

Data Visualization

Part 3

numeric by default, but you can force it to become categorical.



Data 1 - mtcars

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3

Data 2 - penguins

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex	year
1	Adelie	Torgersen	39.1	18.7	181	3750	male	2007
2	Adelie	Torgersen	39.5	17.4	186	3800	female	2007
3	Adelie	Torgersen	40.3	18.0	195	3250	female	2007
4	Adelie	Torgersen	NA	NA	NA	NA	NA	2007
5	Adelie	Torgersen	36.7	19.3	193	3450	female	2007
6	Adelie	Torgersen	39.3	20.6	190	3650	male	2007
7	Adelie	Torgersen	38.9	17.8	181	3625	female	2007
8	Adelie	Torgersen	39.2	19.6	195	4675	male	2007
9	Adelie	Torgersen	34.1	18.1	193	3475	NA	2007
10	Adelie	Torgersen	42.0	20.2	190	4250	NA	2007
11	Adelie	Torgersen	37.8	17.1	186	3300	NA	2007
12	Adelie	Torgersen	37.8	17.3	180	3700	NA	2007
13	Adelie	Torgersen	41.1	17.6	182	3200	female	2007
14	Adelie	Torgersen	38.6	21.2	191	3800	male	2007

Changing Colors in Plots

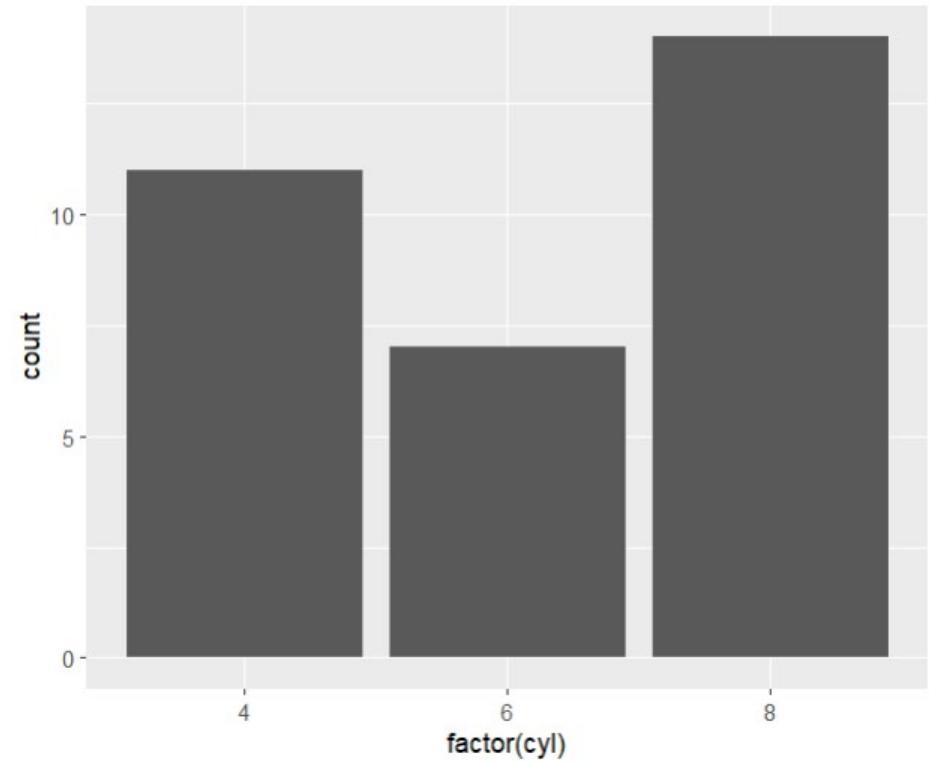
Univariate Displays

Example 1: Univariate simple bar graph (no color)

```
fig_1 <- mtcars %>%  
  ggplot(aes(x = factor(cyl))) +  
  geom_bar()  
fig_1
```

↓
forces cyl to
become a factor
variable.

* See what happens if
you don't change it to
factor.

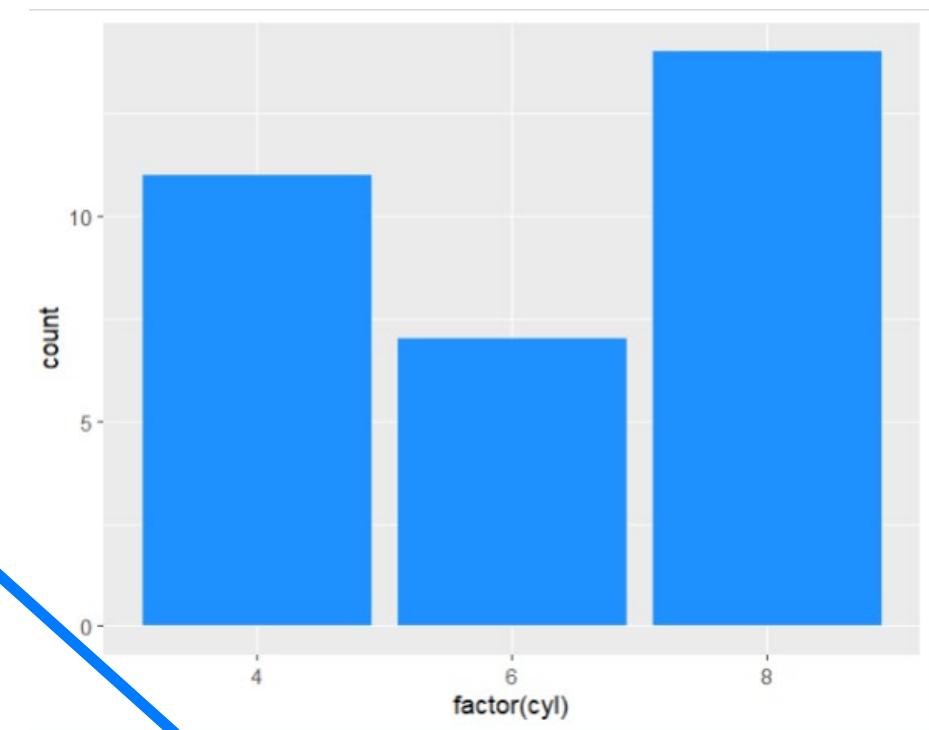


Example 2: Fills all the bars with the same color.

```
fig_2 <- mtcars %>%  
  ggplot(aes(x = factor(cyl))) +  
  geom_bar(fill = "dodgerblue")
```

```
fig_2
```

↓
notice that fill is
NOT inside aes.



changes all the
colors of the
bars to the same
colors.

We saw this code in part 2 of the notes.

