Slow ggplot2

Evangeline Reynolds

2018-11-30

Contents

Chapter 1

Introduction

The ggplot2 package in R implements the "grammer of graphics" — a proposal of Leland Wilkinson and the product of the PhD and ongoing work of Hadley Wickham. This data visualization system idea is at once powerful, novel, and intuitive. Wickham conceives of and has built a system where data visualization as dividable into parameters: data to visualize, aesthetics that represent variables in the data, geometric objects, the coordinate system, specific scales, and statistical transformation.

While intuitive, using ggplotting effectively and efficiently requires practice. The "slow ggplotting" method and examples are designed to facilitate rapid incorporation of the ggplot logic and syntax. The method relys on action-reaction thinking — one of the most powerful tools in our "how-to" teaching tool kit. "Slow ggplotting" makes modifications to plots as incrementally as possible so that it is clear to users what code triggers each new layer or modification. The aim is less to be concise, but to be explicit about modifications, and facilitating more interactions with ggplot functions for newcomer internalization of the code.

Working incrementally is facilitatied by using the following (non conventional) conventions:

- pulling out aes() from the ggplot() function:
- using fewer functions; example - using labs() to add a title instead of ggtitle()
- using functions multiple times; example aes(x = var1) + aes(y = var2) rather than aes(x = var1, y = var2)
- using base R functions and tidyverse functions. For other packages, the :: style to call them
- write out arguments (no shortcuts) aes(x = gdppercap) not aes(gdppercap)
- order ggplot commands so that reactivity is obvious; scale adjustments to aesthetics might also be near the aesthetic declaration.

Here, I contrast the usual plotting method to slow ggplotting:

Usual approach:

```
ggplot(my_data, aes(var1, y = var2, col = var3)) +
  geom_point() +
  ggtitle("My Title") +
  labs(x = "the x label", y = "the y label", col = "legend title")
```

Using new slow ggplotting conventions:

```
ggplot(data = my_data) +
  aes(x = var1) +
  labs(x = "the x label") +
  aes(y = var2) +
  labs(y = "the y label") +
```

```
  geom_point() +
  aes(col = var3) +
  labs(col = "legend title") +
  labs(title = "My title")
```

The particular collection of visualizations here was produced for the Tableau-users-associated initiative #MakeoverMonday.

Chapter 2

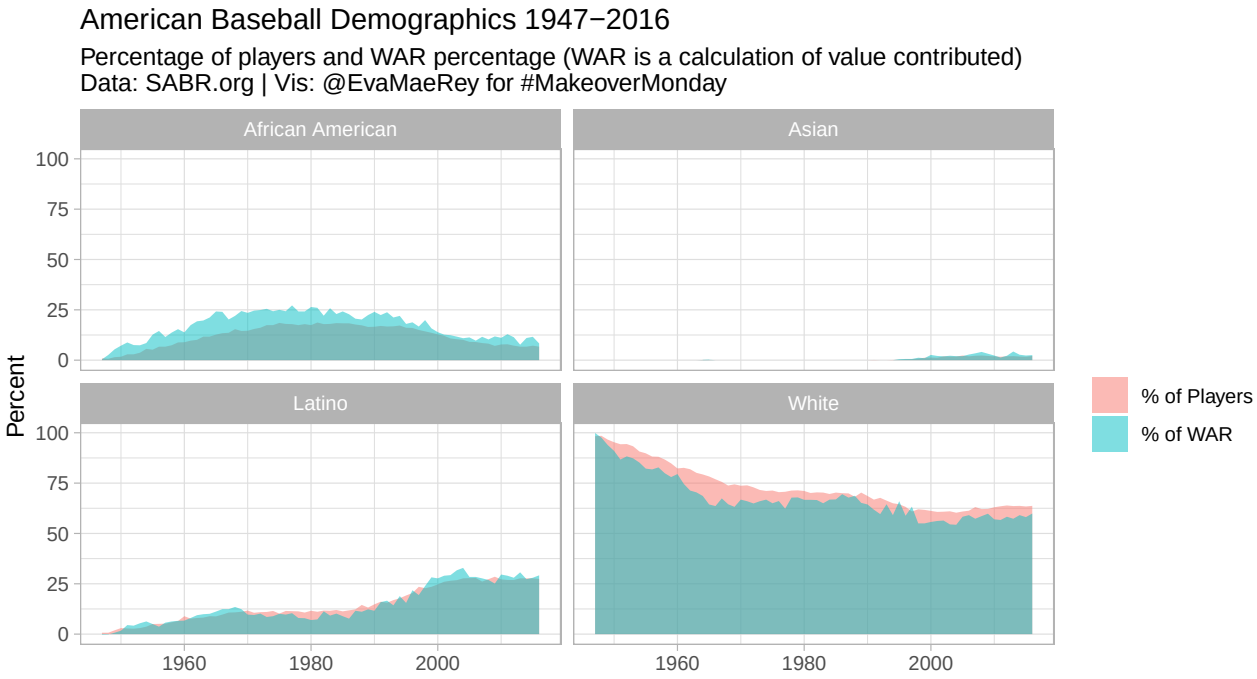
Baseball, WAR, and Ethnicity

This data visualization uses the WAR measure in baseball, a calculation based on the contributions of players. The visualizations show that new ethnicities and races started to be included in Major League baseball, the minority players that joined tended to contribute more than the expected value for players overall. For example, from 1947, when Jackie Robinson joined Major League baseball, and onward, the percent of African American players was outpaced by the percent calculated contributions (WAR) of African American players.

A random sample from the data set:

Year	Ethnicity	type	Percent
1952	African American	% of WAR	7.5
2013	Asian	% of WAR	4.3
1982	Asian	% of Players	0.0
2001	Latino	% of WAR	29.0
1950	African American	% of Players	1.7

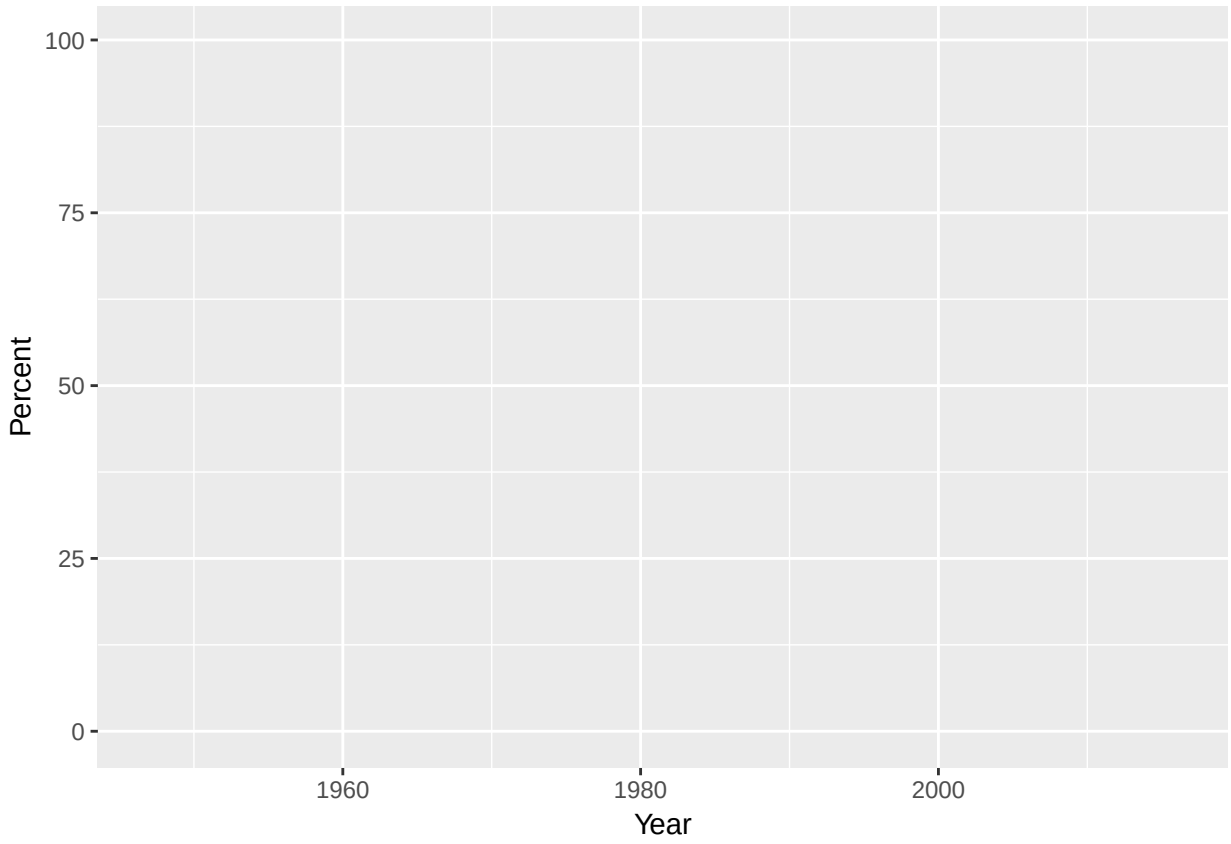
```
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type) +
  facet_wrap(~ Ethnicity) +
  geom_area(alpha = .5, position = "dodge") +
  labs(fill = "") +
  labs(x = "") +
  labs(title = "American Baseball Demographics 1947-2016") +
  labs(subtitle = "Percentage of players and WAR percentage (WAR is a calculation of value contributed)
  theme_light()
```


American Baseball Demographics 1947−2016
Percentage of players and WAR percentage (WAR is a calculation of value contributed)
Data: SABR.org | Vis: @EvaMaeRey for #MakeoverMonday
African American
Asian
Latino
White
Percent
100
75
50
25
0
1960
1980
2000
% of Players
% of WAR

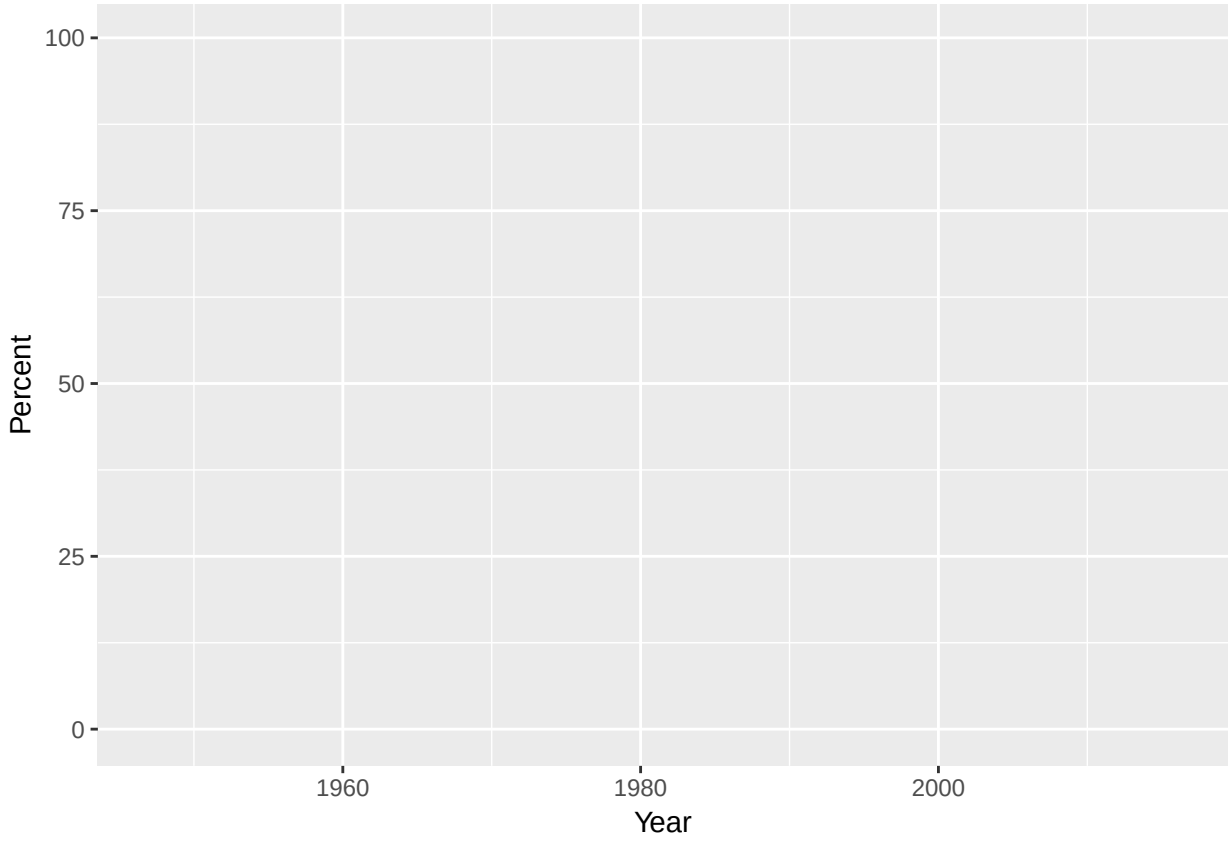
```
ggplot(df_gather)
```

```
ggplot(df_gather) +
  aes(x = Year)
```

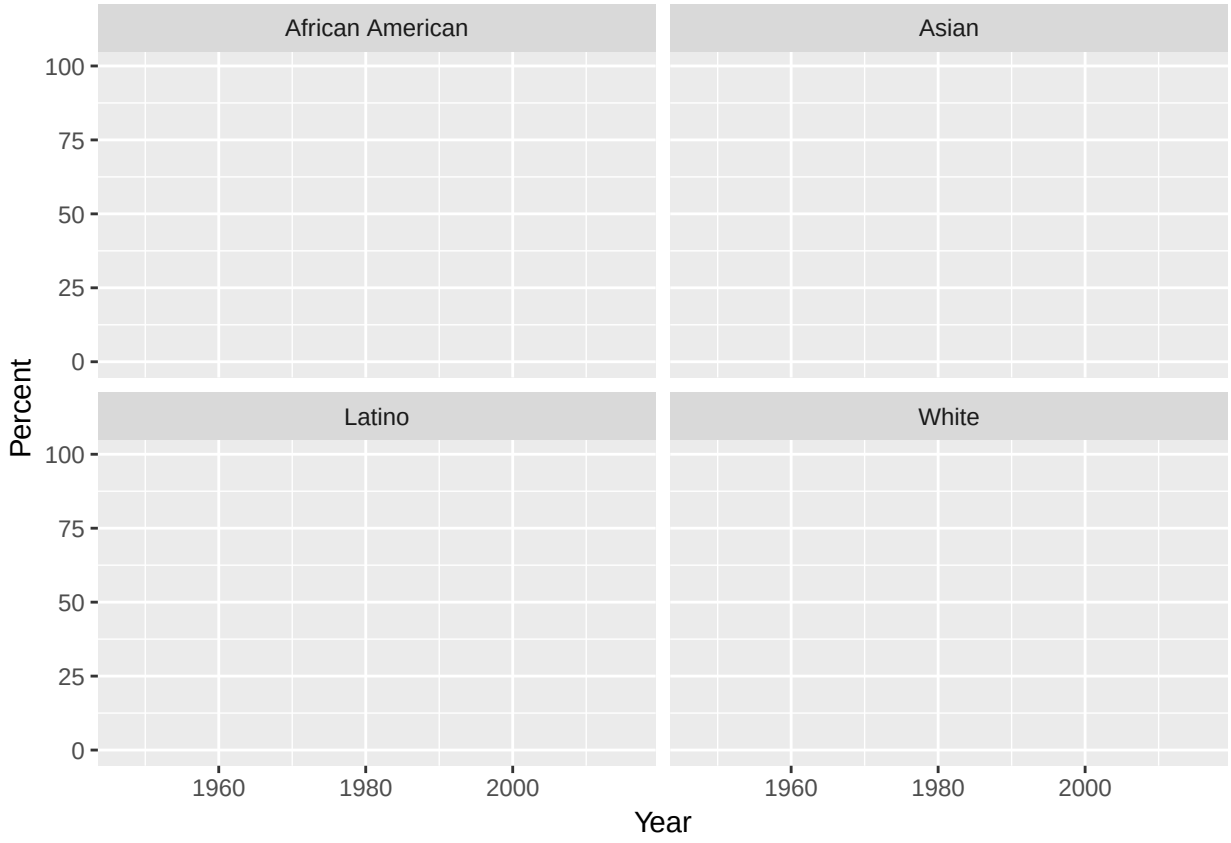
```
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent)
```



