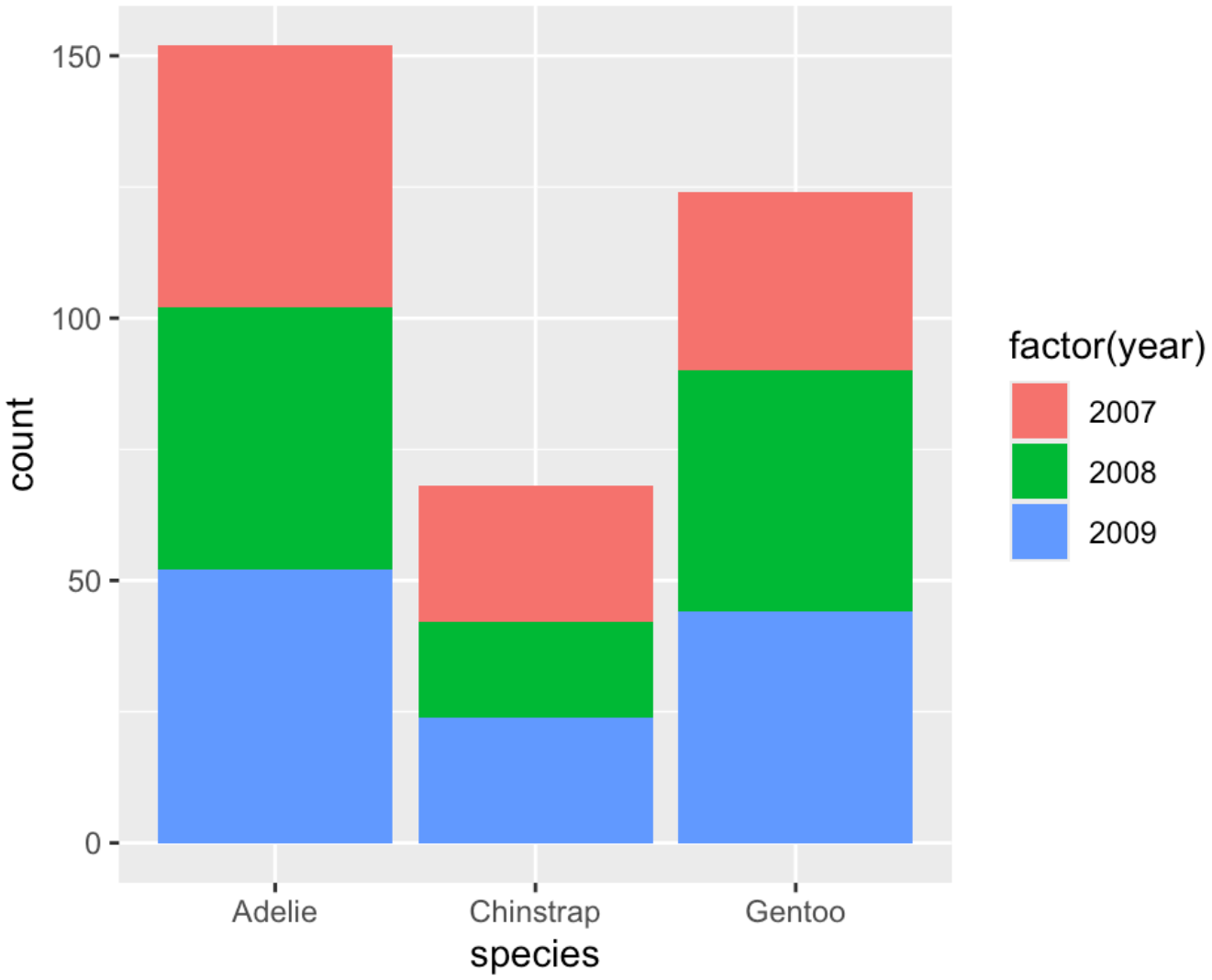
Frequency distribution of `species`

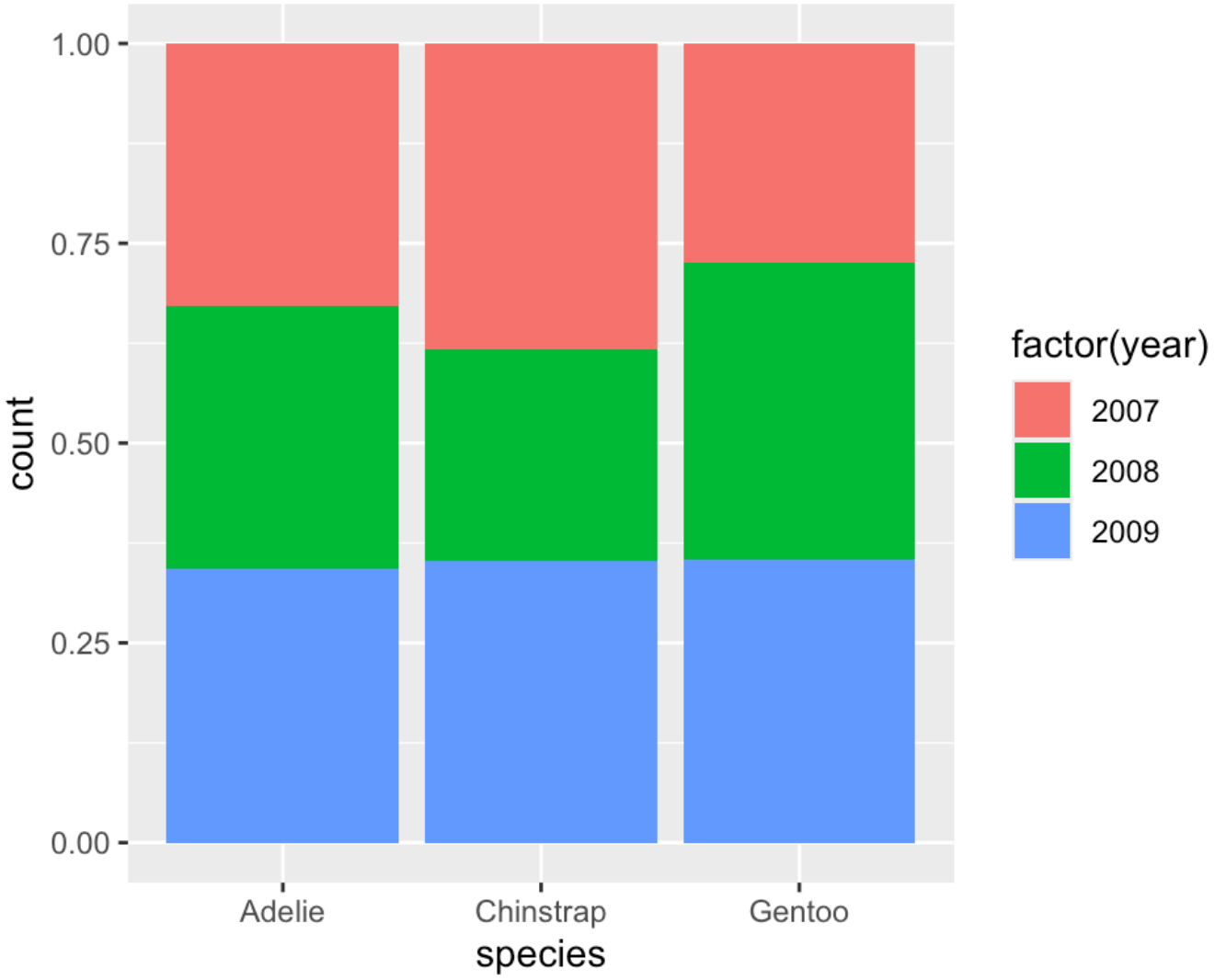


Barchart with two variables

Distribution of `species` by `year`




150
100
50
0
count
Adelie
Chinstrap
Gentoo
species
factor(year)
2007
2008
2009


1.00
0.75
0.50
0.25
0.00
count
Adelie
Chinstrap
Gentoo
species
factor(year)
2007
2008
2009

Exercise 3.2.5

(use `diamonds` data to answer the followings)

- Create a barplot of `cut`
- Create a barplot of `color`
- Create a barplot of `cut` with showing the distribution of `color` at different levels of `cut`
- Check the use three different value of the argument `position` when creating a barplot with `cut` and `color`

Homework

- Use the package `gapminder` to get an access to the data `gapminder`
- `gapminder` has 6 variables and 1704 observations, where the variables are:

```
#> [1] "country"   "continent" "year"      "lifeExp"   "pop"       "gdpPercap"
```