Example 3: Change each bar to a different color using fill inside "aes"

```
fig_3 <- mtcars %>%
```

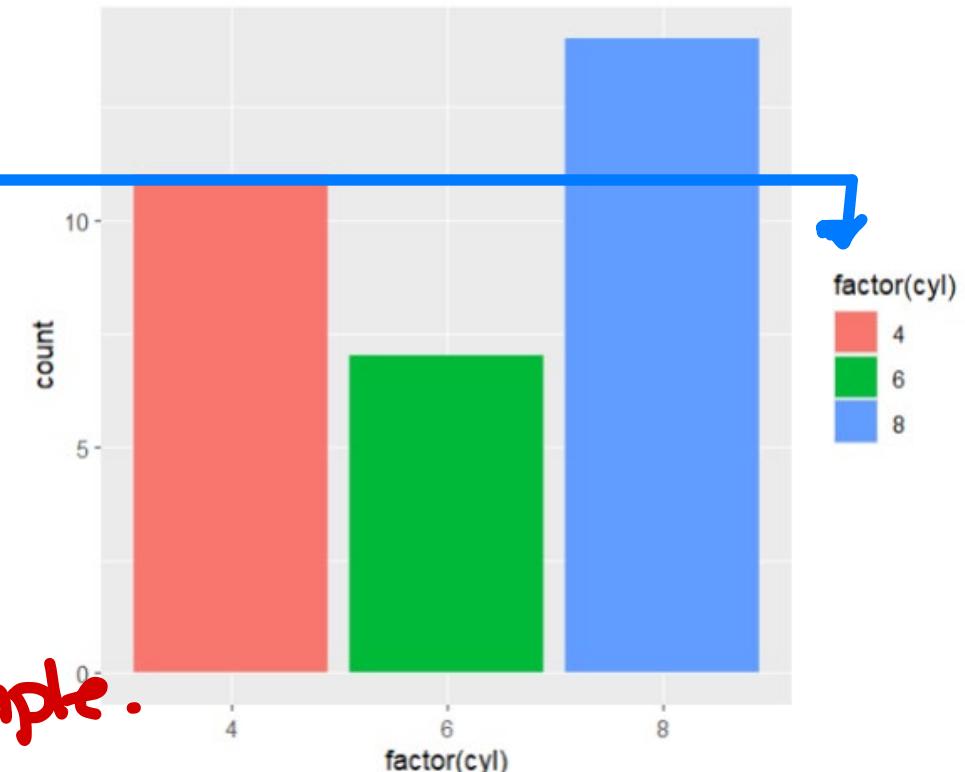
```
  ggplot(aes(x = factor(cyl),  
             fill = factor(cyl))) +
```

```
  geom_bar()
```

```
fig_3
```

This creates the legend.

Same variable.



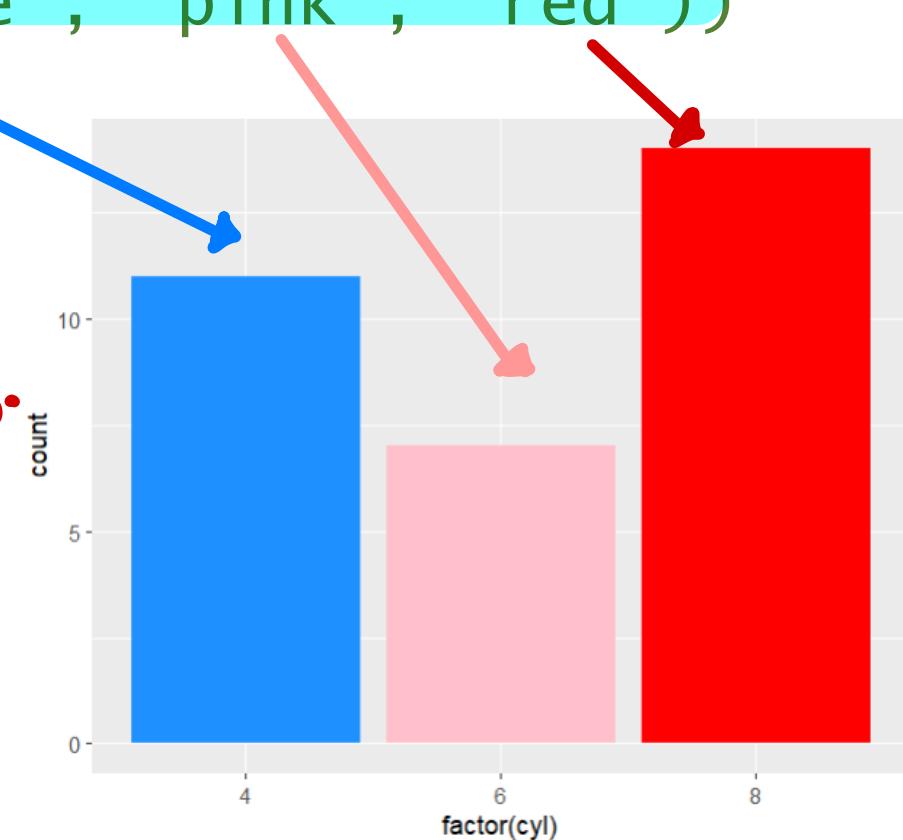
What if I want a different color for each bar, but I don't want a legend? See next example.

Example 4: Change each bar to a different color without the legend.

```
fig_4 <- mtcars %>%  
  ggplot(aes(x = factor(cyl))) +  
  geom_bar(fill = c("dodgerblue", "pink", "red"))
```

```
fig_4
```

Put the colors in the order you want them, and as many colors as categories. *If the # of colors doesn't match the # of categories, it will "crash".