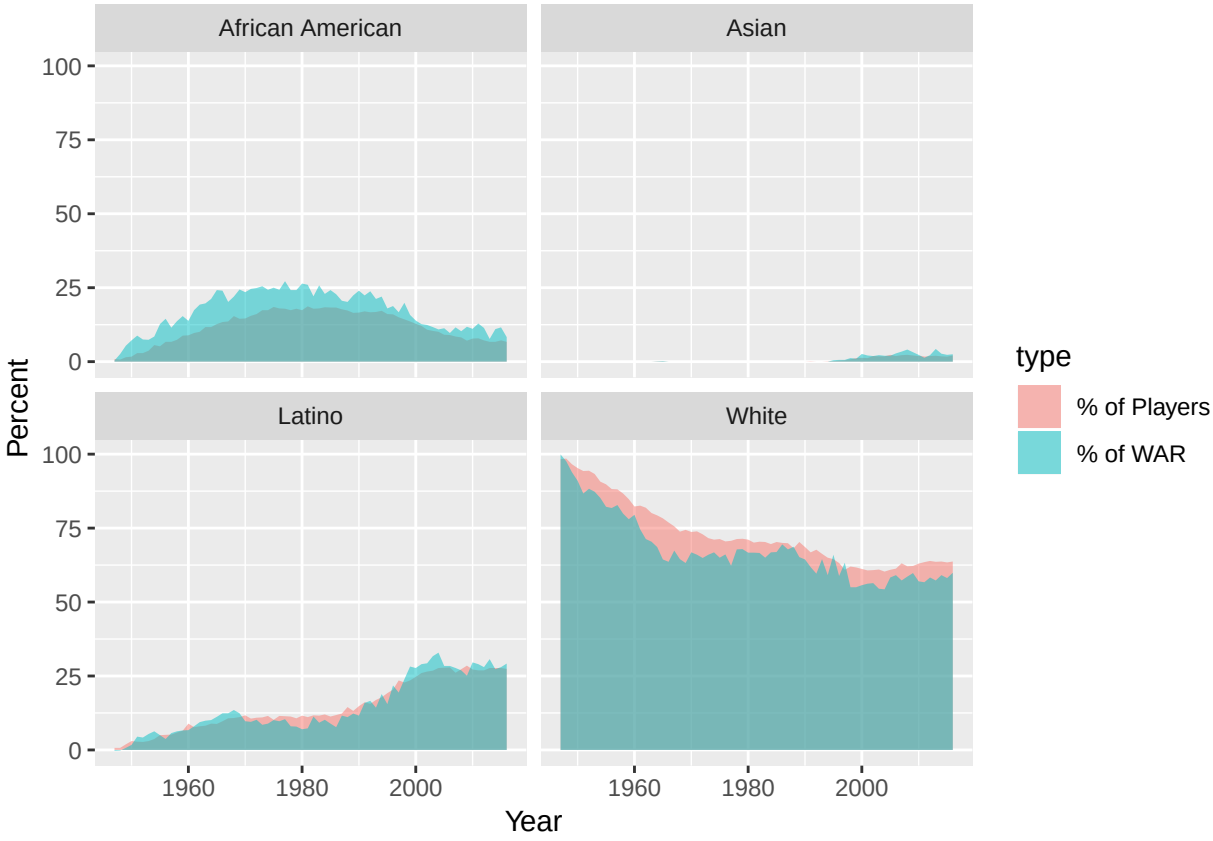
```
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type)
```



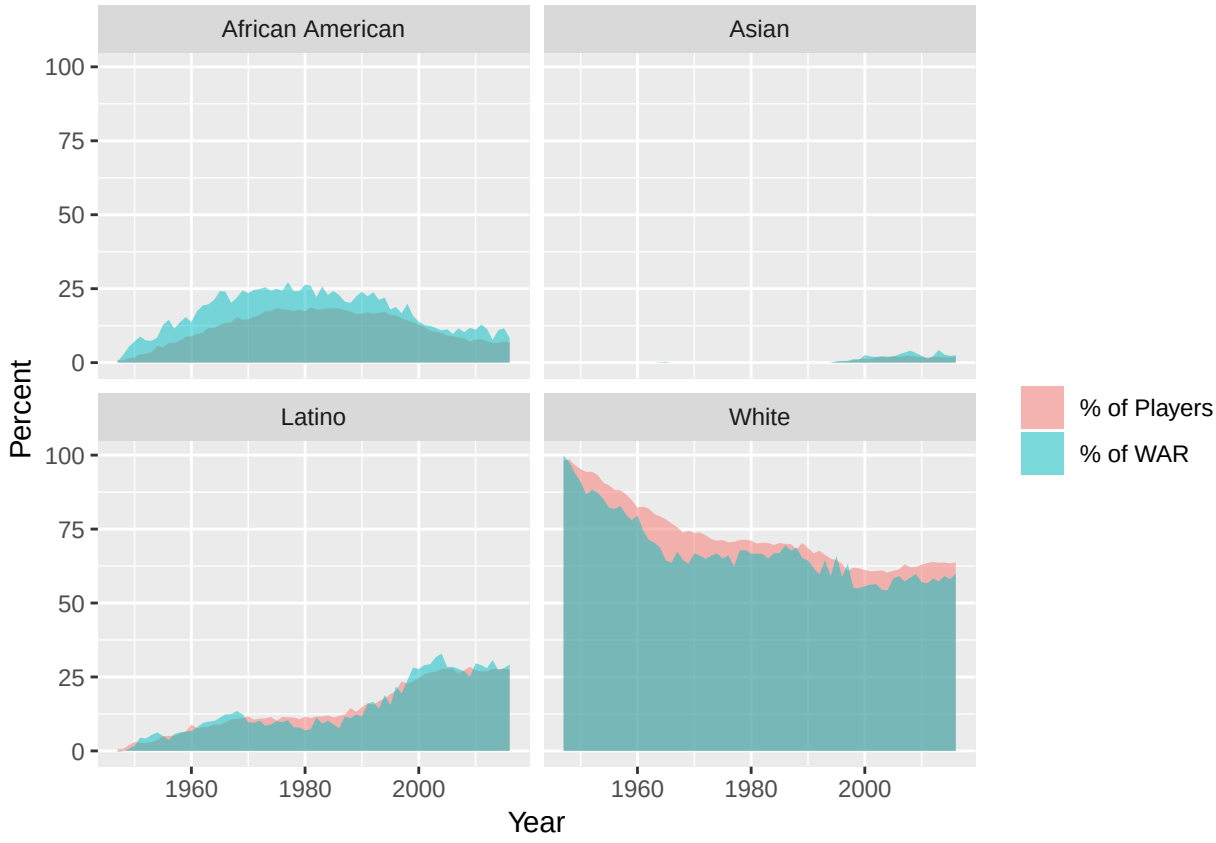
```
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type) +
  facet_wrap(~ Ethnicity)
```



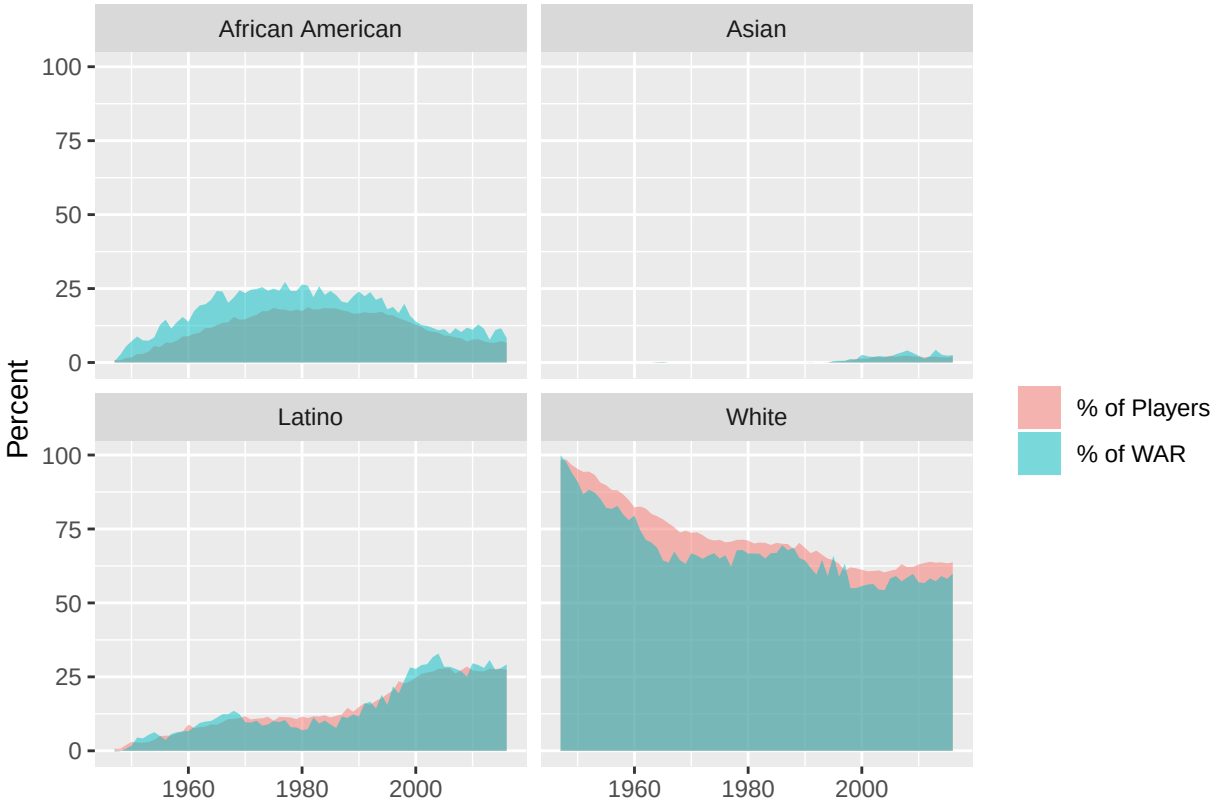
```
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type) +
  facet_wrap(~ Ethnicity) +
  geom_area(alpha = .5, position = "dodge")
```



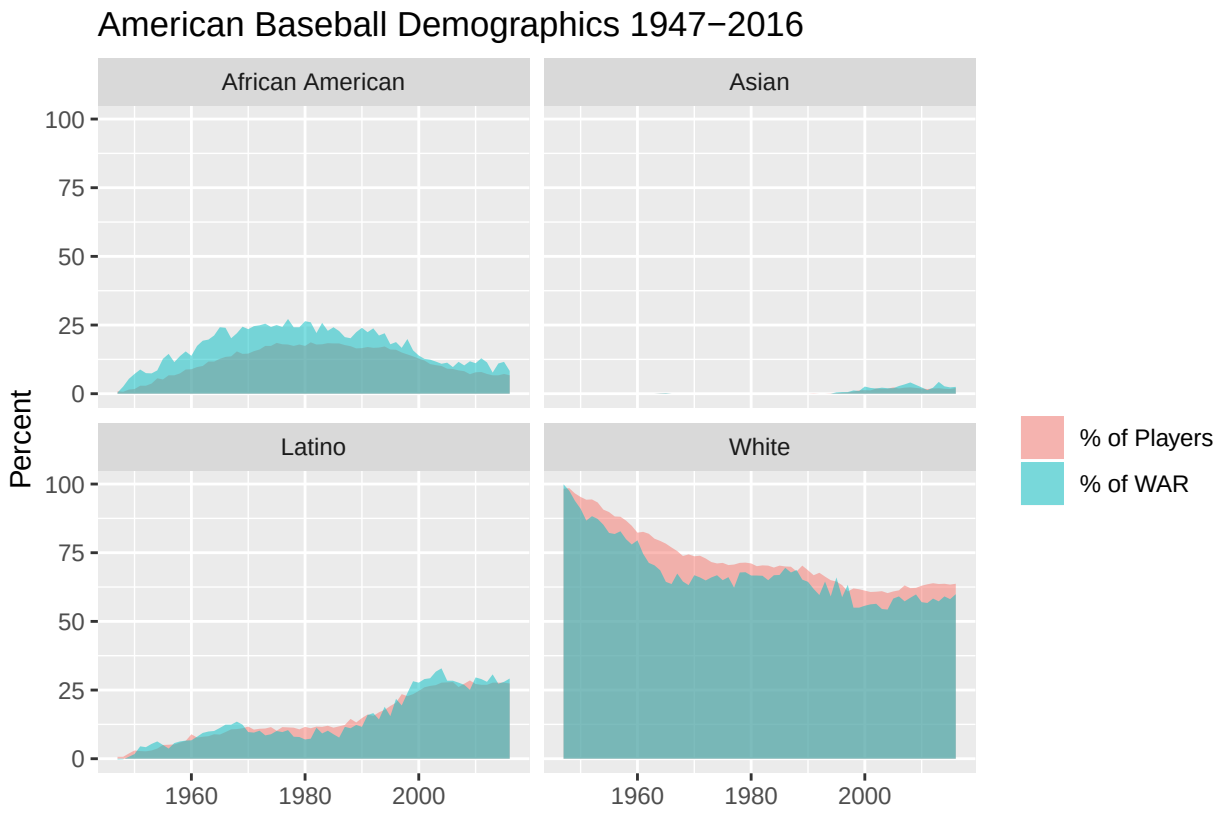
```
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type) +
  facet_wrap(~ Ethnicity) +
  geom_area(alpha = .5, position = "dodge") +
  labs(fill = "")
```



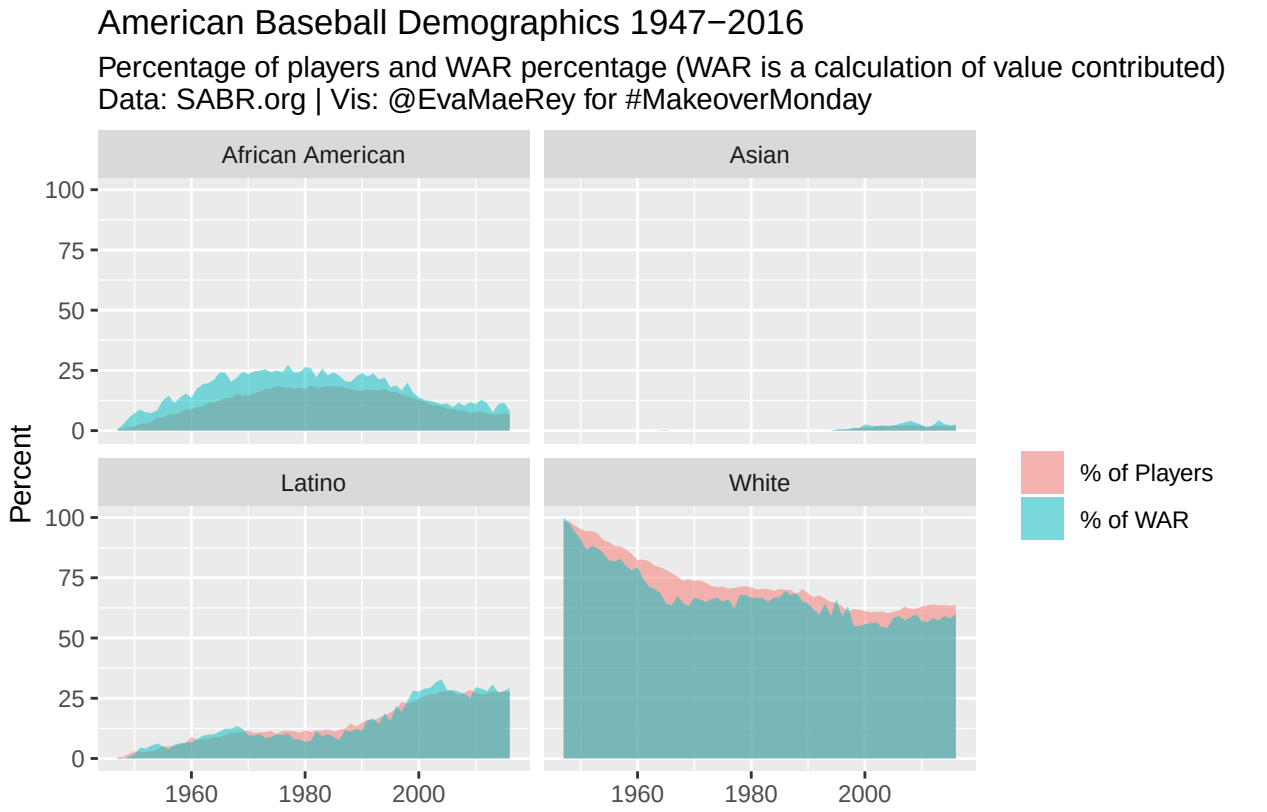
```
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type) +
  facet_wrap(~ Ethnicity) +
  geom_area(alpha = .5, position = "dodge") +
  labs(fill = "") +
  labs(x = "")
```