- Create a scatter plot to examine how `gdpPercap` affects `lifeExp`
- Change the scale of x-axis to log base 10
- Add a color layer corresponding to `continent` to the previous graph

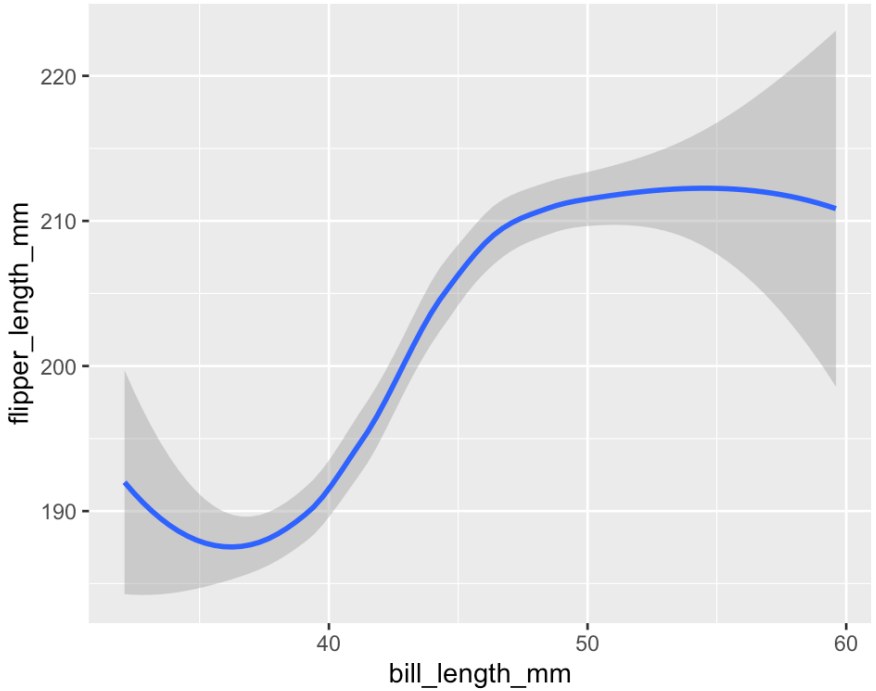
Homework

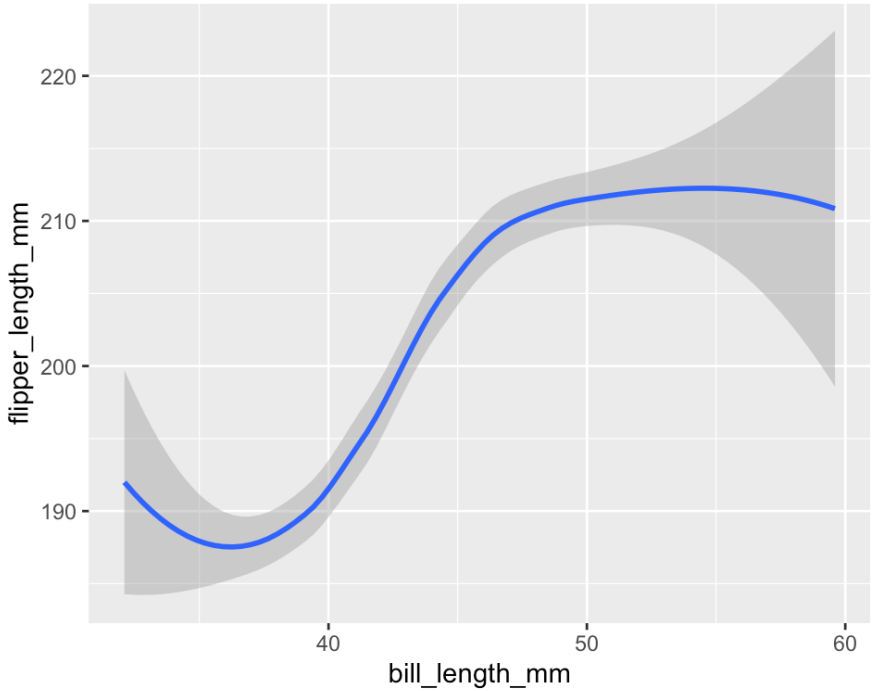
- Create a scatter plot of `gdpPercap` versus `lifeExp` for different continents in different plotting regions
- Add smooth lines to describe relationship between `gdpPercap` and `lifeExp` for different continents separately
- Draw a boxplot of `lifeExp` to compare distribution life expectancy for different continents
- Draw a histogram of `lifeExp` and check it shapes for different bin size
- Draw density plots of `lifeExp` for different continents in a single plot