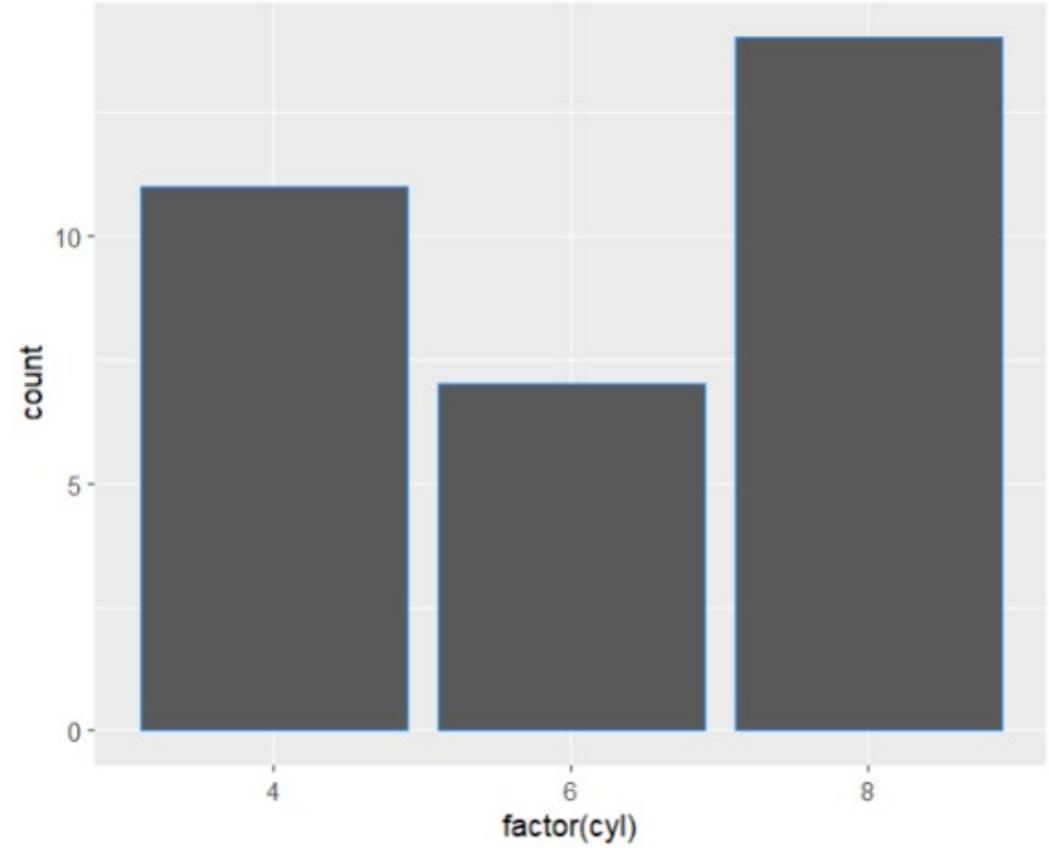


Example 5: Attempt to change bar color using color instead of fill.
What happens?

```
fig_5 <- mtcars %>%  
  ggplot(aes(x = factor(cyl))) +  
  geom_bar(color = "dodgerblue")
```

fig_5

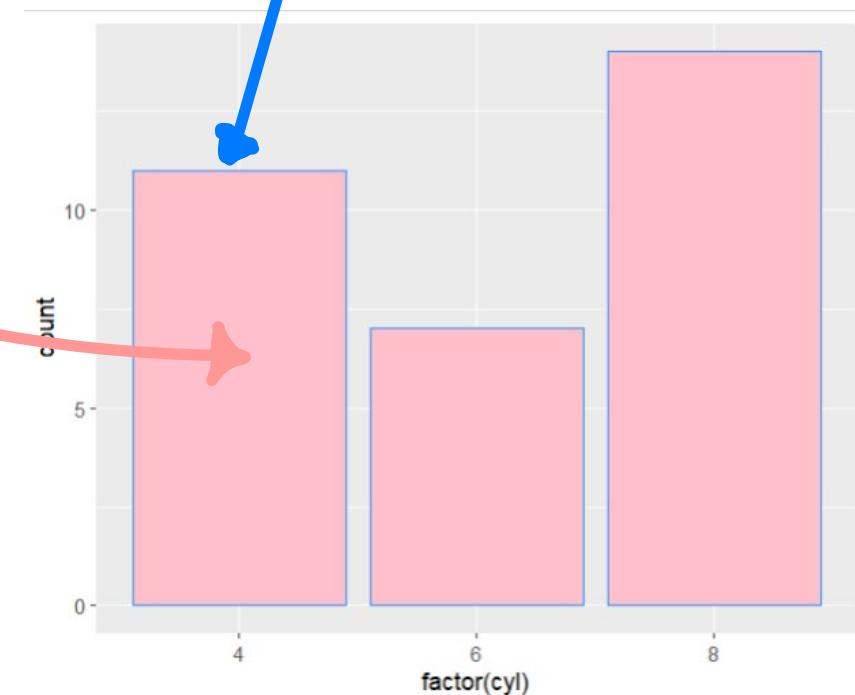
↓
"color" changes the
color of the border lines.
"fill" changes the color of
the inside of the bar.



Example 6: Change the lines of the bars to dodgerblue and the bar colors to pink.

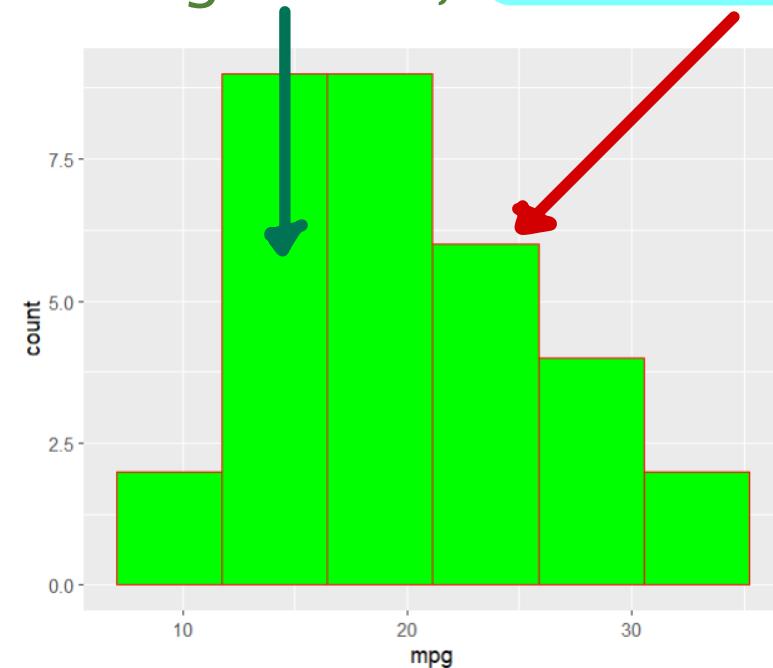
```
fig_6 <- mtcars %>%  
  ggplot(aes(x = factor(cyl))) +  
  geom_bar(fill = "pink", color = "dodgerblue")  
fig_6
```

* How would you write the codes such that each bar is filled with a different color and each borderline has a different color as well?



Example 7: Create a histogram for mpg variable with 6 bins. Then, change the lines of the histogram to red and the bar colors to green.

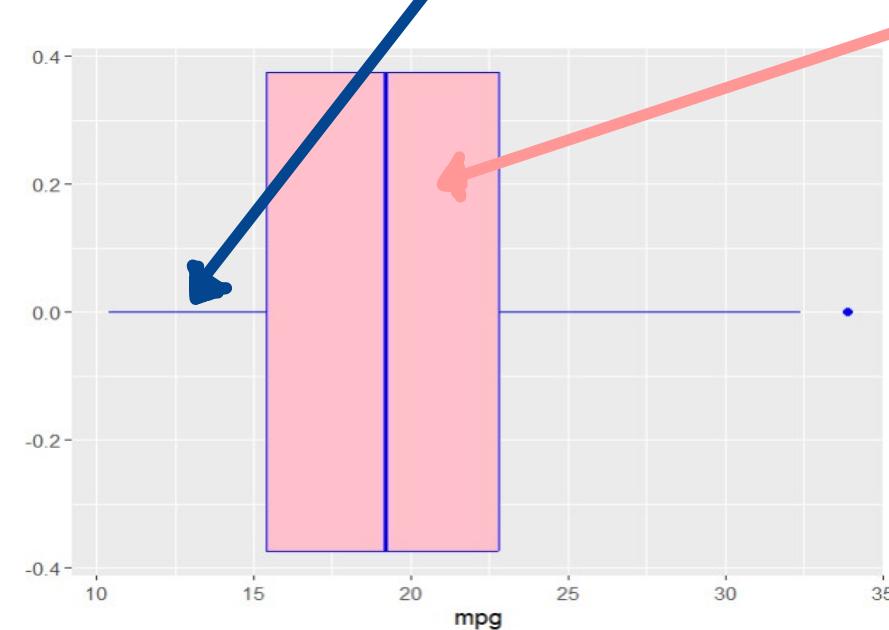
```
fig_7 <- mtcars %>%  
  ggplot(aes(x = mpg)) +  
  geom_histogram(bins = 6, fill = "green", color = "red")  
fig_7
```