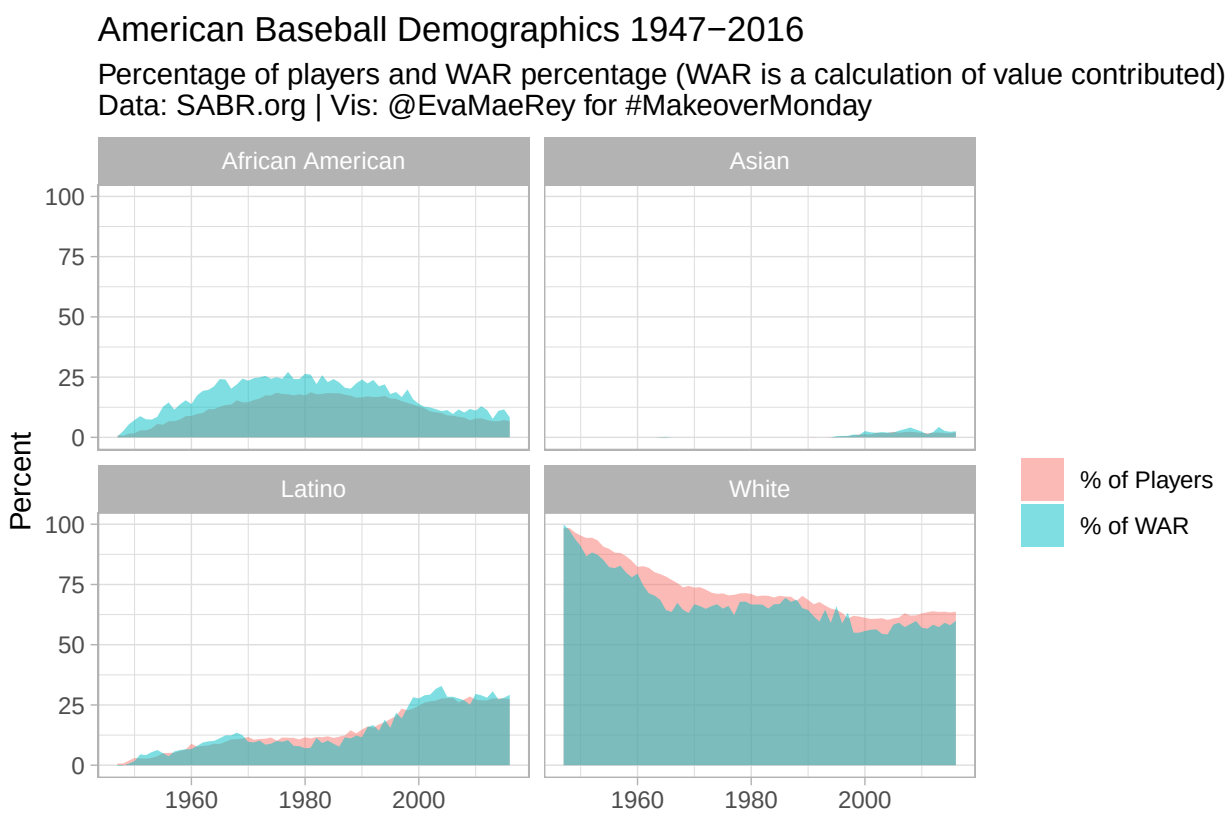
```
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type) +
  facet_wrap(~ Ethnicity) +
  geom_area(alpha = .5, position = "dodge") +
  labs(fill = "") +
  labs(x = "") +
  labs(title = "American Baseball Demographics 1947-2016")
```



```
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type) +
  facet_wrap(~ Ethnicity) +
  geom_area(alpha = .5, position = "dodge") +
  labs(fill = "") +
  labs(x = "") +
  labs(title = "American Baseball Demographics 1947-2016") +
  labs(subtitle = "Percentage of players and WAR percentage (WAR is a calculation of value contributed)
```



```
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type) +
  facet_wrap(~ Ethnicity) +
  geom_area(alpha = .5, position = "dodge") +
  labs(fill = "") +
  labs(x = "") +
  labs(title = "American Baseball Demographics 1947-2016") +
  labs(subtitle = "Percentage of players and WAR percentage (WAR is a calculation of value contributed)
  theme_light()
```



Chapter 3

Christmas Trees

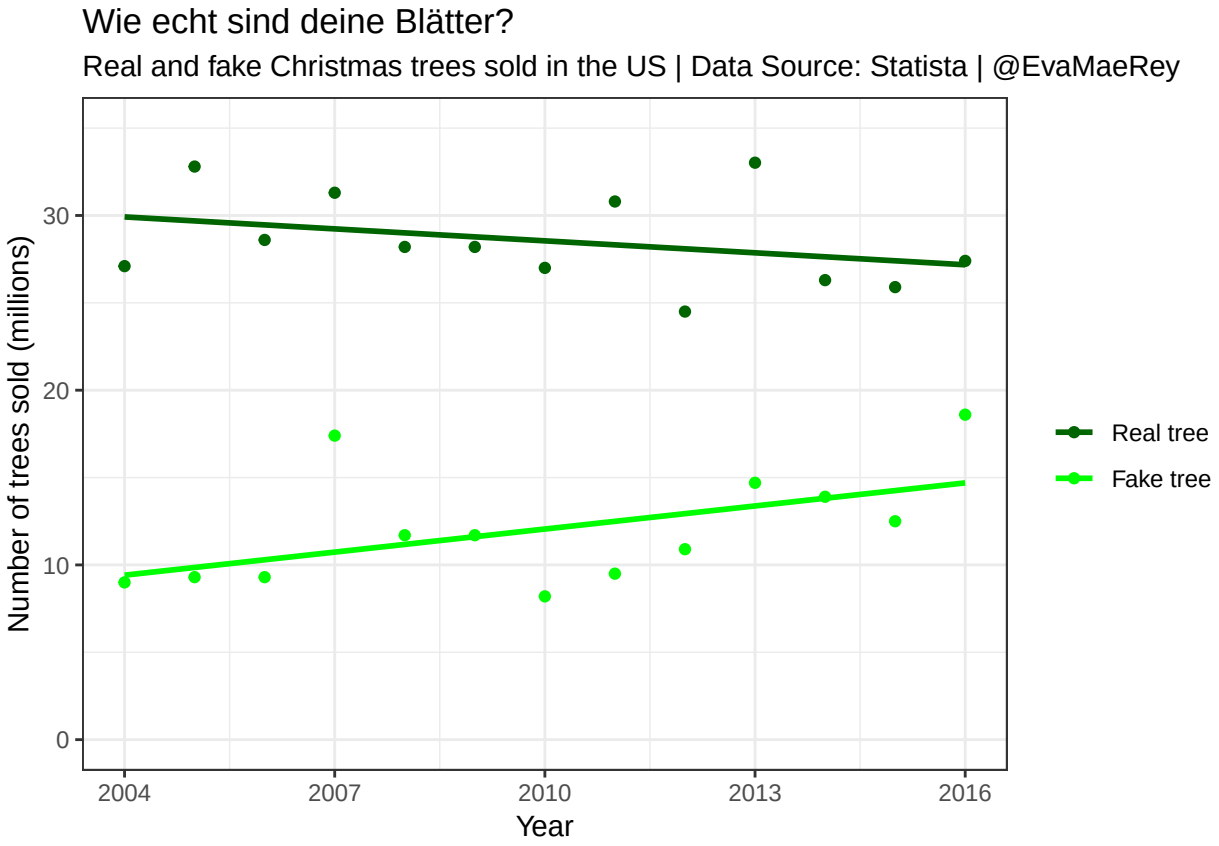
Here is a simple plot of Christmas Tree Sales in the U.S. The plot shows that artificial tree sales are on the rise, contrasting with declines in real trees. The title plays on the German Christmas Carol "O Tannenbaum", "Oh Christmas Tree" in English. "Wie echt sind deine Blätter?" means "how real are your leaves"; the original text from the carol is "Wie treu sind deine Blätter!" which means "How true your leaves are!"

I also plot the cumulative number of trees purchased of each type, artificial and real, from 2004 to 2014, comparing that to the 2016 U.S. population. Almost one real tree per person was bought over the course of 10 years!

A random sample from the data set:

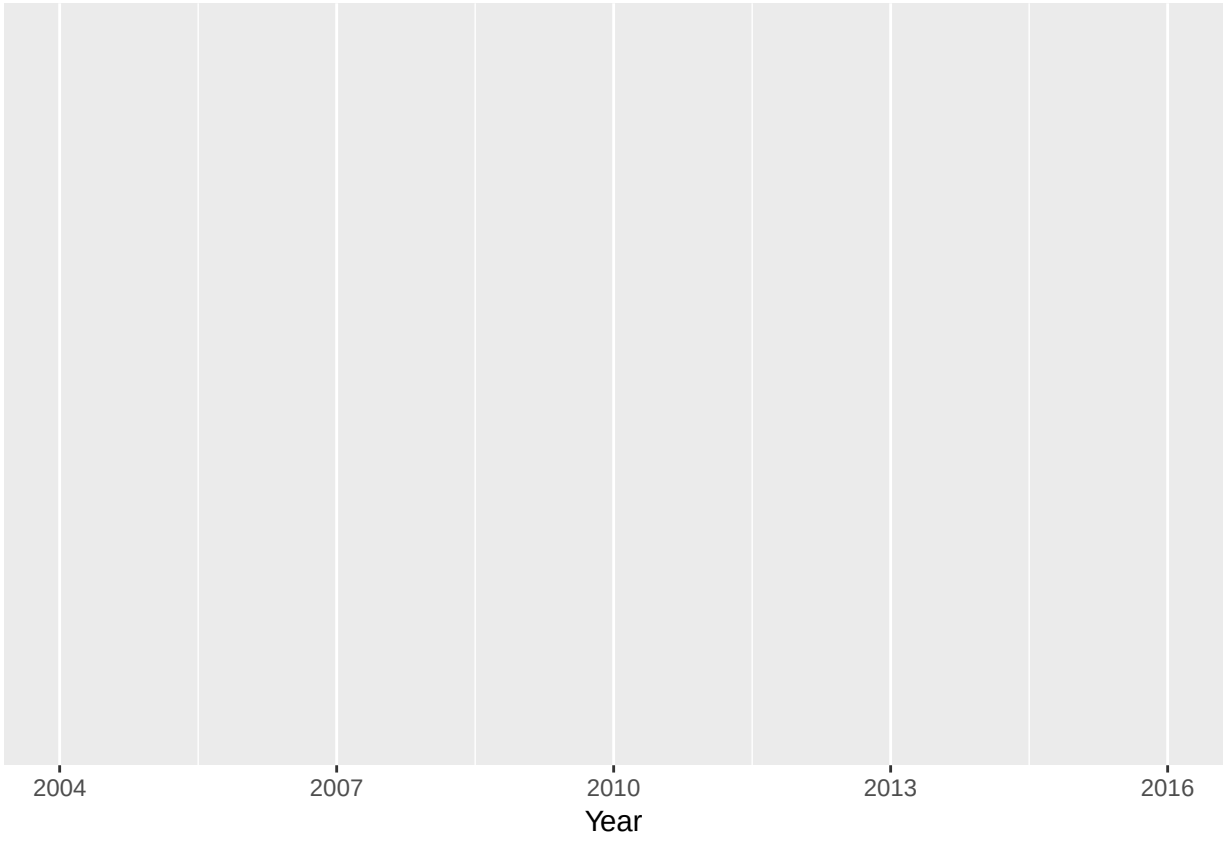
Year	Number of trees sold	Type of tree	Number of trees sold (millions)
2013	14700000	Fake tree	14.7
2008	11700000	Fake tree	11.7
2014	13900000	Fake tree	13.9
2007	17400000	Fake tree	17.4
2015	12500000	Fake tree	12.5

```
ggplot(data = dta) +
  aes(Year) +
  aes(y = `Number of trees sold (millions)`) +
  geom_point() +
  aes(col = fct_rev(`Type of tree`)) +
  geom_smooth(method = "lm", se = F) +
  scale_color_manual(values = c("darkgreen", "green")) +
  ylim(c(0, 35)) +
  labs(col = "") +
  labs(title = "Wie echt sind deine Blätter?") +
  labs(subtitle = "Real and fake Christmas trees sold in the US | Data Source: Statista | @EvaMaeRey ")
  theme_bw()
```

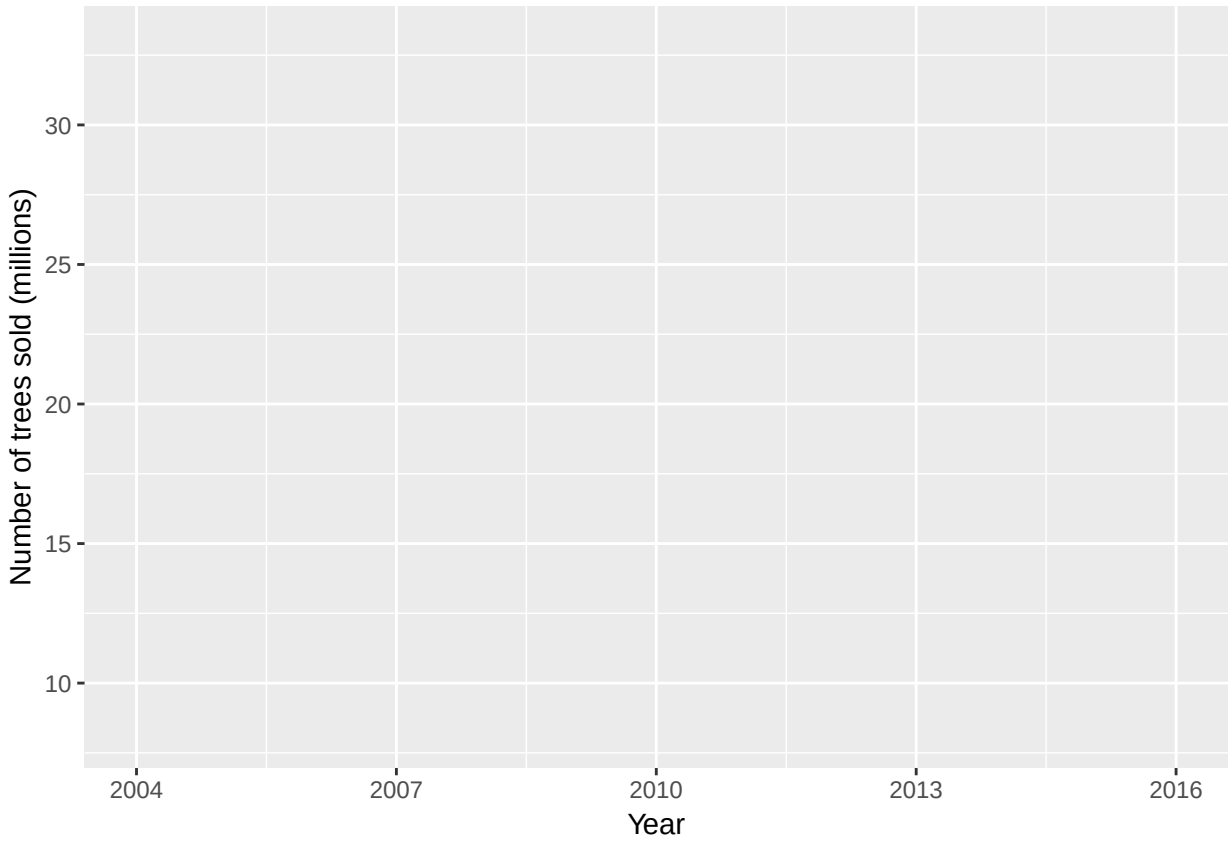

Wie echt sind deine Blätter?
Real and fake Christmas trees sold in the US | Data Source: Statista | @EvaMaeRey
Number of trees sold (millions)
0
10
20
30
2004
2007
2010
2013
2016
Year
Real tree
Fake tree

```
ggplot(data = dta)
```