Homework

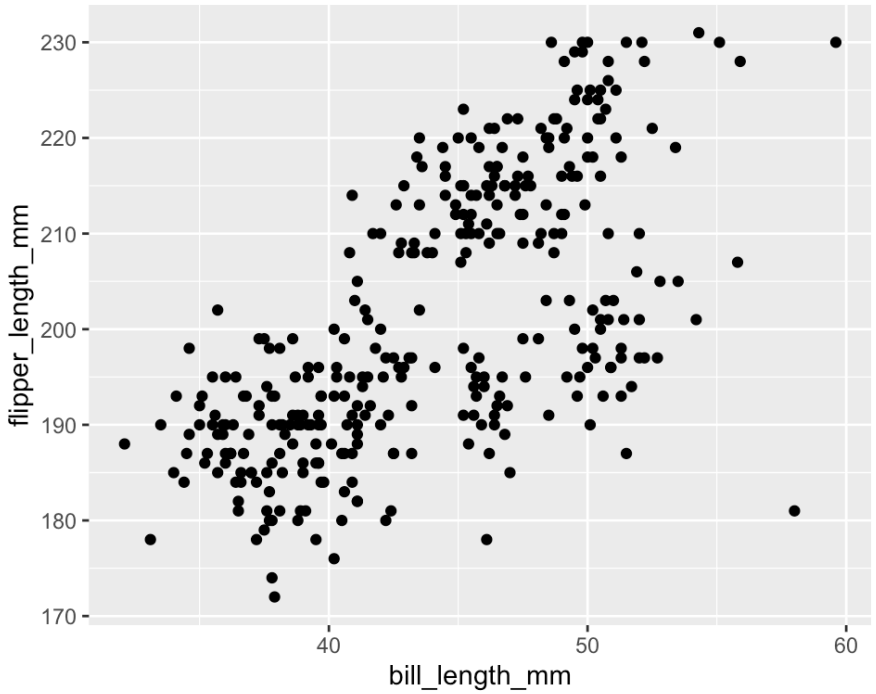
- Make a scatter plot of `lifeExp` on the y-axis against `year` on the `x`
- Fit a straight line to estimate mean life expectancy for a year for different countries
- Split the plot for different continents
- Add a continent-specific mean line to the plot

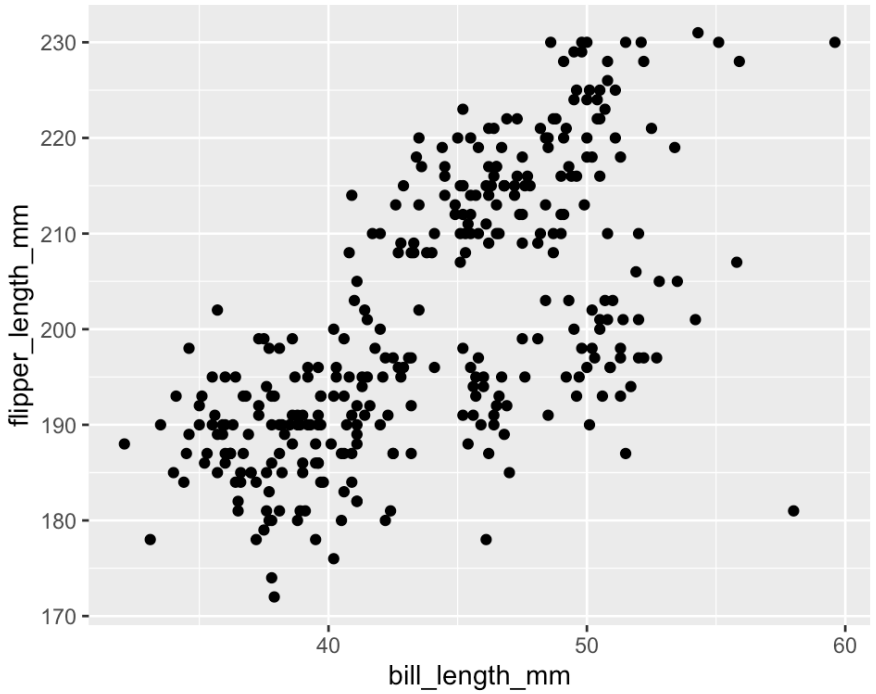
Statistical layers `geom_*()` vs `stat_*()`