Example 8: Create a boxplot for mpg variable. Then, change the lines of the plot to blue and the inside colors to pink.

```
fig_8 <- mtcars %>%  
  ggplot(aes(x = mpg)) +  
  geom_boxplot(color = "blue", fill = "pink")
```

```
fig_8
```



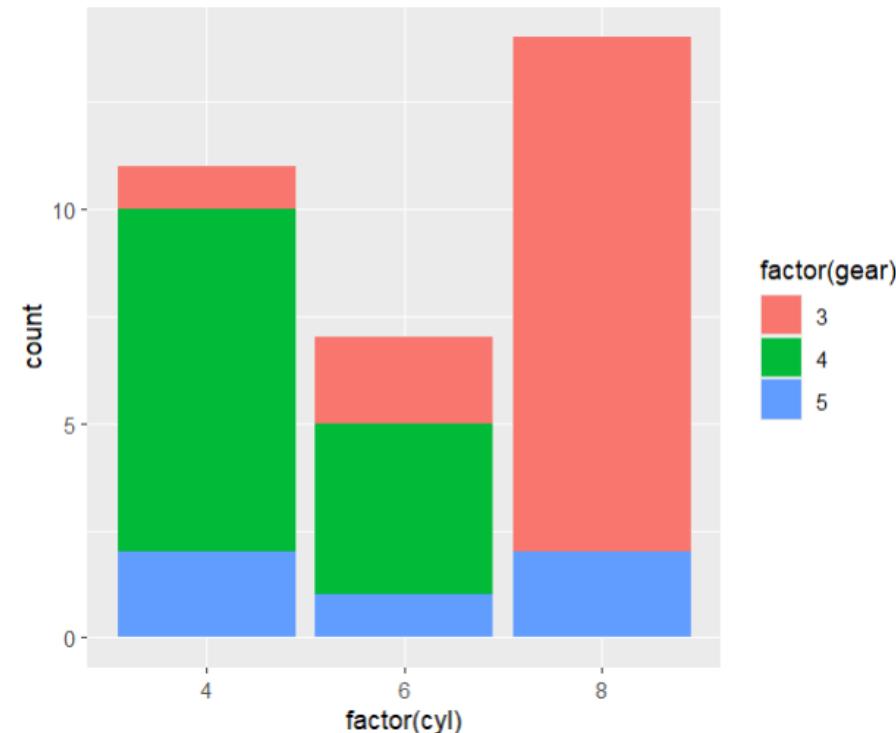
Multivariate Displays

- `scale_fill_manual()`: manually changes fill in the aes.
- `scale_color_manual()`: manually changes color in the aes.

Example 9: Create a stacked bar graph with cyl in the x axis and gear as the categorical variable to split each column.

```
fig_9 <- mtcars %>%  
  ggplot(aes(x = factor(cyl), fill = factor(gear)))+  
  geom_bar()
```

```
fig_9
```

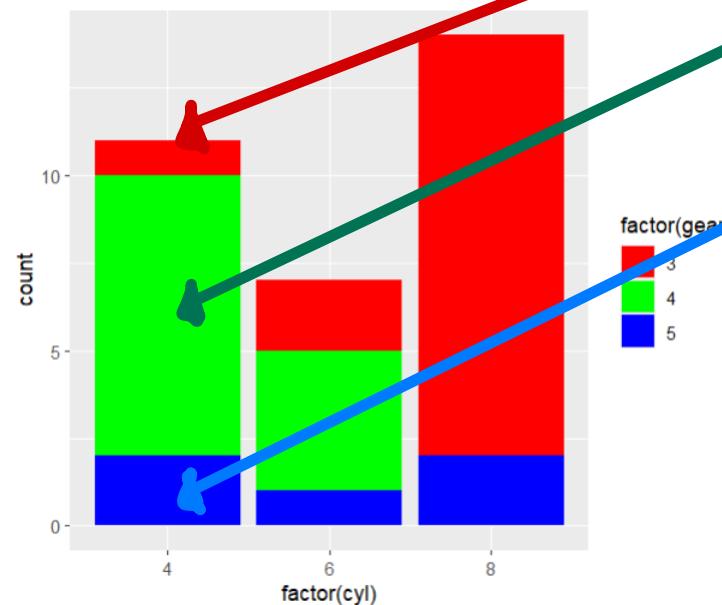


We learned this in our previous class.
What if I want to change those colors?

Example 10: Change the colors from Example 9 to red, green, and blue (in that order).

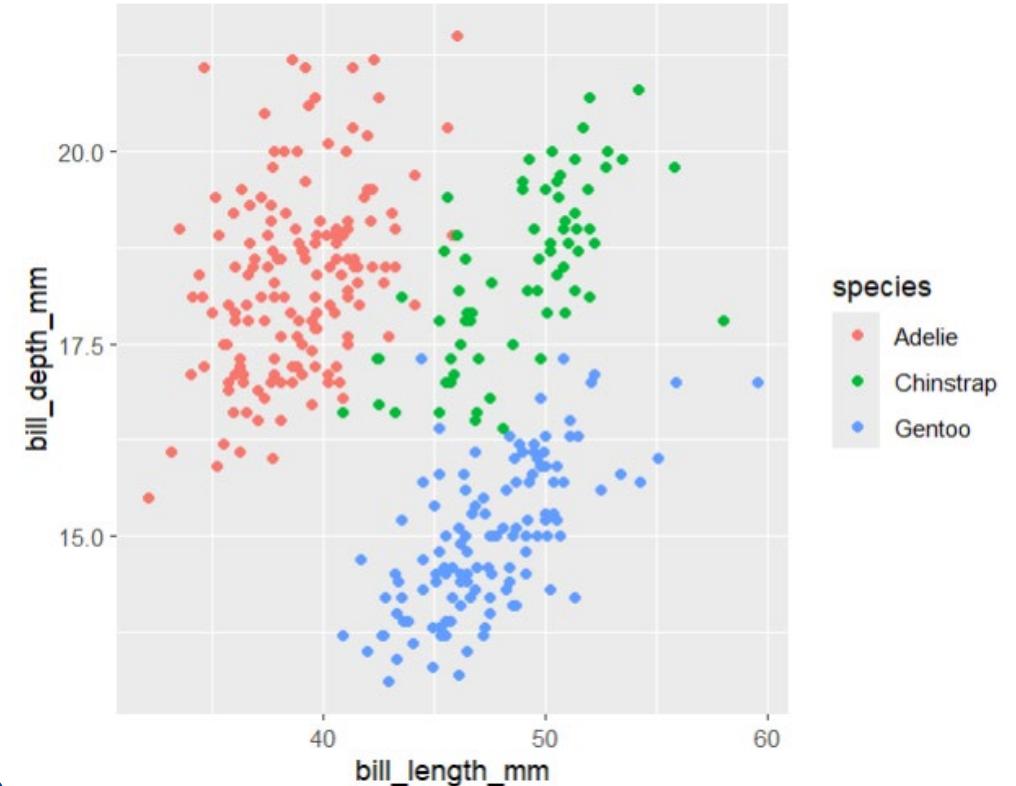
some from defal

```
fig_10 <- mtcars %>%
  ggplot(aes(x = factor(cyl), fill = factor(gear)))+
  geom_bar()+
  scale_fill_manual(values = c("red", "green", "blue"))
fig_10
```