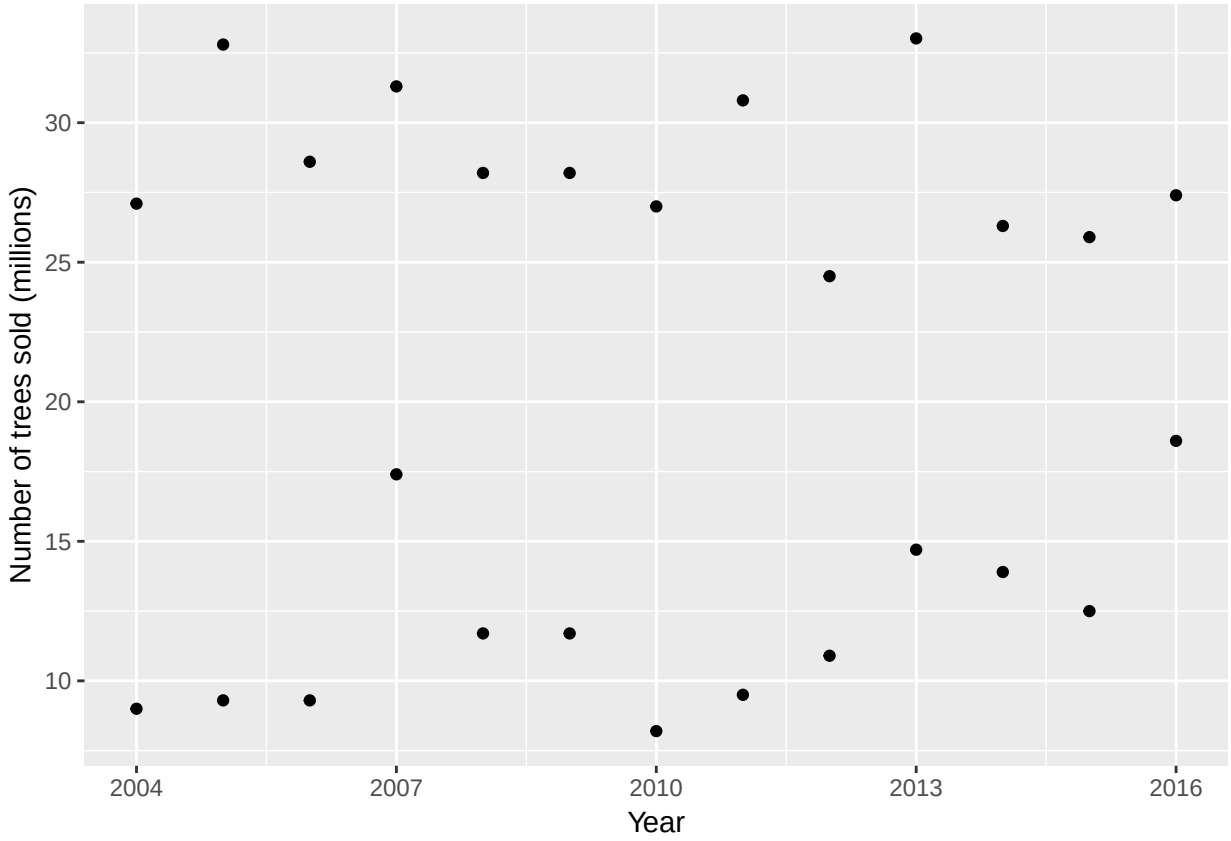
```
ggplot(data = dta) +
  aes(Year)
```



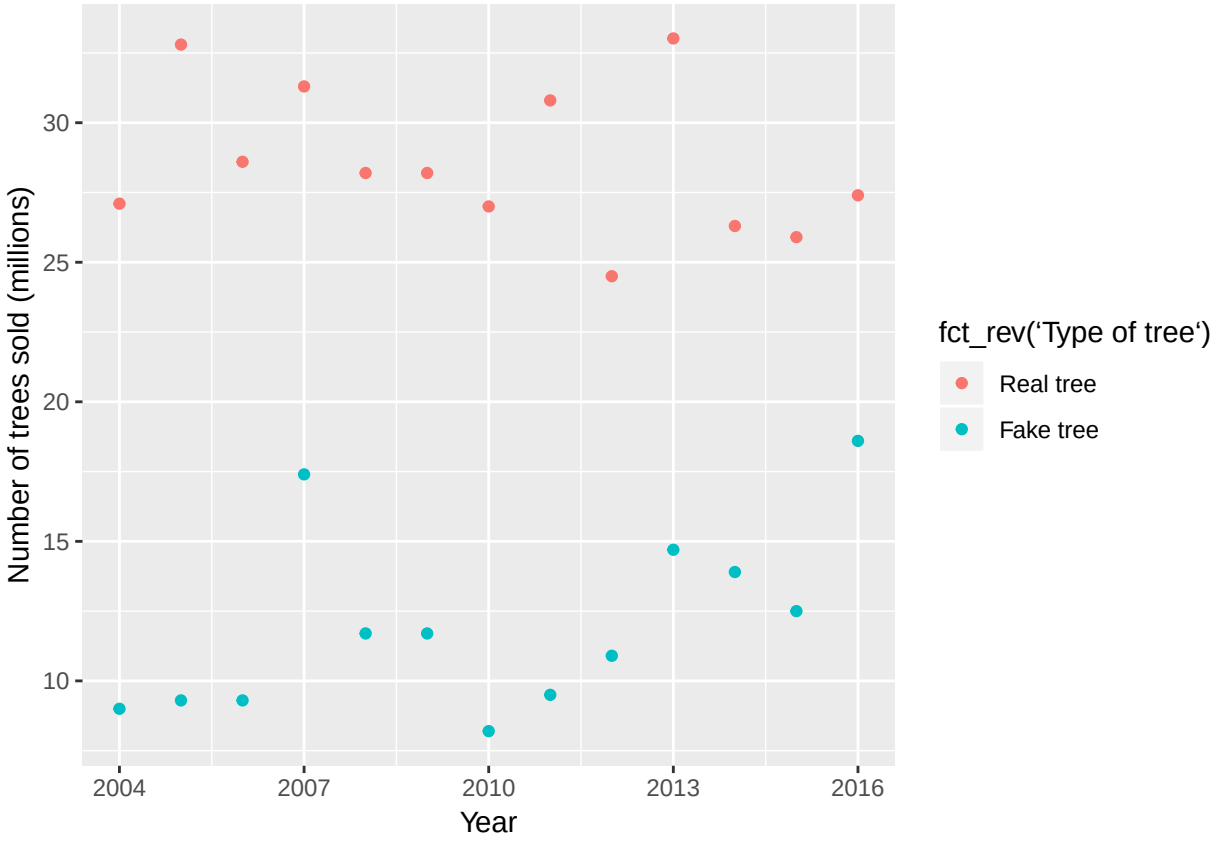
```
ggplot(data = dta) +
  aes(Year) +
  aes(y = `Number of trees sold (millions)`)
```



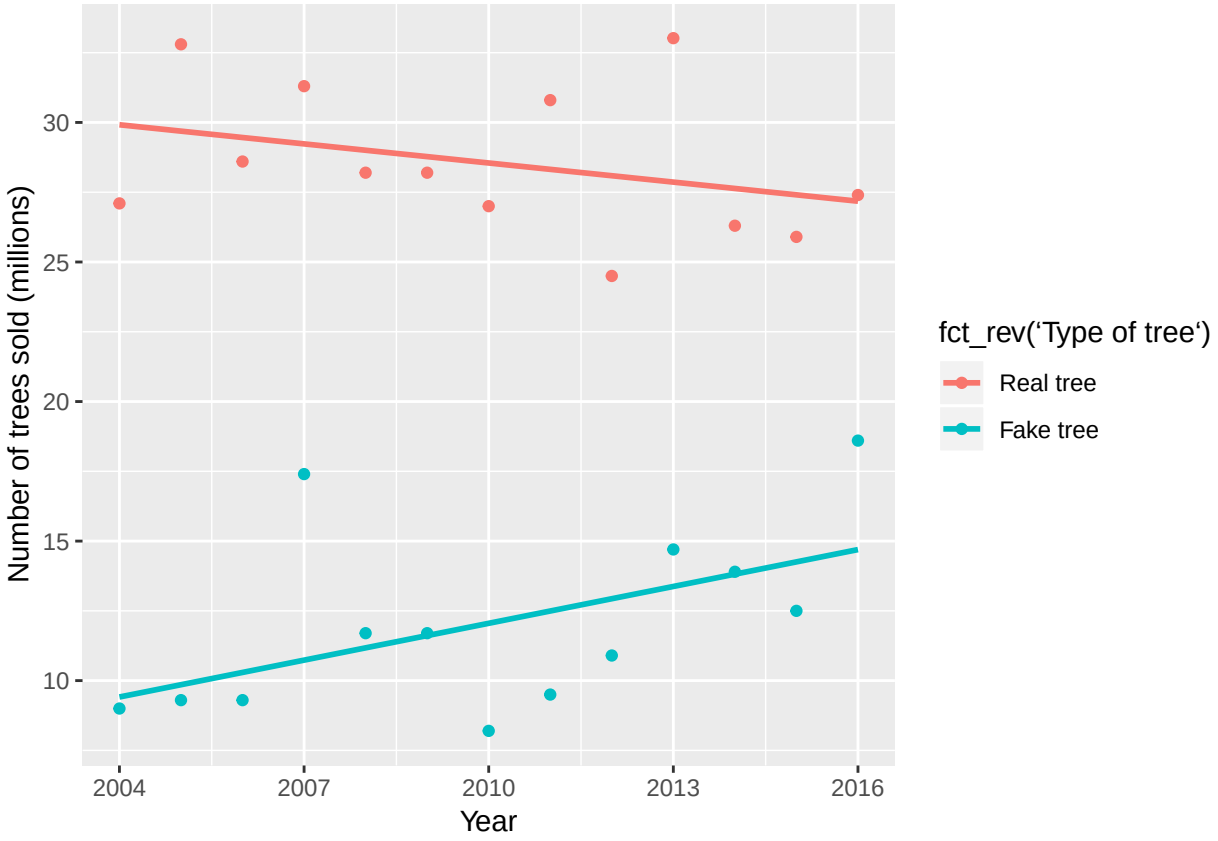
```
ggplot(data = dta) +
  aes(Year) +
  aes(y = `Number of trees sold (millions)`) +
  geom_point()
```



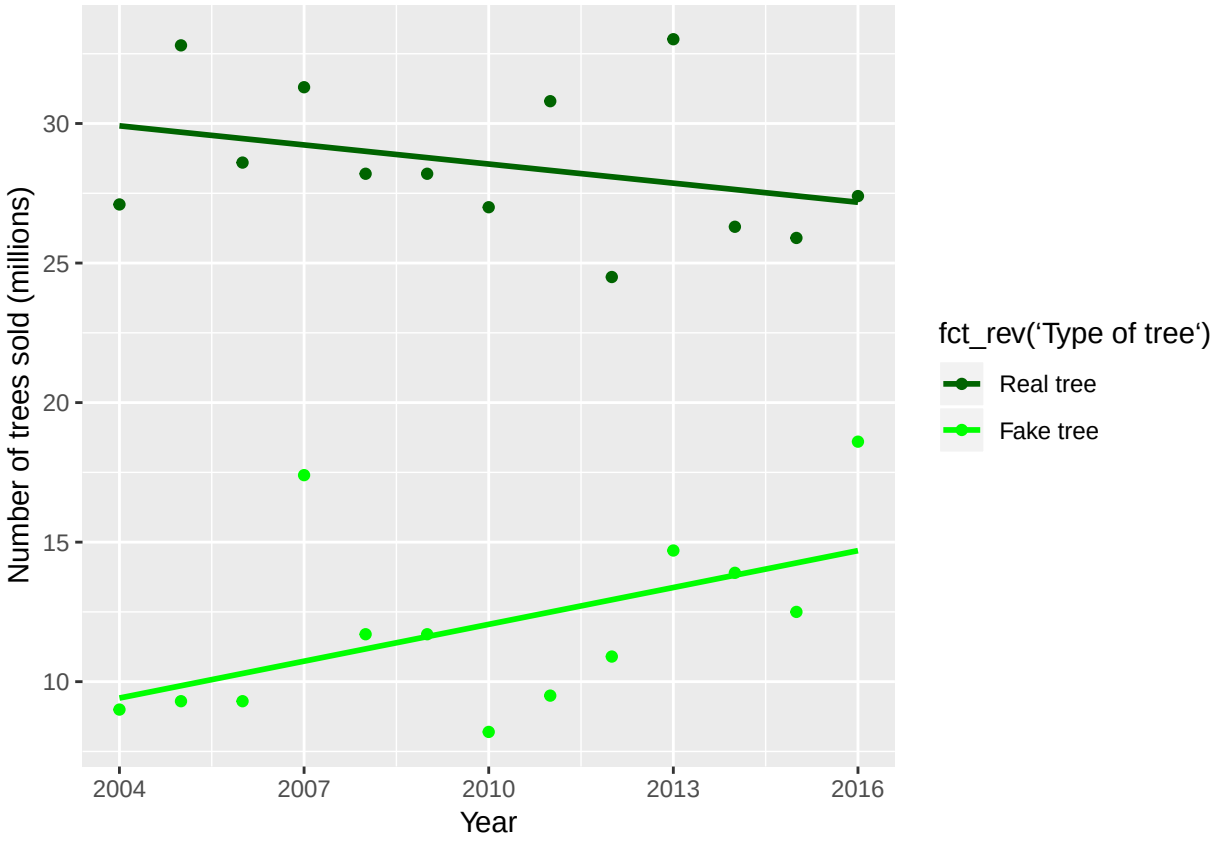
```
ggplot(data = dta) +
  aes(Year) +
  aes(y = `Number of trees sold (millions)`) +
  geom_point() +
  aes(col = fct_rev(`Type of tree`))
```



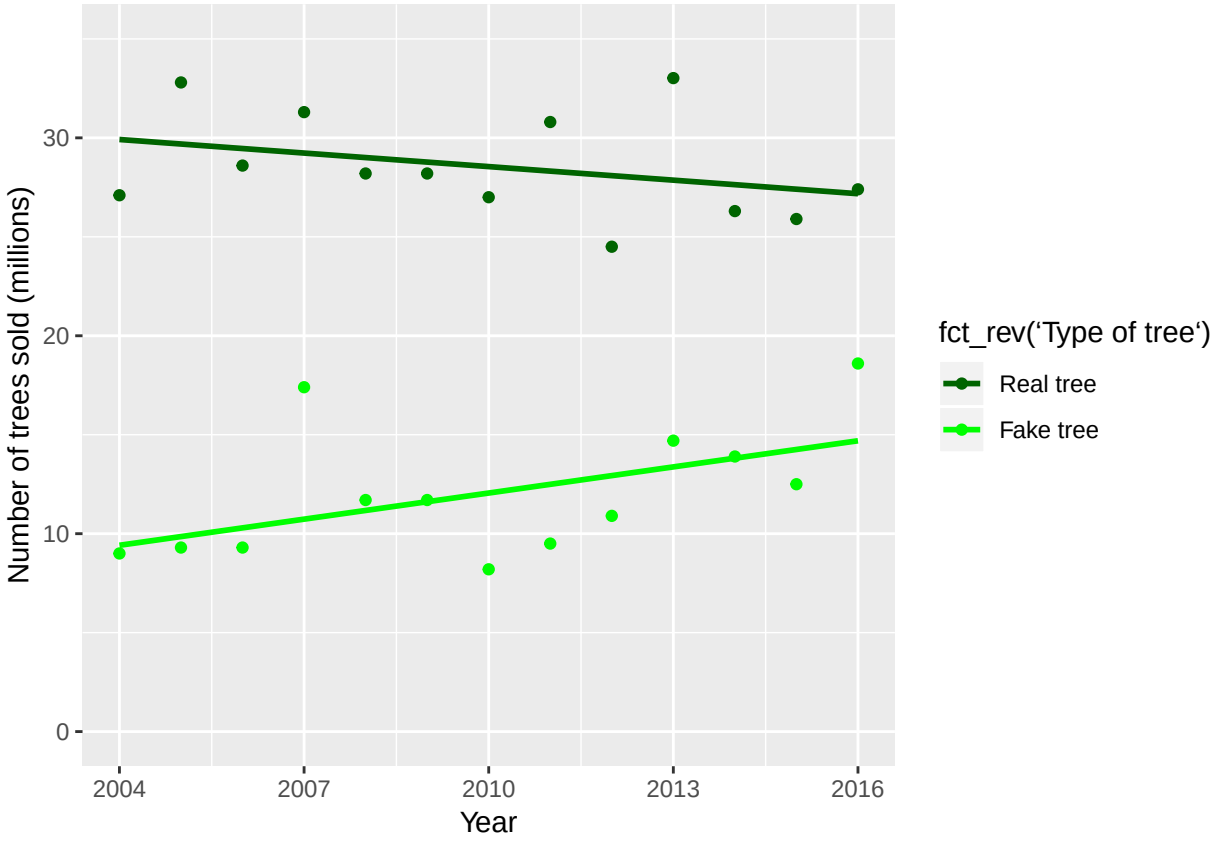
```
ggplot(data = dta) +
  aes(Year) +
  aes(y = `Number of trees sold (millions)`) +
  geom_point() +
  aes(col = fct_rev(`Type of tree`)) +
  geom_smooth(method = "lm", se = F)
```