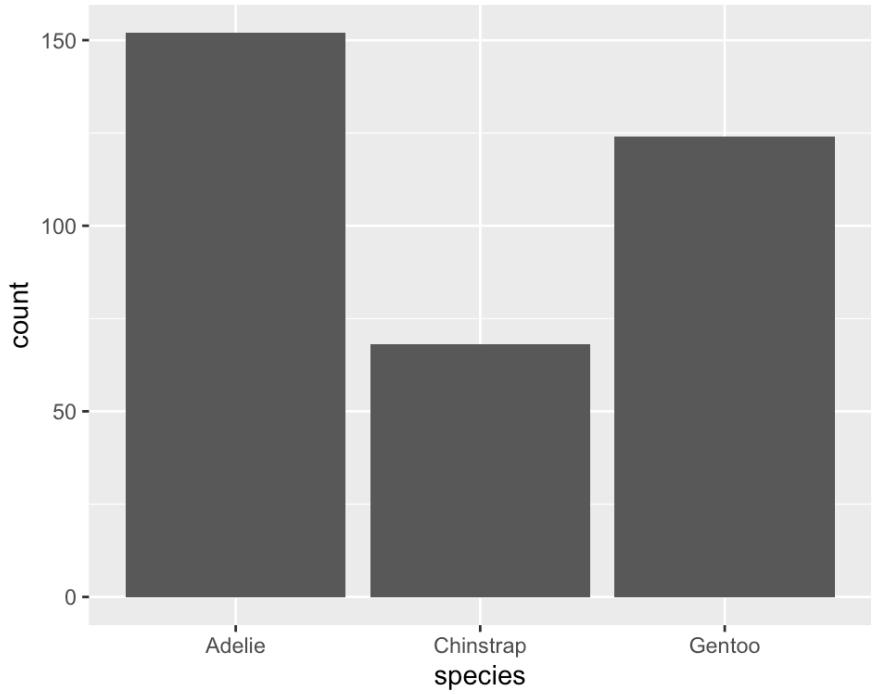


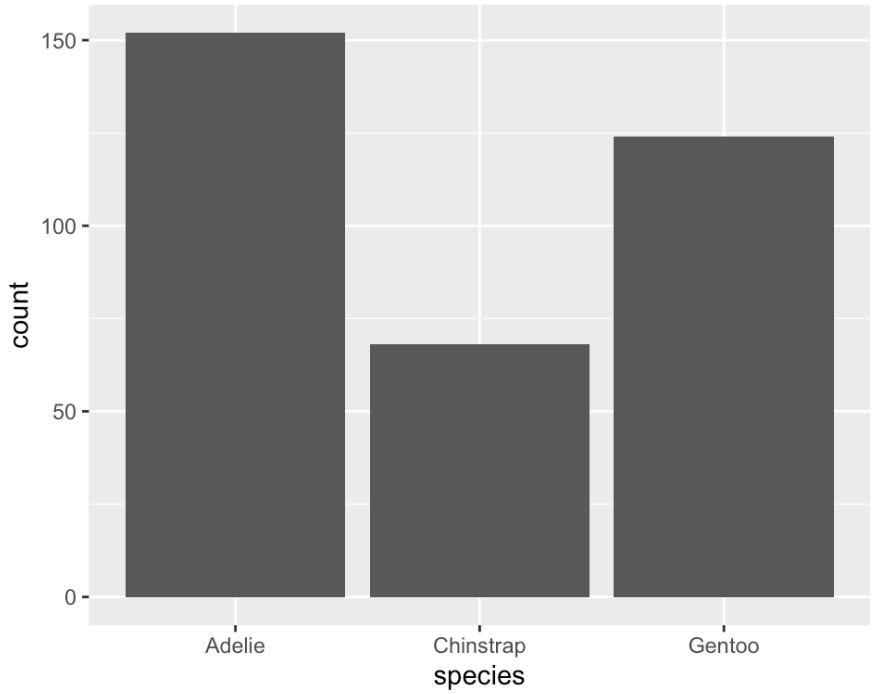
Homework


230
220
210
200
190
180
170
flipper_length_mm
40
50
60
bill_length_mm