Example 11: Create a scatter plot with $x = \text{bill_length_mm}$, $y = \text{bill_depth_mm}$, and species as the categorical variable.

```
fig_11 <- penguins %>%
  ggplot(aes(x = bill_length_mm,
             y = bill_depth_mm,
             color = species)) +
  geom_point()
fig_11
```



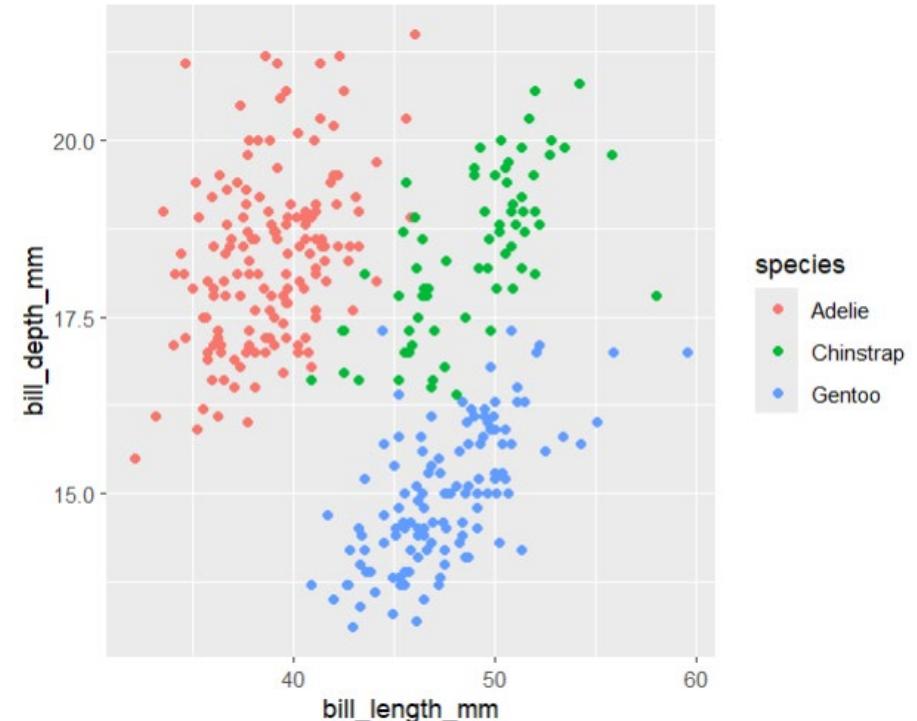
We learned this in our previous class.
What if I want to change those colors?

Example 12: Change the colors from Example 11 to red, green, and blue, in that order. What is wrong with these codes? Why isn't this one working?

```
fig_12 <- penguins %>%  
  ggplot(aes(x = bill_length_mm,  
             y = bill_depth_mm,  
             Used color here → color = species)) +  
  geom_point() +  
  scale_fill_manual(values = c("red", "green", "blue"))
```

fig_12
Used fill here ↑

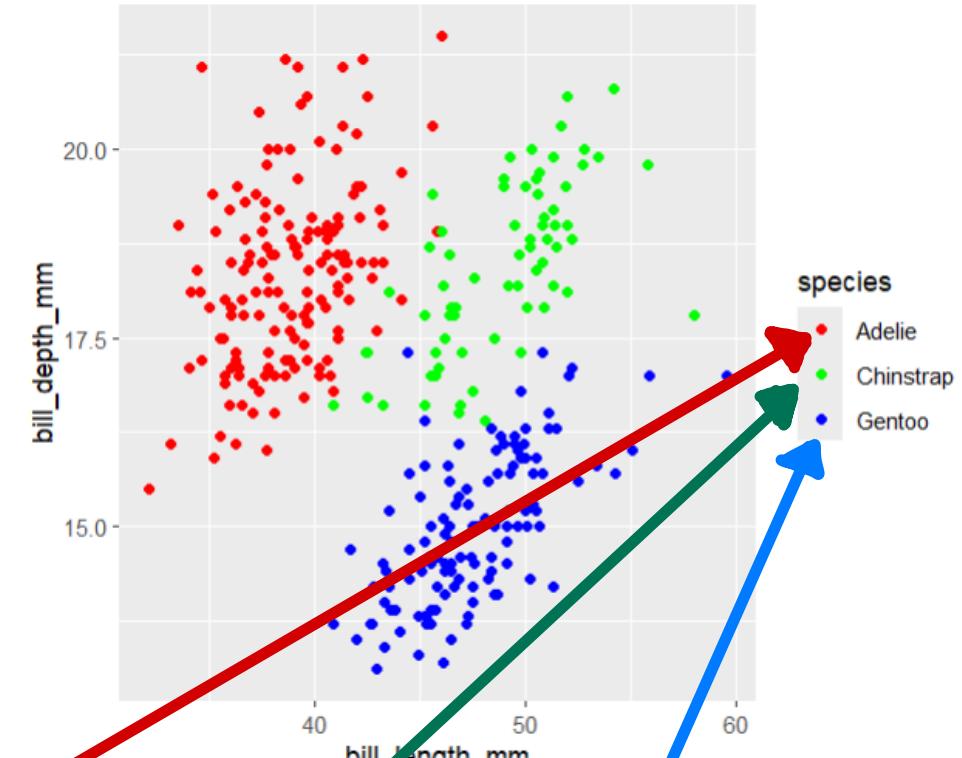
They don't match that is why
it didn't work.