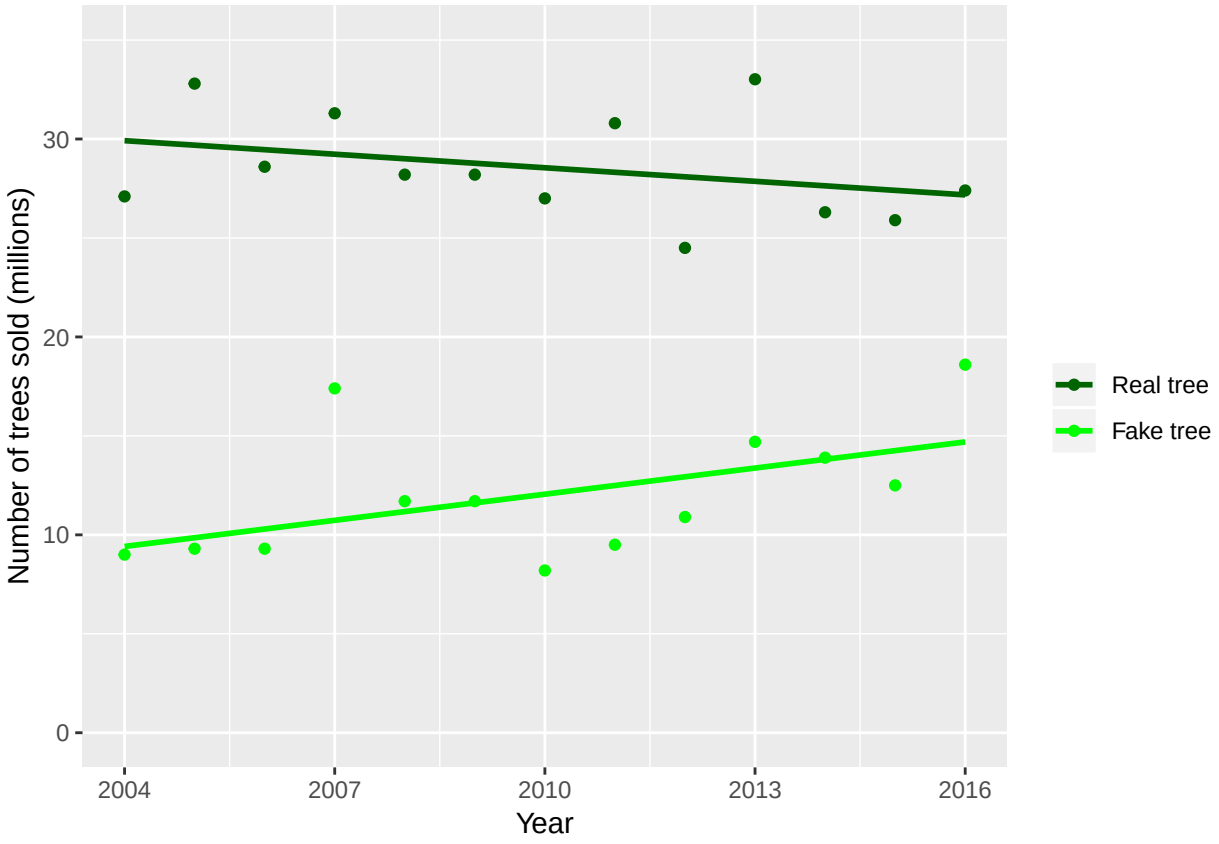
```
ggplot(data = dta) +
  aes(Year) +
  aes(y = `Number of trees sold (millions)`) +
  geom_point() +
  aes(col = fct_rev(`Type of tree`)) +
  geom_smooth(method = "lm", se = F) +
  scale_color_manual(values = c("darkgreen", "green"))
```



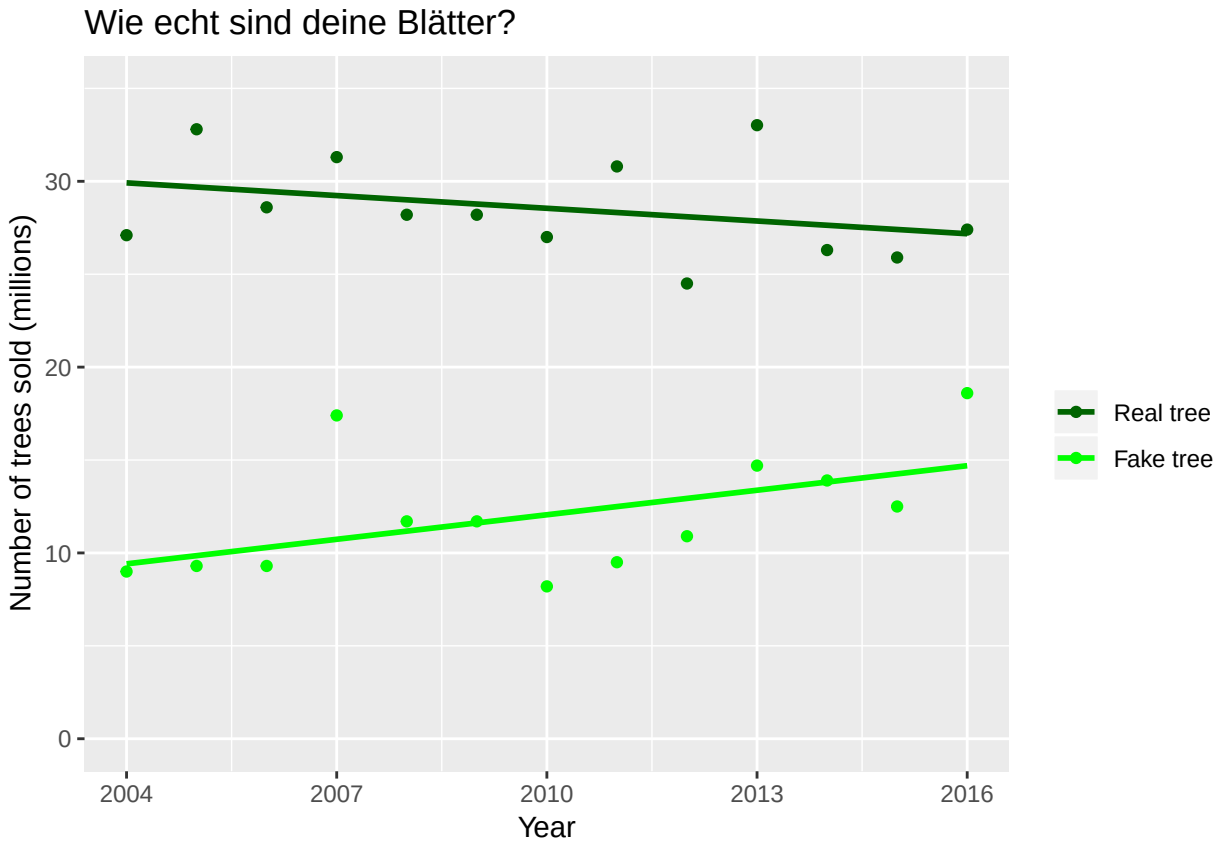
```
ggplot(data = dta) +
  aes(Year) +
  aes(y = `Number of trees sold (millions)`) +
  geom_point() +
  aes(col = fct_rev(`Type of tree`)) +
  geom_smooth(method = "lm", se = F) +
  scale_color_manual(values = c("darkgreen", "green")) +
  ylim(c(0, 35))
```



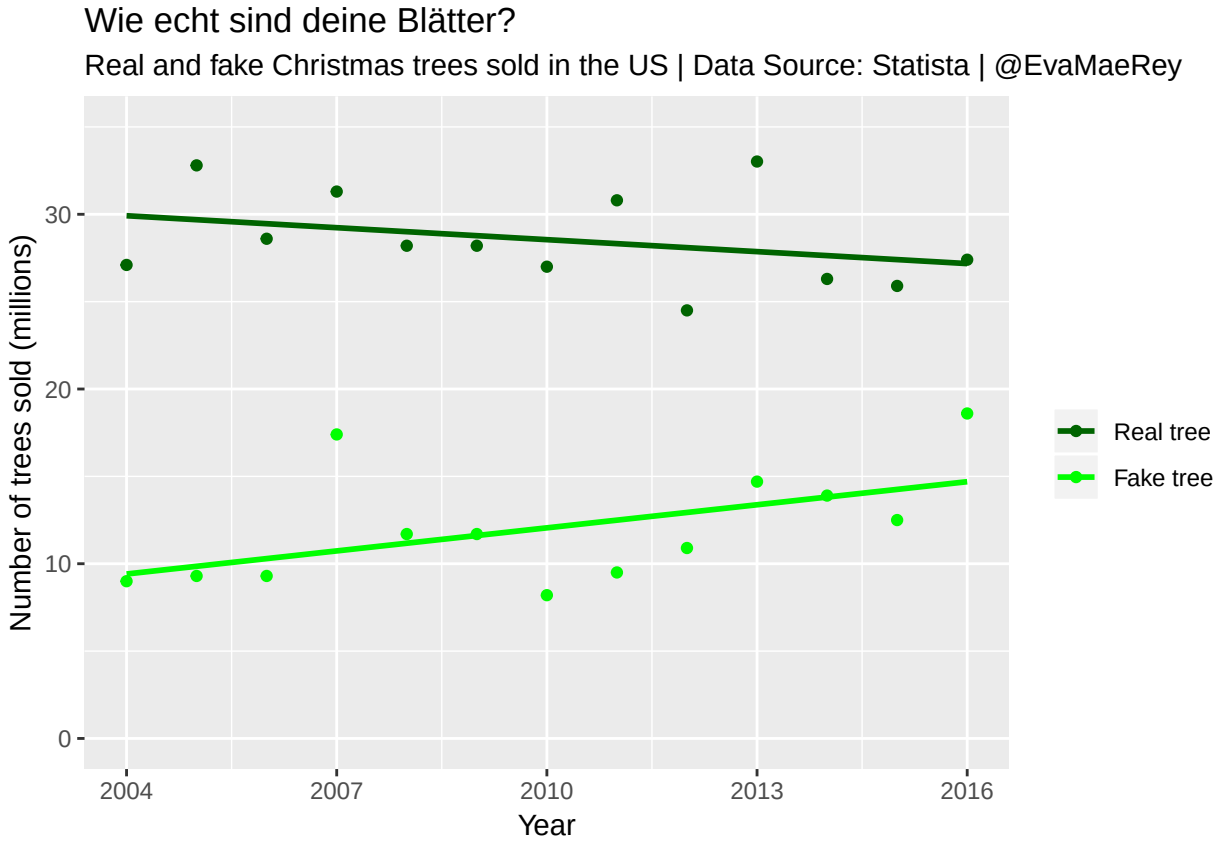
```
ggplot(data = dta) +
  aes(Year) +
  aes(y = `Number of trees sold (millions)`) +
  geom_point() +
  aes(col = fct_rev(`Type of tree`)) +
  geom_smooth(method = "lm", se = F) +
  scale_color_manual(values = c("darkgreen", "green")) +
  ylim(c(0, 35)) +
  labs(col = "")
```



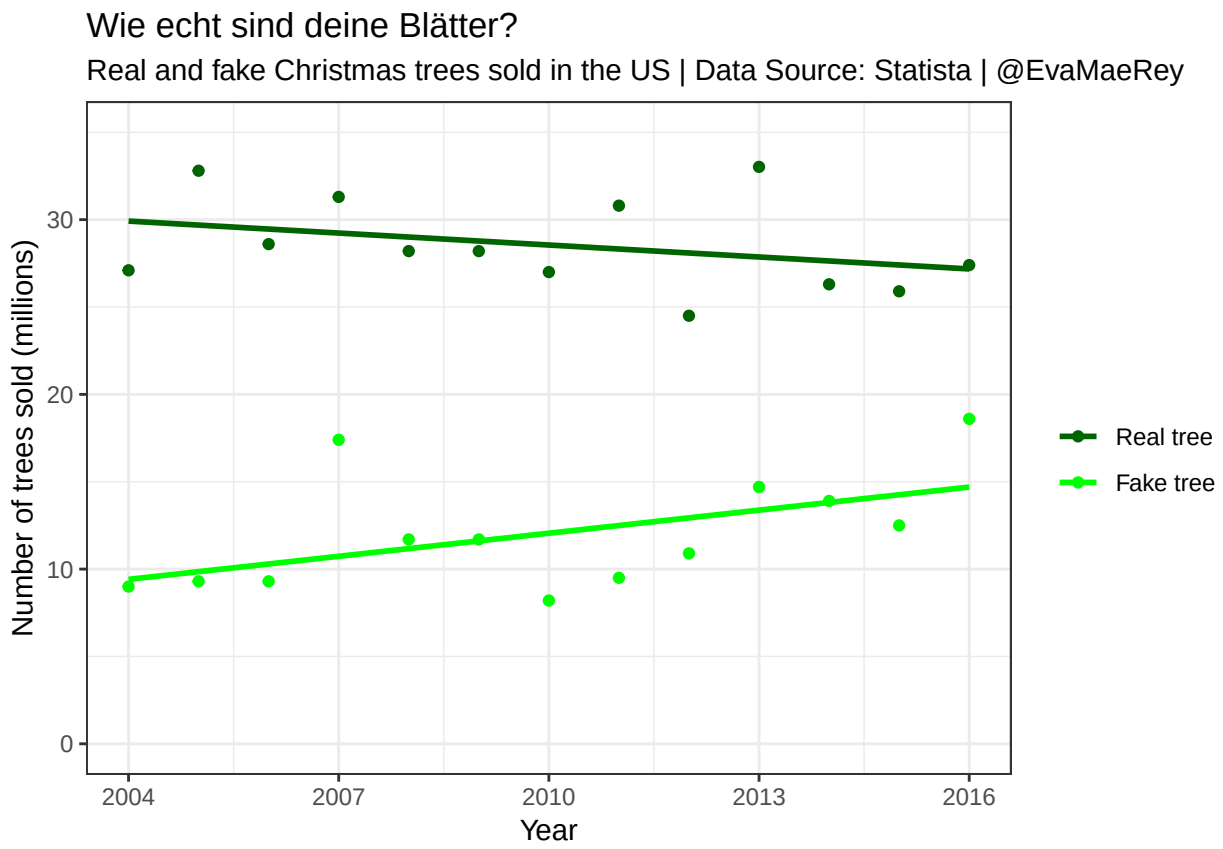
```
ggplot(data = dta) +
  aes(Year) +
  aes(y = `Number of trees sold (millions)`) +
  geom_point() +
  aes(col = fct_rev(`Type of tree`)) +
  geom_smooth(method = "lm", se = F) +
  scale_color_manual(values = c("darkgreen", "green")) +
  ylim(c(0, 35)) +
  labs(col = "") +
  labs(title = "Wie echt sind deine Blätter?")
```



```
ggplot(data = dta) +
  aes(Year) +
  aes(y = `Number of trees sold (millions)`) +
  geom_point() +
  aes(col = fct_rev(`Type of tree`)) +
  geom_smooth(method = "lm", se = F) +
  scale_color_manual(values = c("darkgreen", "green")) +
  ylim(c(0, 35)) +
  labs(col = "") +
  labs(title = "Wie echt sind deine Blätter?") +
  labs(subtitle = "Real and fake Christmas trees sold in the US | Data Source: Statista | @EvaMaeRey ")
```