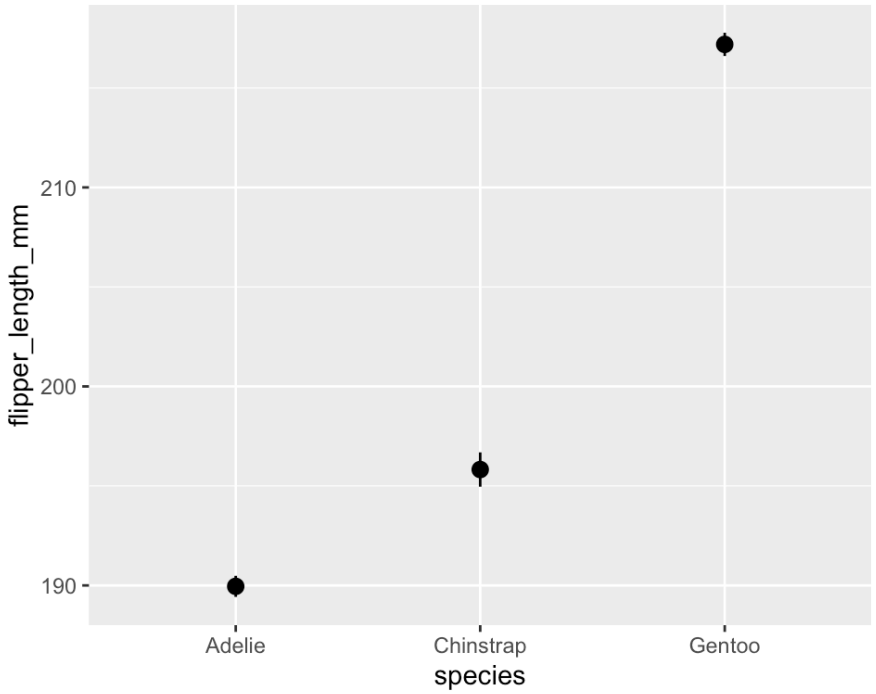

230
220
210
200
190
180
170
flipper_length_mm
40
50
60
bill_length_mm

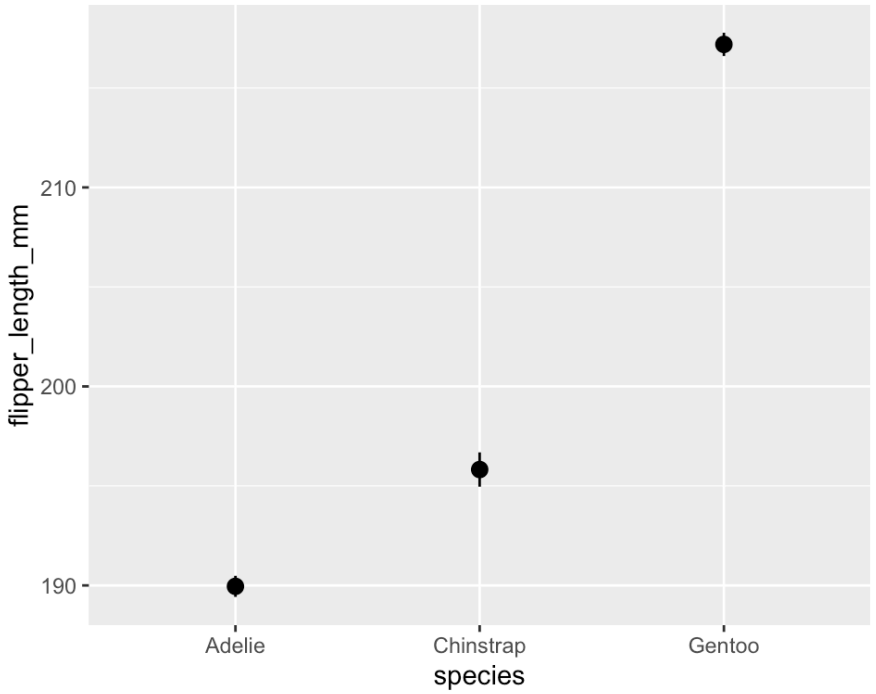
Homework


150
100
50
0
count
Adelie
Chinstrap