Example 13: Change the colors from Example 11 to red, green, and blue, in that order.

```
fig_13 <- penguins %>%  
  ggplot(aes(x = bill_length_mm,  
             y = bill_depth_mm,  
             color = species)) +  
  geom_point() + matches, now  
  it works  
  scale_color_manual(values = c("red", "green", "blue"))
```

fig_13



Using External Color Packages

`'wesanderson'` & `'viridis'`

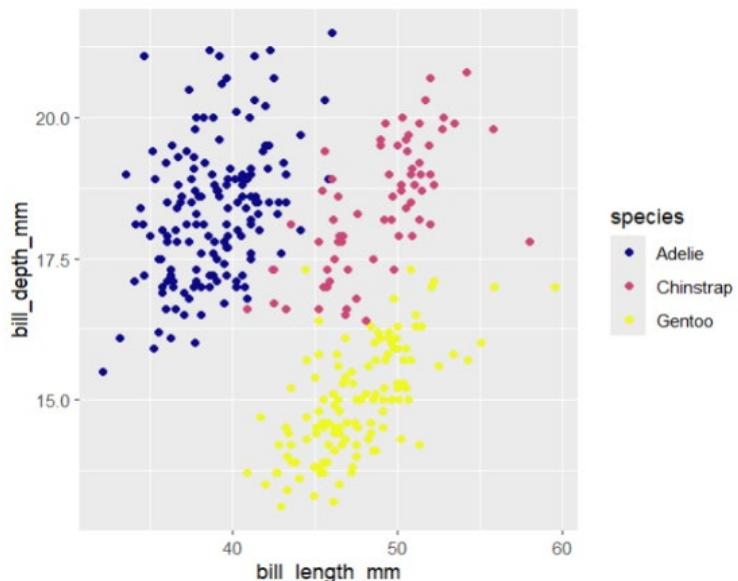


will cover this one.
It provides palettes for
you. You will need to
install the package.

Example 14: Create a scatter plot with $x = \text{bill_length_mm}$, $y = \text{bill_depth_mm}$, and species as the categorical variable. Use the viridis package to change the colors.

```
fig_14 <- penguins %>%  
  ggplot(aes(x = bill_length_mm, y = bill_depth_mm, color = species)) +  
  geom_point() +  
  scale_color_viridis(discrete = TRUE, option = "C", alpha = 1)
```

fig_14



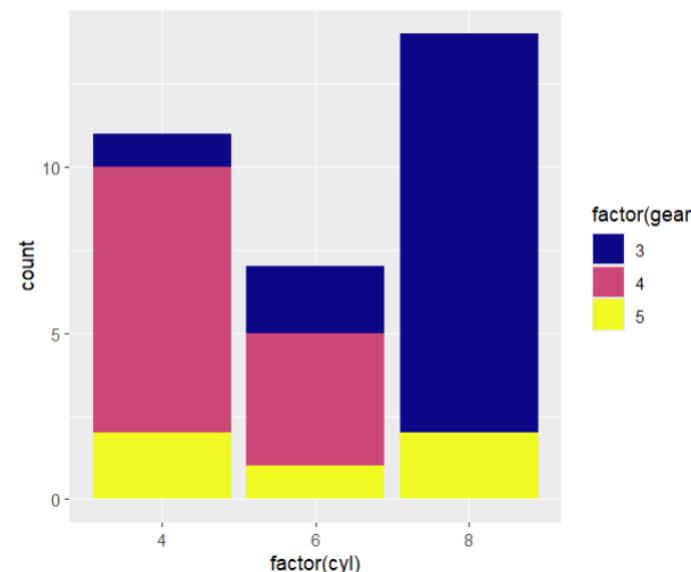
must match

palette.
Try "A", "B"
... etc
to see other
choices

between
0 and 1
It's on
the transparency
of the colors.

Example 15: Create a stacked bar graph with cyl in the x axis and gear as the categorical variable to split each column. Use the viridis package to change the colors.

```
fig_15 <- mtcars %>%  
  ggplot(aes(x = factor(cyl), fill = factor(gear)))+  
  geom_bar() + must match  
  scale_fill_viridis(discrete = TRUE, option = "C", alpha = 1)  
fig_15
```

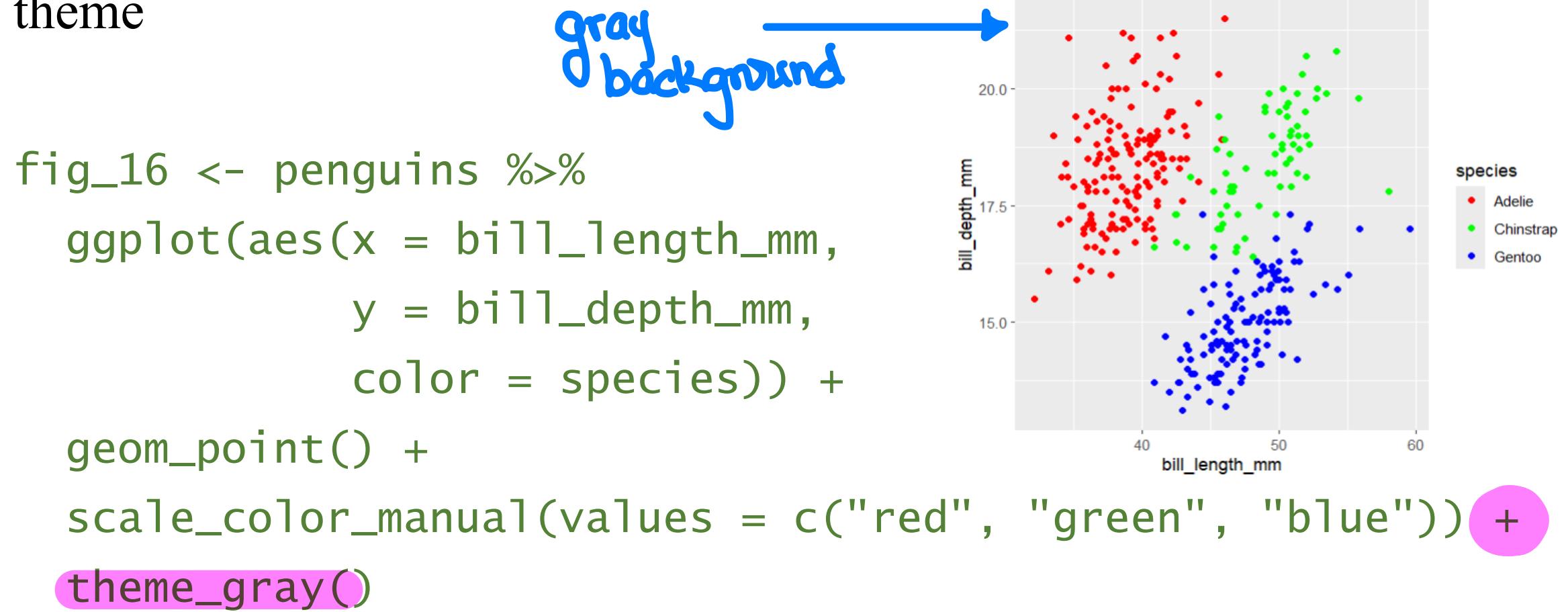


*Same description
as previously
example.*

Themes

- 1. theme_gray(): *Changes the background.*
- 2. theme_bw(): *Simply add it as*
- 3. theme_minimal(): *a "layer".*
- 4. theme_void():

Example 16: Create a scatter plot with $x = \text{bill_length_mm}$, $y = \text{bill_depth_mm}$, and species as the categorical variable. Use a gray theme