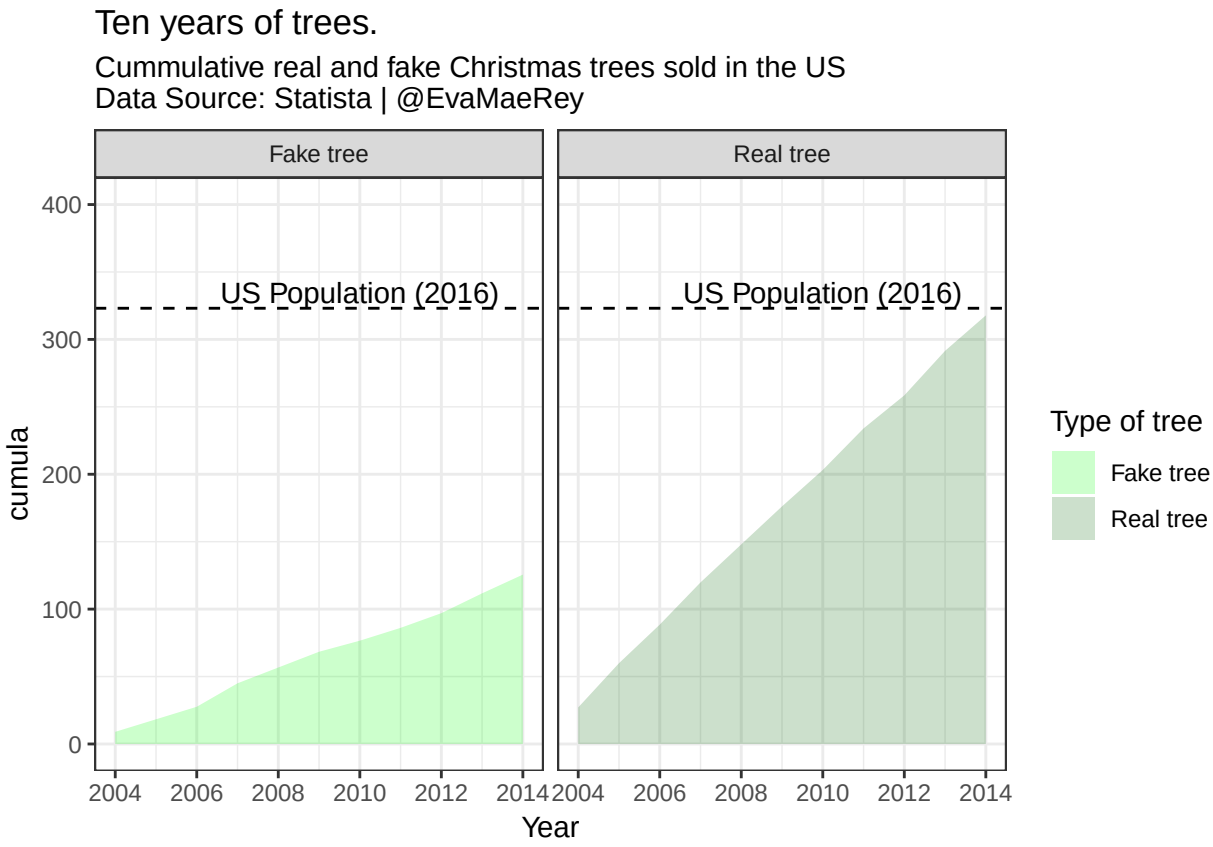


```
ggplot(data = dta) +
  aes(Year) +
  aes(y = `Number of trees sold (millions)`) +
  geom_point() +
  aes(col = fct_rev(`Type of tree`)) +
  geom_smooth(method = "lm", se = F) +
  scale_color_manual(values = c("darkgreen", "green")) +
  ylim(c(0, 35)) +
  labs(col = "") +
  labs(title = "Wie echt sind deine Blätter?") +
  labs(subtitle = "Real and fake Christmas trees sold in the US | Data Source: Statista | @EvaMaeRey ")
  theme_bw()
```




```
dta <- dta %>%
  group_by(`Type of tree`) %>%
  mutate(cumula = cumsum(`Number of trees sold (millions)`))

ggplot(dta %>% filter(Year <= 2014)) +
  aes(Year) +
  aes(y = cumula) +
  aes(fill = `Type of tree`) +
  geom_hline(yintercept = 323.1, lty = 2) +
  geom_area(alpha = .2) + facet_wrap(~ `Type of tree`) +
  annotate(geom = "text", x = 2010, y = 335, label = "US Population (2016)") +
  labs(title = "Ten years of trees.") +
  labs(subtitle = "Cummulative real and fake Christmas trees sold in the US\nData Source: Statista | @E
  scale_fill_manual(values = c("green", "darkgreen")) +
```

```
  theme_bw() +
  ylim(c(0, 400))
```



Chapter 4

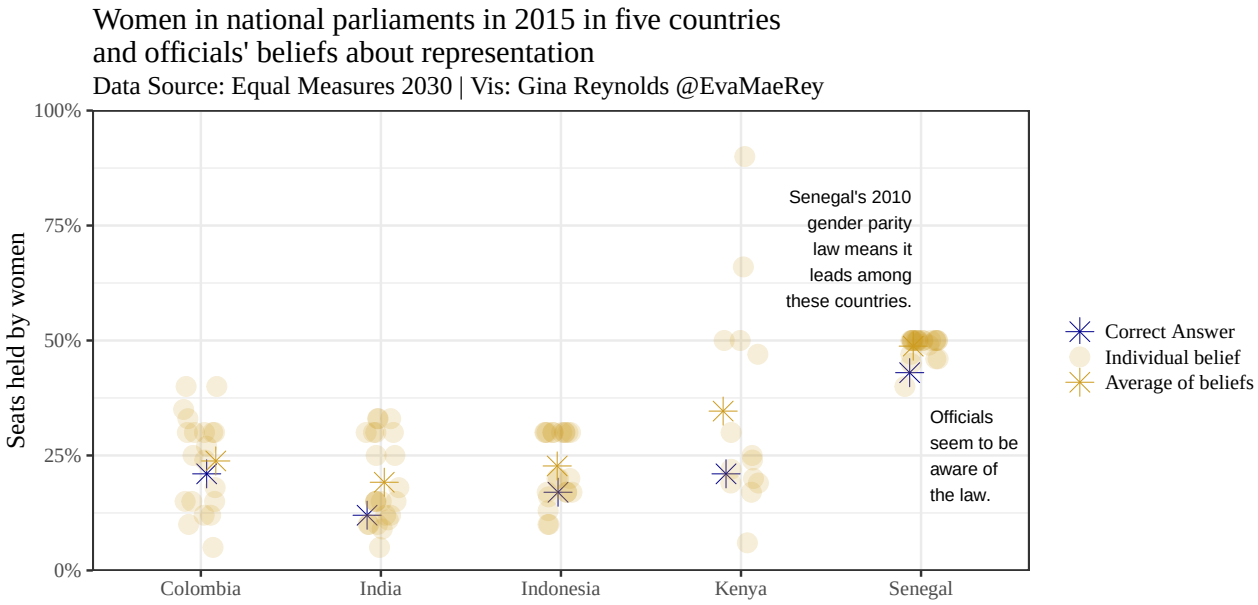
Officials' beliefs about women's representation

The data provided is based on a small survey of elite officials in five less developed countries. The question that arrises from the data is: How well do elites know the conditions in their countries. In general, the elites overestimate women's representation. But this is not the case in Senegal, where there are gender quotas in the Parliament. Most elites therefore estimate that the representation is about equal with men. I jitter the responses of the elites horizontally to avoid overplotting.

A random sample from the data set:

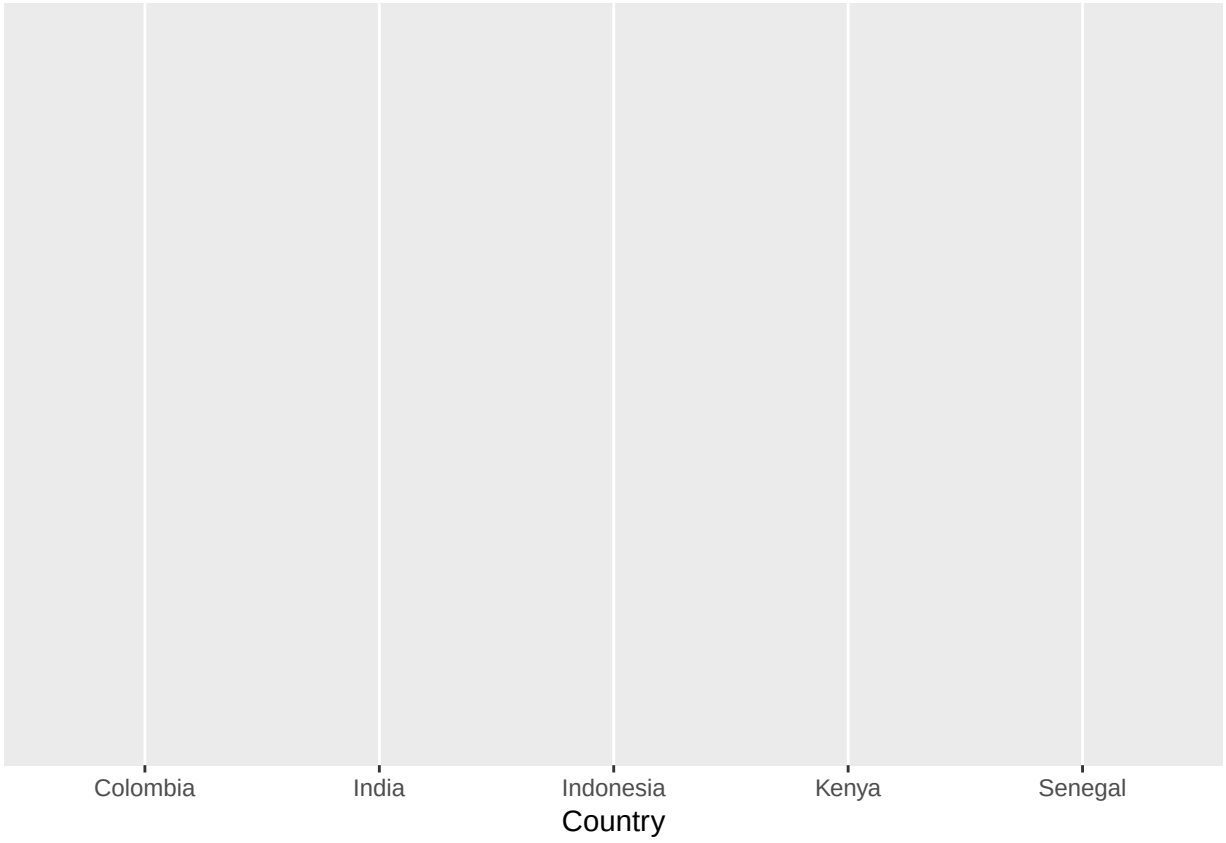
Country	Topic	value	value_type	alpha
Senegal	Share of seats held by women	0.50	Individual belief	0.3
Senegal	Share of seats held by women	0.50	Individual belief	0.3
Kenya	Share of seats held by women	0.19	Individual belief	0.3
Colombia	Share of seats held by women	0.35	Individual belief	0.3
Indonesia	Share of seats held by women	0.30	Individual belief	0.3

```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times") +
  scale_y_continuous(limits = c(0, 1), expand = c(0, 0), labels = scales::percent) +
  scale_colour_manual(name = "", values = c("darkblue", "goldenrod3", "goldenrod3")) +
  scale_alpha_manual(name = "", values = c(1, .17, 1)) +
  scale_shape_manual(name = "", values = c(8, 19, 8)) +
  annotate(geom = "text", x = 4.95, y = .70, label = str_wrap("Senegal's 2010 gender parity law means i
  annotate(geom = "text", x = 5.05, y = .250, label = str_wrap("Officials seem to be aware of the law."
  labs(x = "") +
  labs(y = "Seats held by women") +
  labs(title = "Women in national parliaments in 2015 in five countries \nand officials' beliefs about
  labs(subtitle = "Data Source: Equal Measures 2030 | Vis: Gina Reynolds @EvaMaeRey")
```

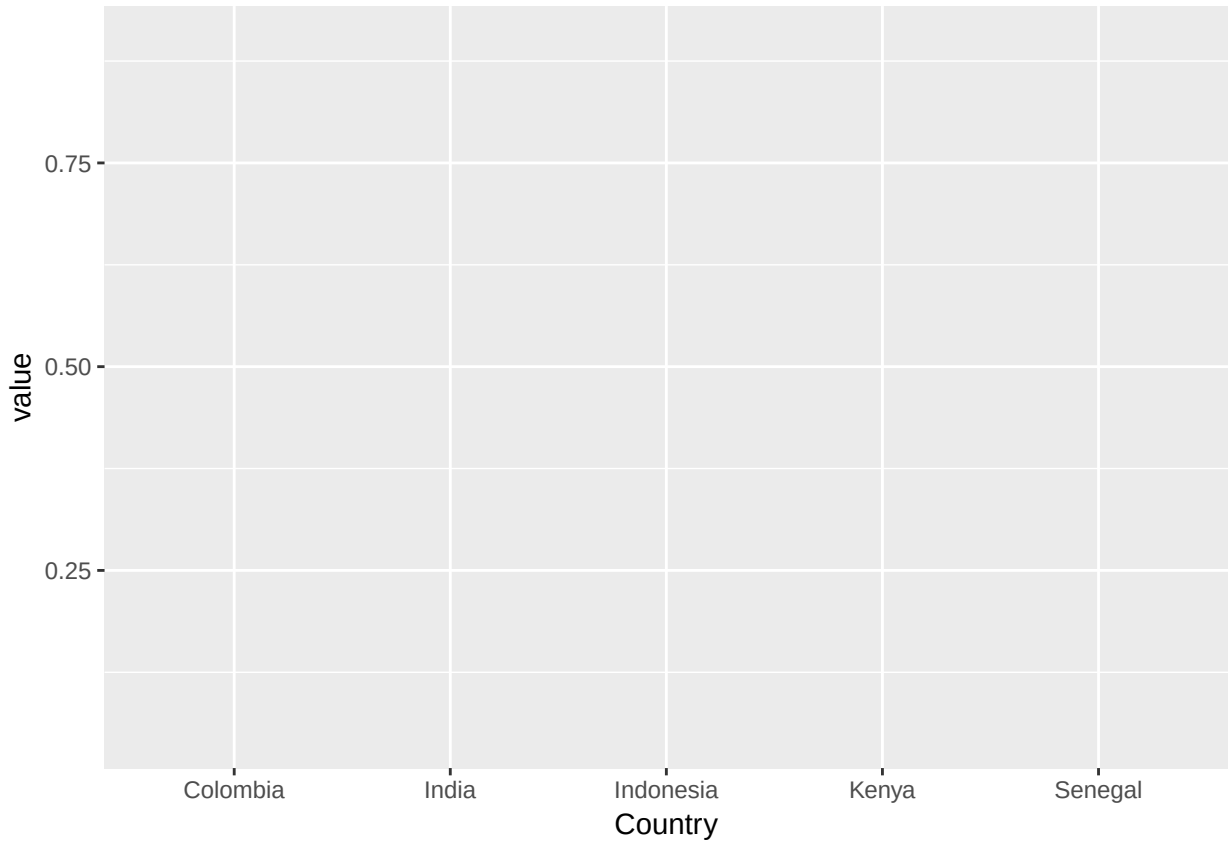

Women in national parliaments in 2015 in five countries
and officials' beliefs about representation
Data Source: Equal Measures 2030 | Vis: Gina Reynolds @EvaMaeRey
Seats held by women
100%
75%
50%
25%
0%
Colombia
India
Indonesia
Kenya
Senegal
Senegal's 2010 gender parity law means it leads among these countries.
Officials seem to be aware of the law.
Correct Answer
Individual belief
Average of beliefs

```
ggplot(data = df_all)
```

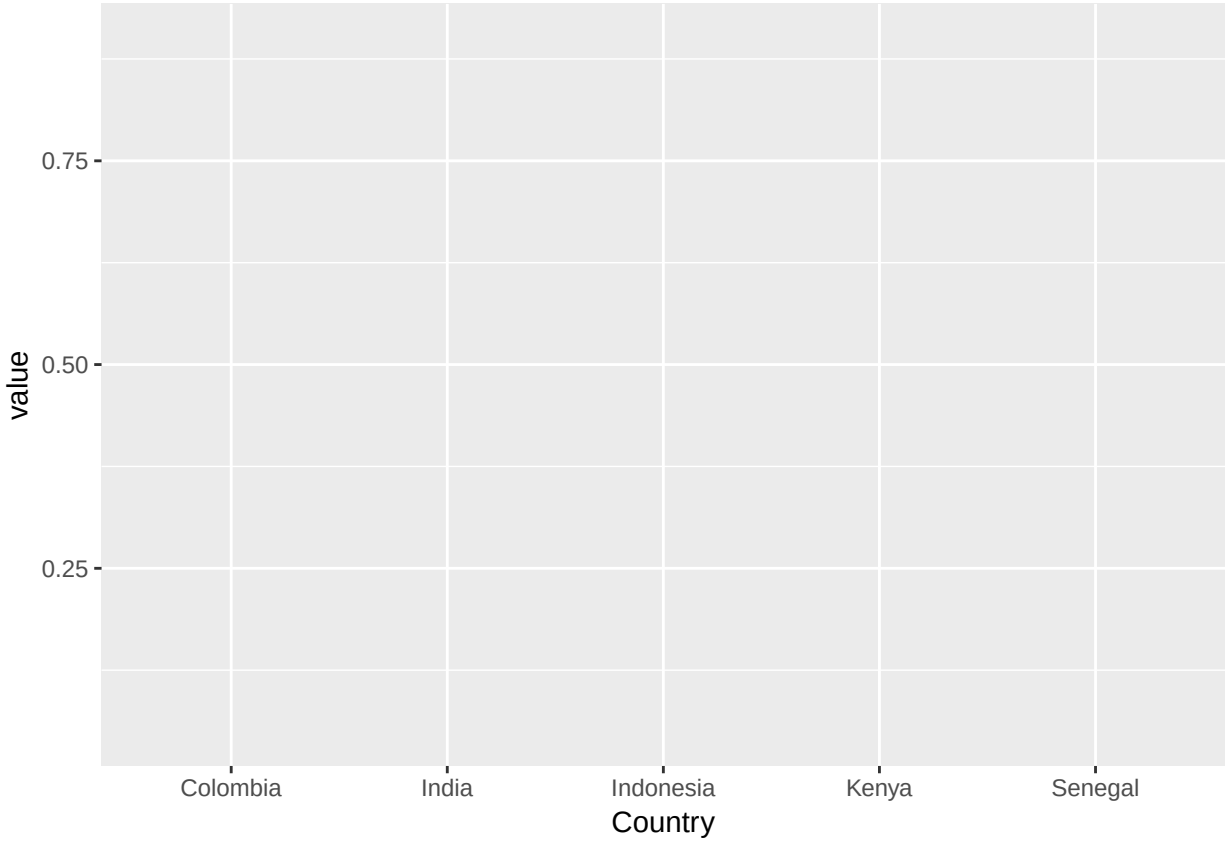
```
ggplot(data = df_all) +
  aes(x = Country)
```