Gentoo
species

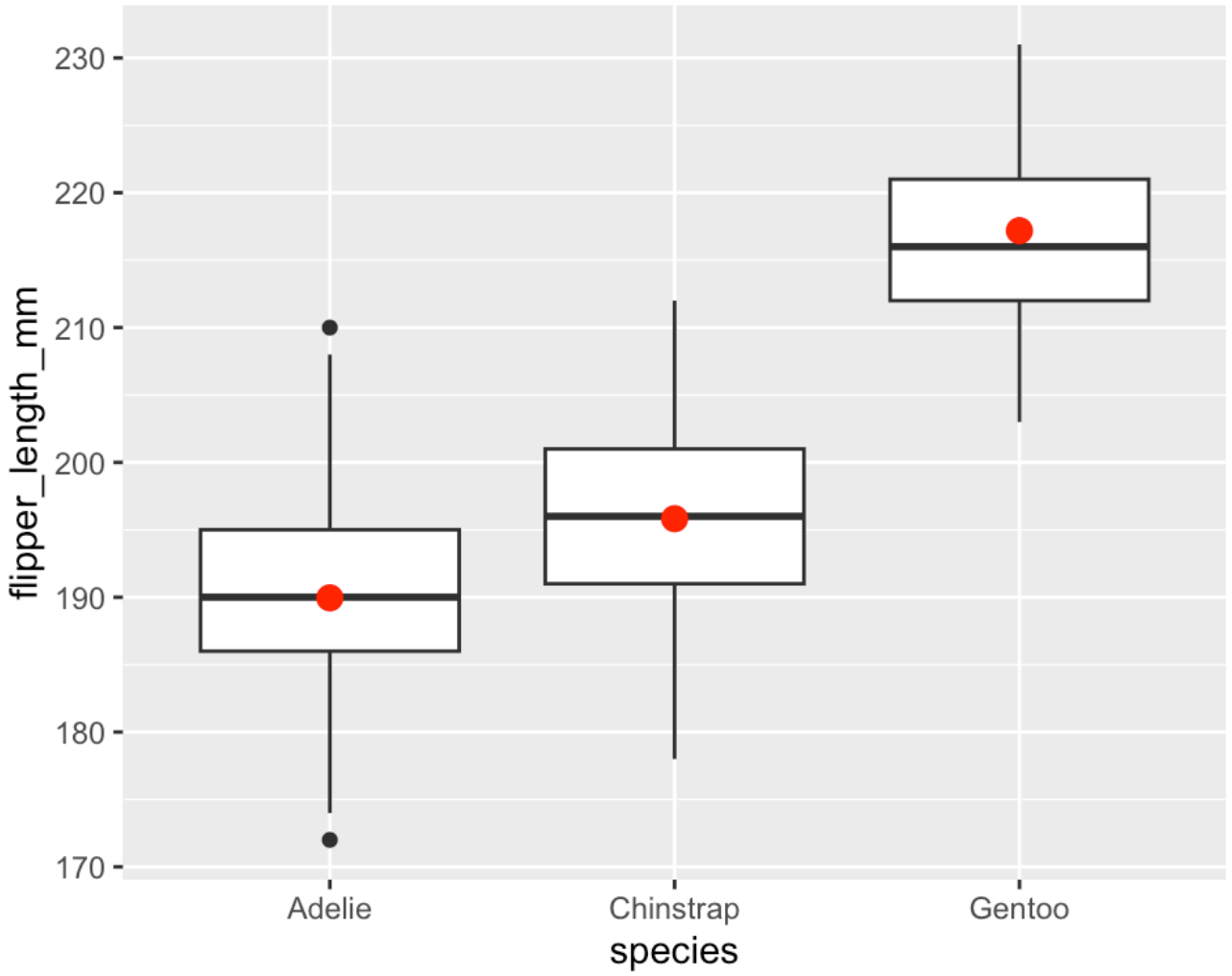

150
100
50
0
count
Adelie
Chinstrap
Gentoo
species

Statistical summaries





Statistical summaries



Statistical summaries