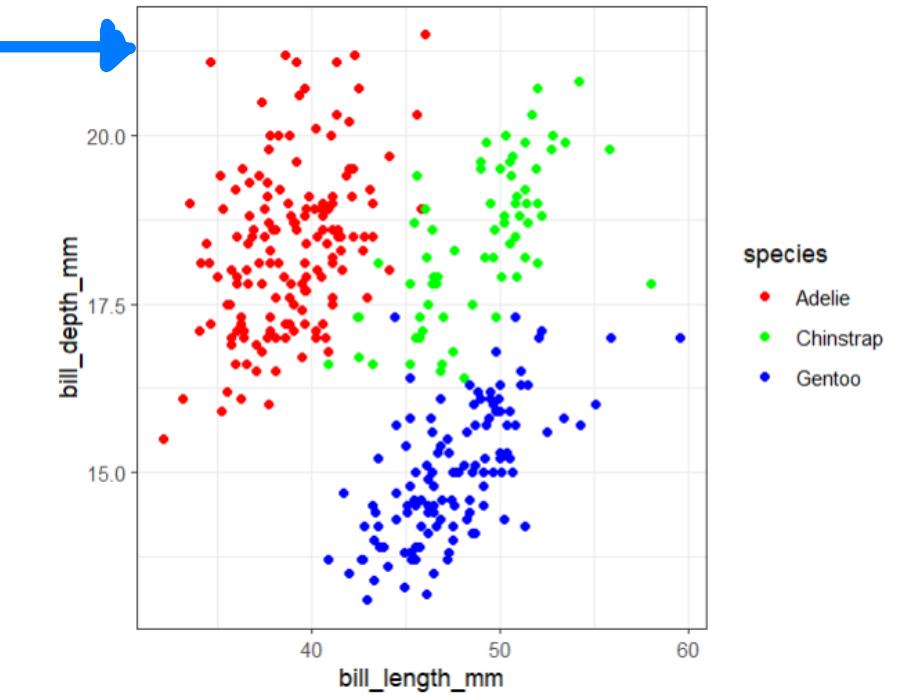
fig_16

Example 17: Create a scatter plot with $x = \text{bill_length_mm}$, $y = \text{bill_depth_mm}$, and species as the categorical variable. Use a black and white theme

```
fig_17 <- penguins %>%
  ggplot(aes(x = bill_length_mm,
             y = bill_depth_mm,
             color = species)) +
  geom_point() +
  scale_color_manual(values = c("red", "green", "blue")) +
  theme_bw()
```

fig_17

black
and white
background

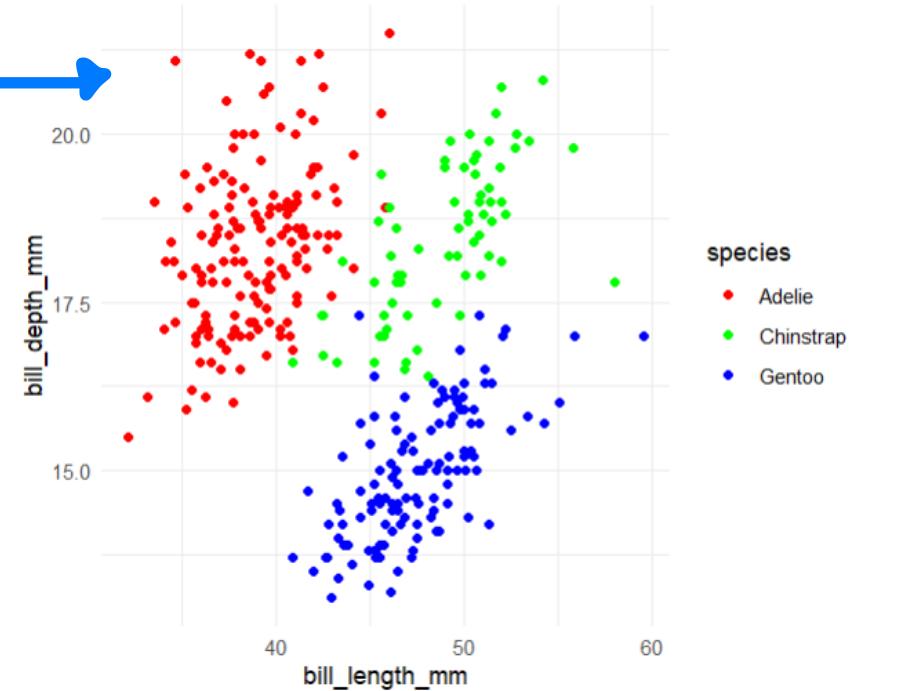


Example 18: Create a scatter plot with $x = \text{bill_length_mm}$, $y = \text{bill_depth_mm}$, and species as the categorical variable. Use a minimal theme.

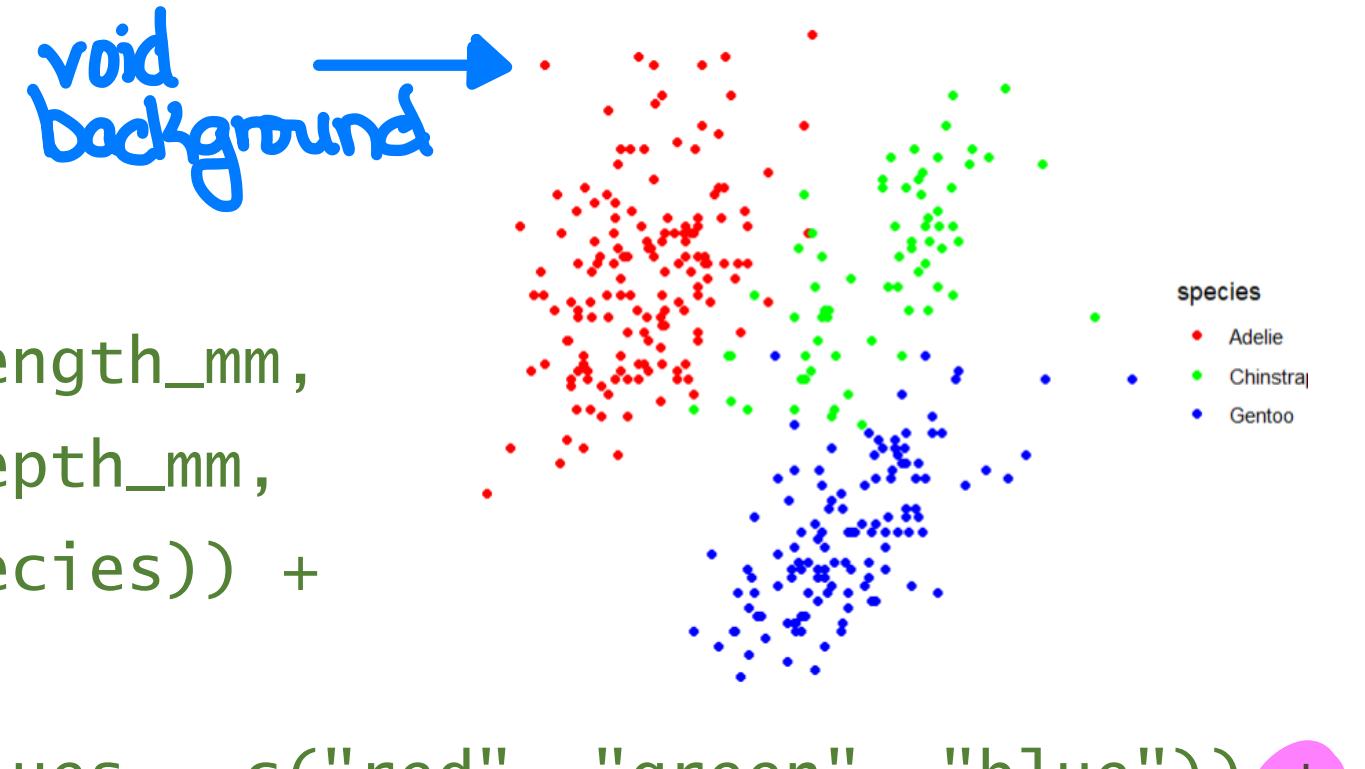
minimalistic
background

```
fig_18 <- penguins %>%
  ggplot(aes(x = bill_length_mm,
             y = bill_depth_mm,
             color = species)) +
  geom_point() +
  scale_color_manual(values = c("red", "green", "blue")) +
  theme_minimal()
```