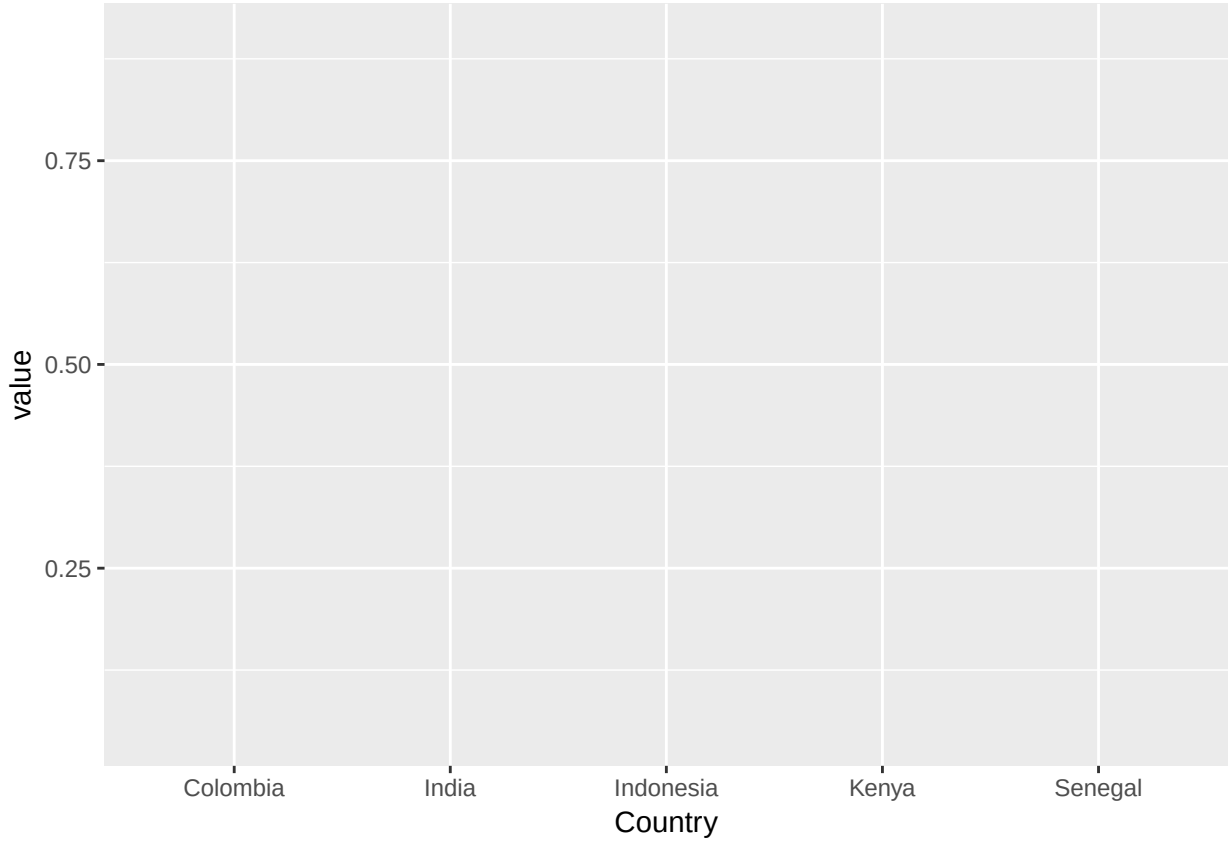
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value)
```



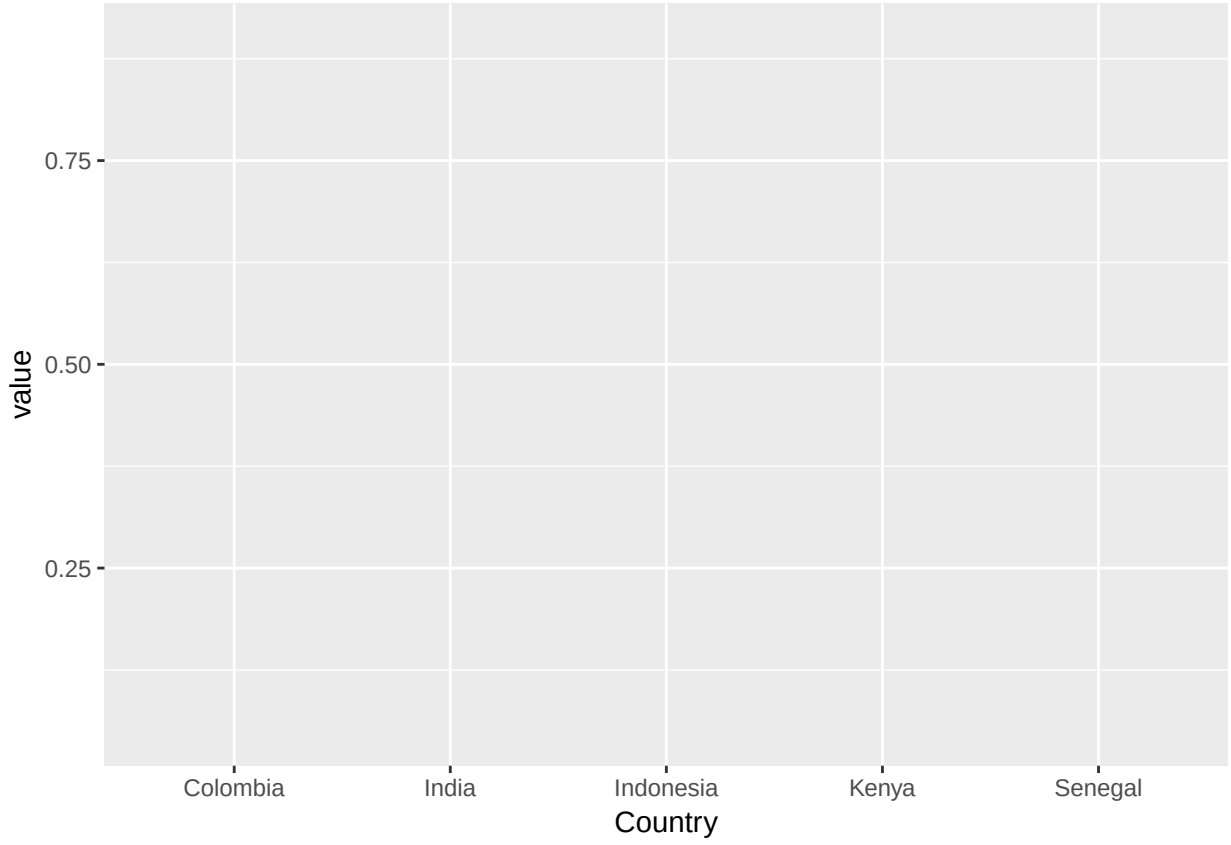
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type))
```



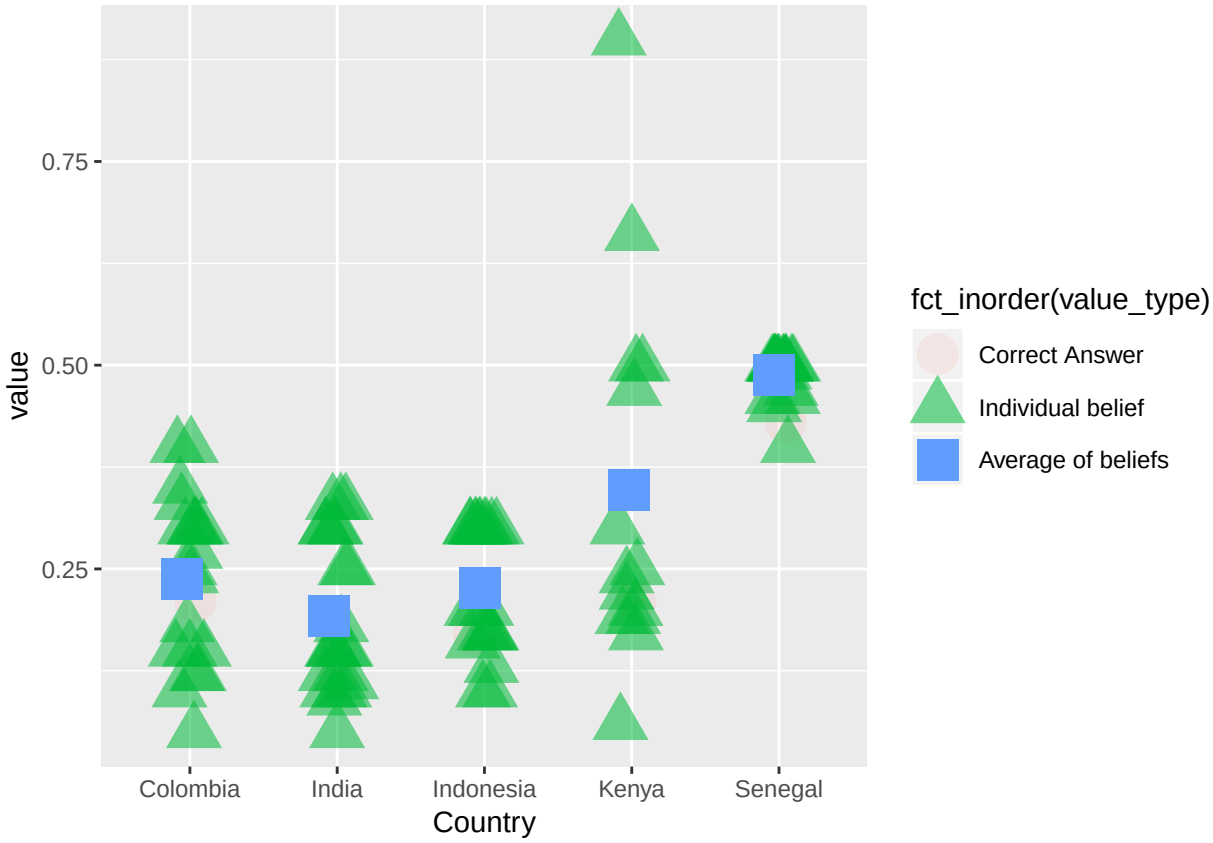
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type))
```



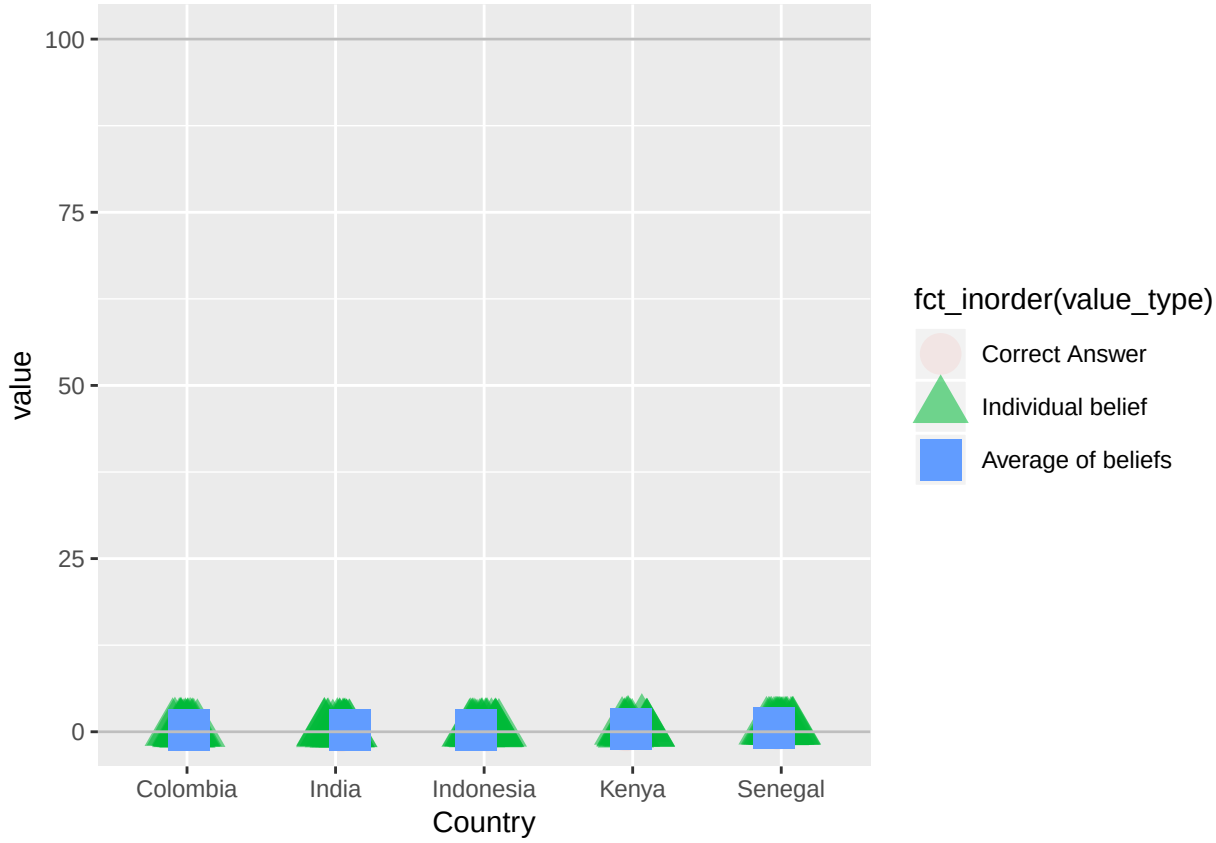
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type))
```