fig_18



Example 19: Create a scatter plot with $x = \text{bill_length_mm}$, $y = \text{bill_depth_mm}$, and species as the categorical variable. Use a void theme.



```
fig_19 <- penguins %>%
  ggplot(aes(x = bill_length_mm,
             y = bill_depth_mm,
             color = species)) +
  geom_point() +
  scale_color_manual(values = c("red", "green", "blue")) +
  theme_void()
```

fig_19

Customizing Labels with `labs()`

`labs()` function in ggplot2

Add it as a "layer"

Example 20: Customizing Plot Labels

```
fig_20 <- penguins %>%
  ggplot(aes(x = bill_length_mm, y = bill_depth_mm, color = species)) +
  geom_point() +
  labs(
    title = "Scatter Plot of Bill Length vs. Bill Depth",
    x = "Length of Bill (mm)",
    y = "Depth of Bill (mm)",
    subtitle = "Data from the PalmerPenguins dataset",
    caption = "Source: Penguin Data",
    color = "La La La"
  )
fig_20
```

must match

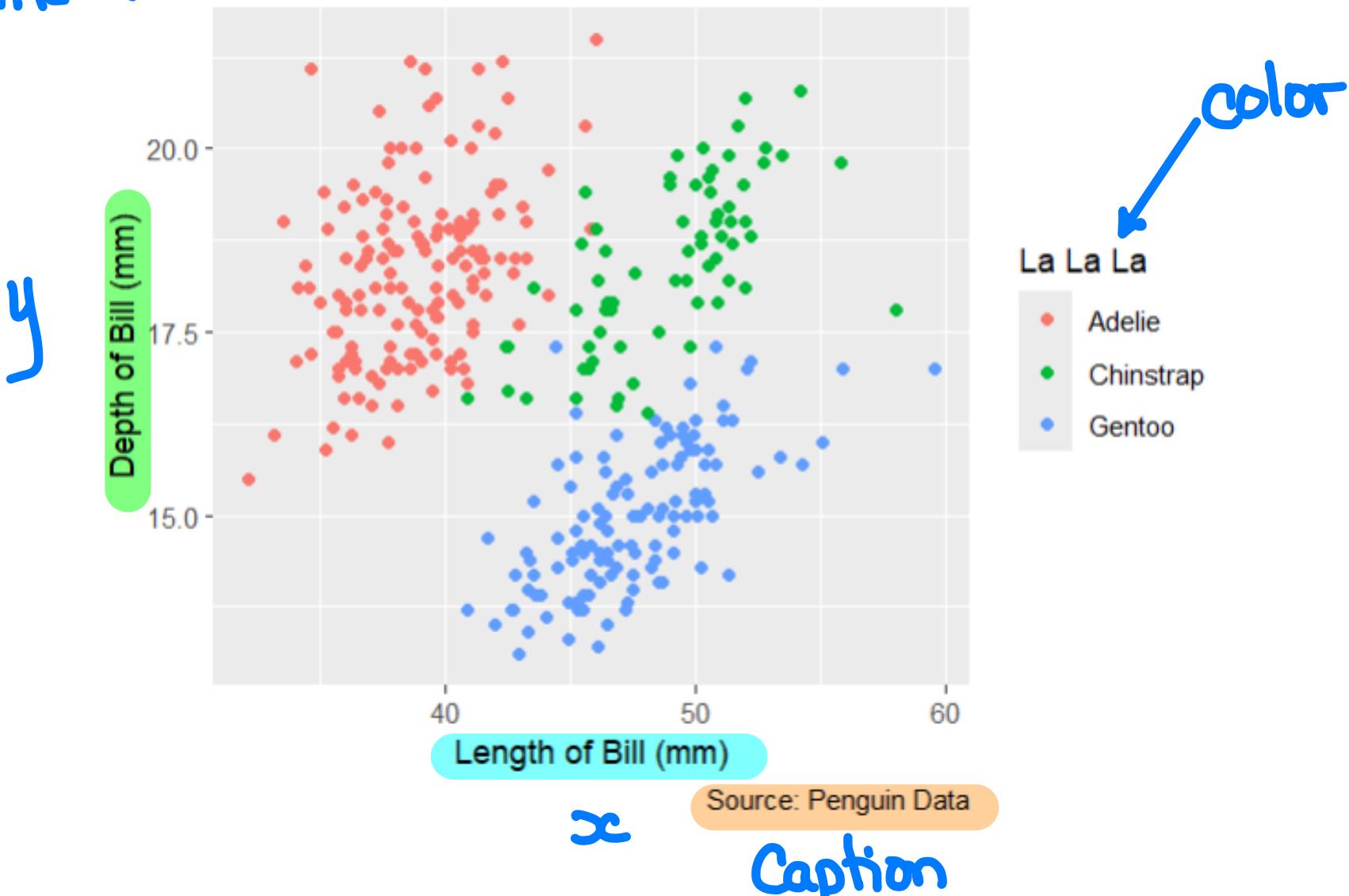
to change the name of the legend

see next slide to locate all of this.

Title
subtitle

← Scatter Plot of Bill Length vs. Bill Depth

← Data from the PalmerPenguins dataset



Example 21: Customizing Plot Labels

```
fig_21 <- mtcars %>%  
  ggplot(aes(x = factor(cyl), fill = factor(gear)))+  
  geom_bar() +  
  labs(  
    title = "Bar Graph",  
    x = "Cylinders",  
    y = "Total Number",  
    fill = "La La La" ← will change title of legend.  
  )  
fig_21
```

must match

Bar Graph