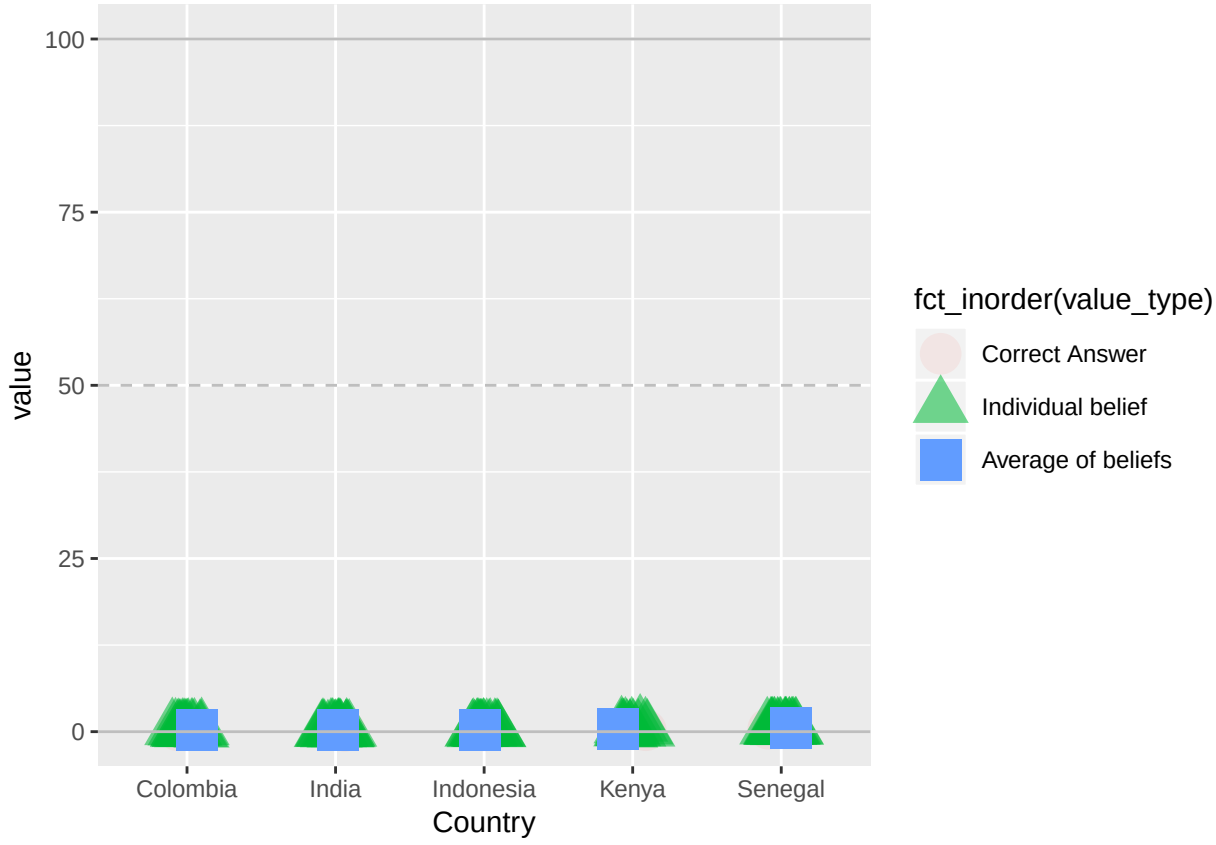
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7)
```



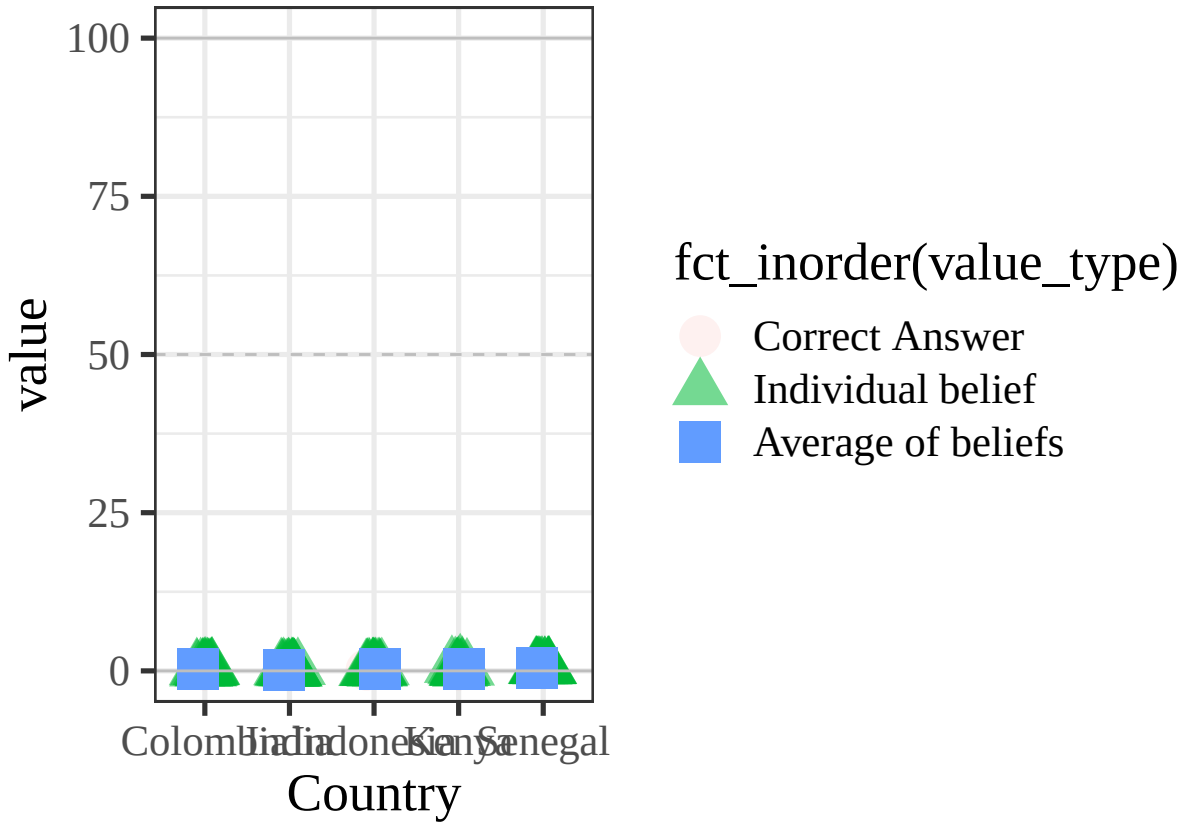
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey")
```



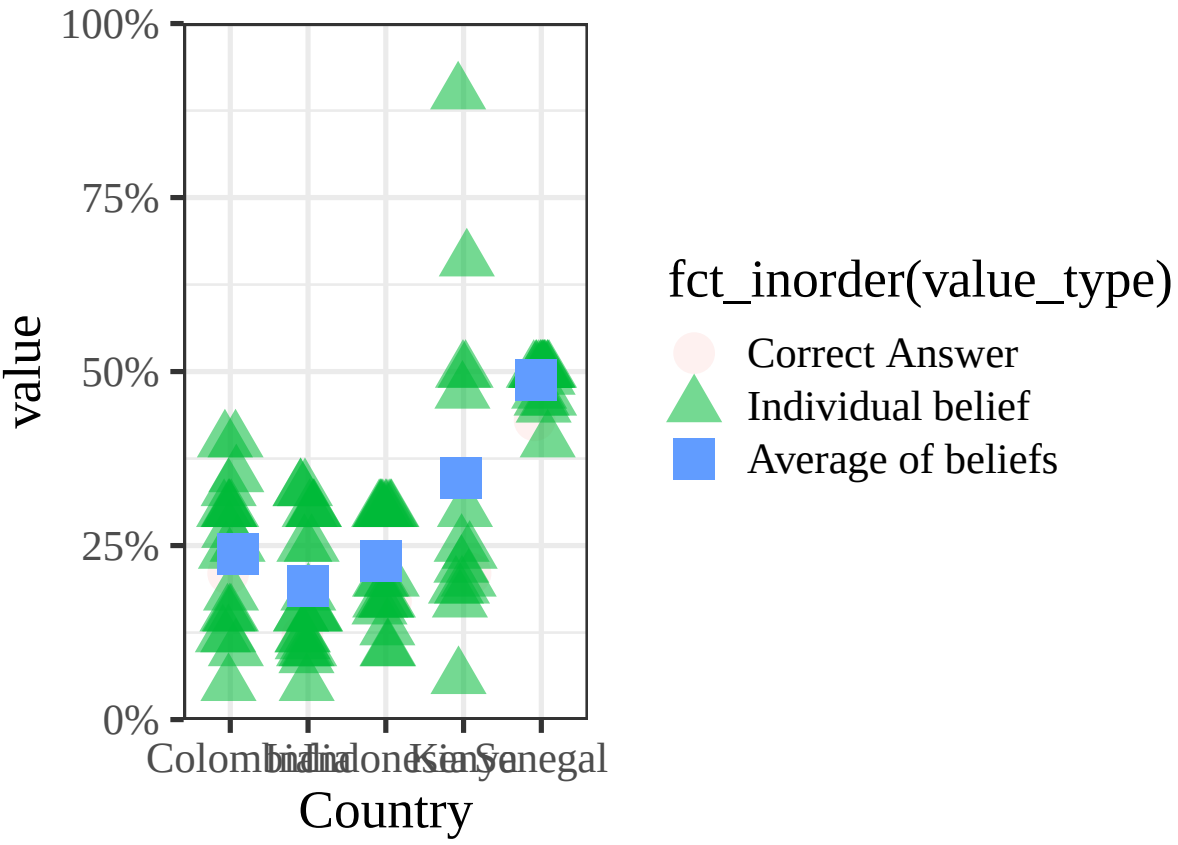
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey")
```



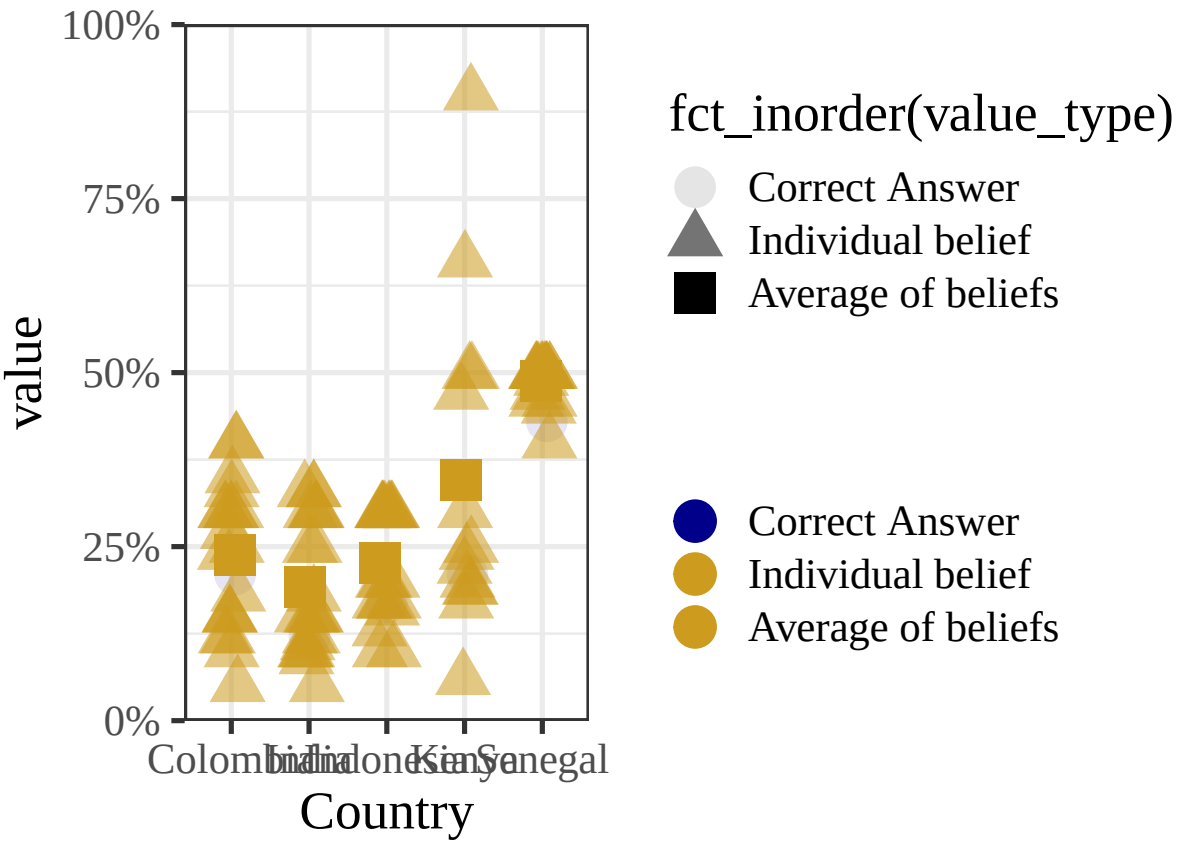
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times")
```



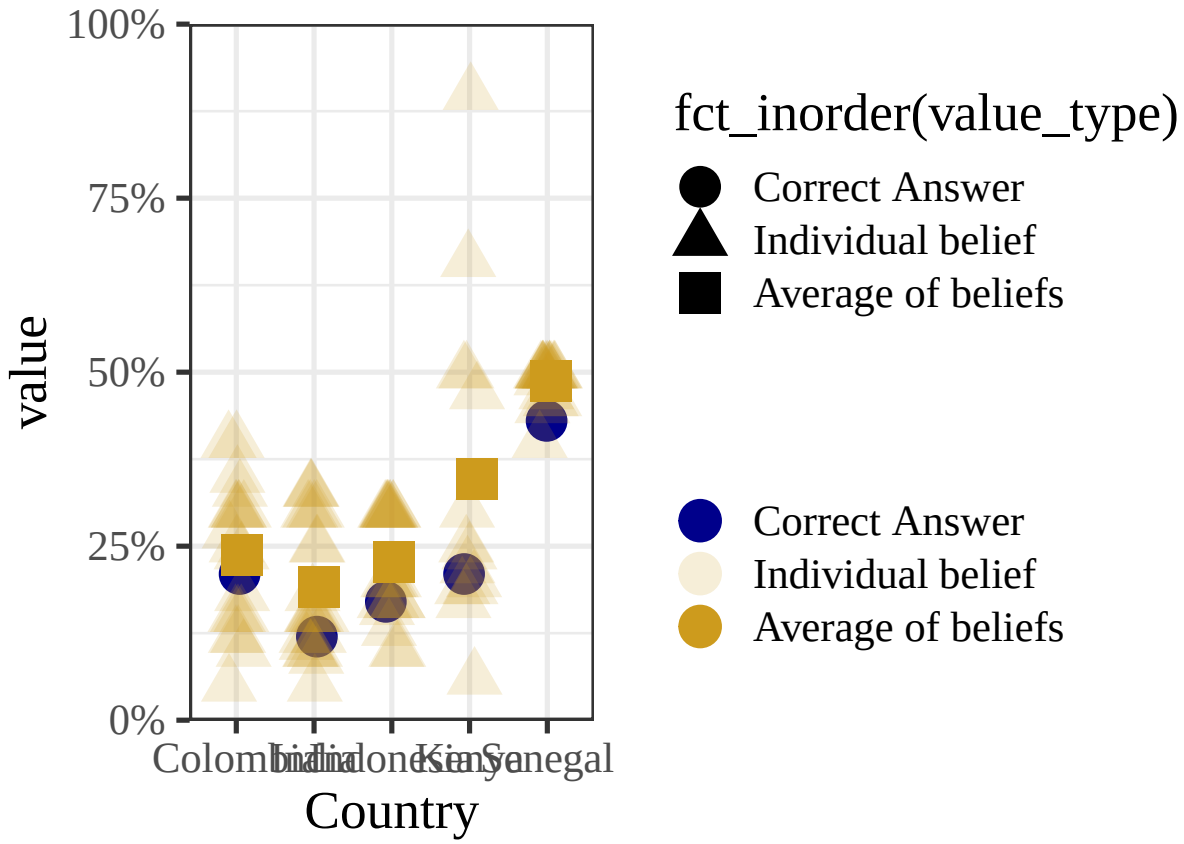
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times") +
  scale_y_continuous(limits = c(0, 1), expand = c(0, 0), labels = scales::percent)
```



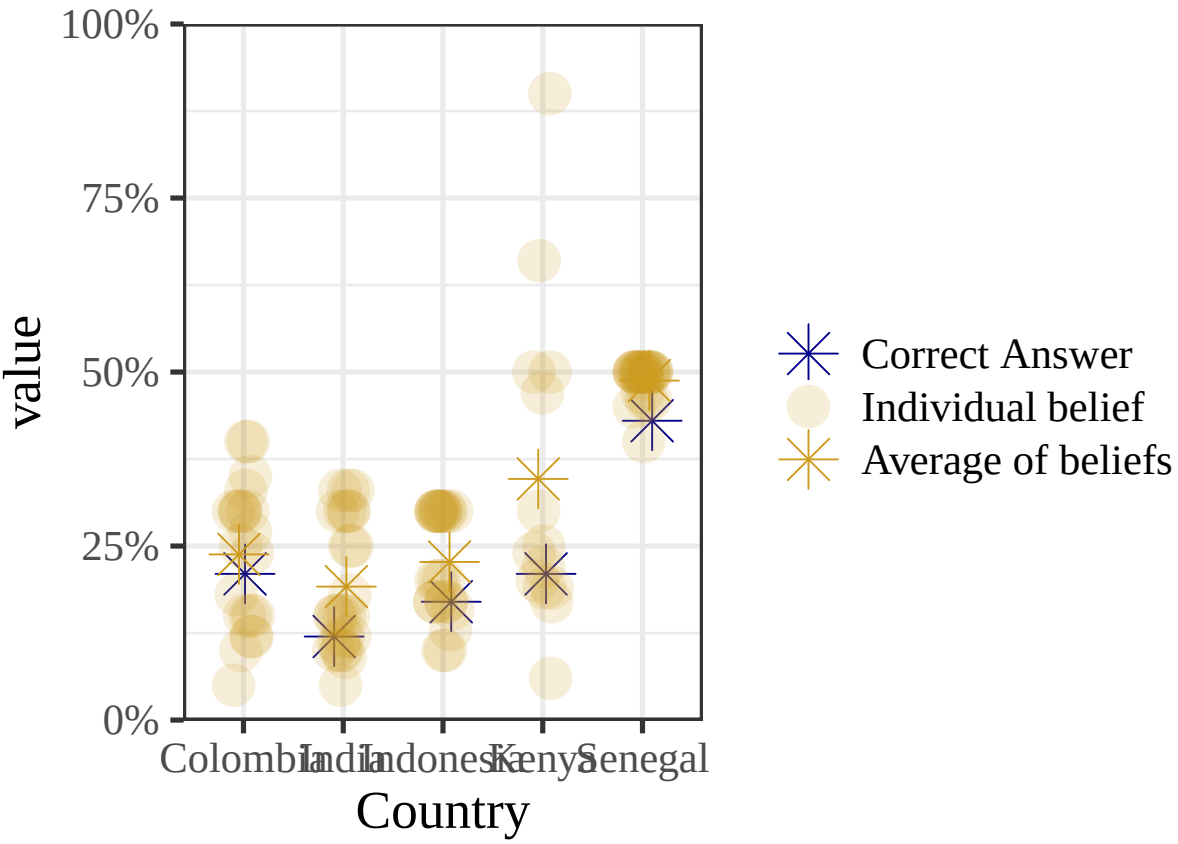
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times") +
  scale_y_continuous(limits = c(0, 1), expand = c(0, 0), labels = scales::percent) +
  scale_colour_manual(name = "", values = c("darkblue", "goldenrod3", "goldenrod3"))
```



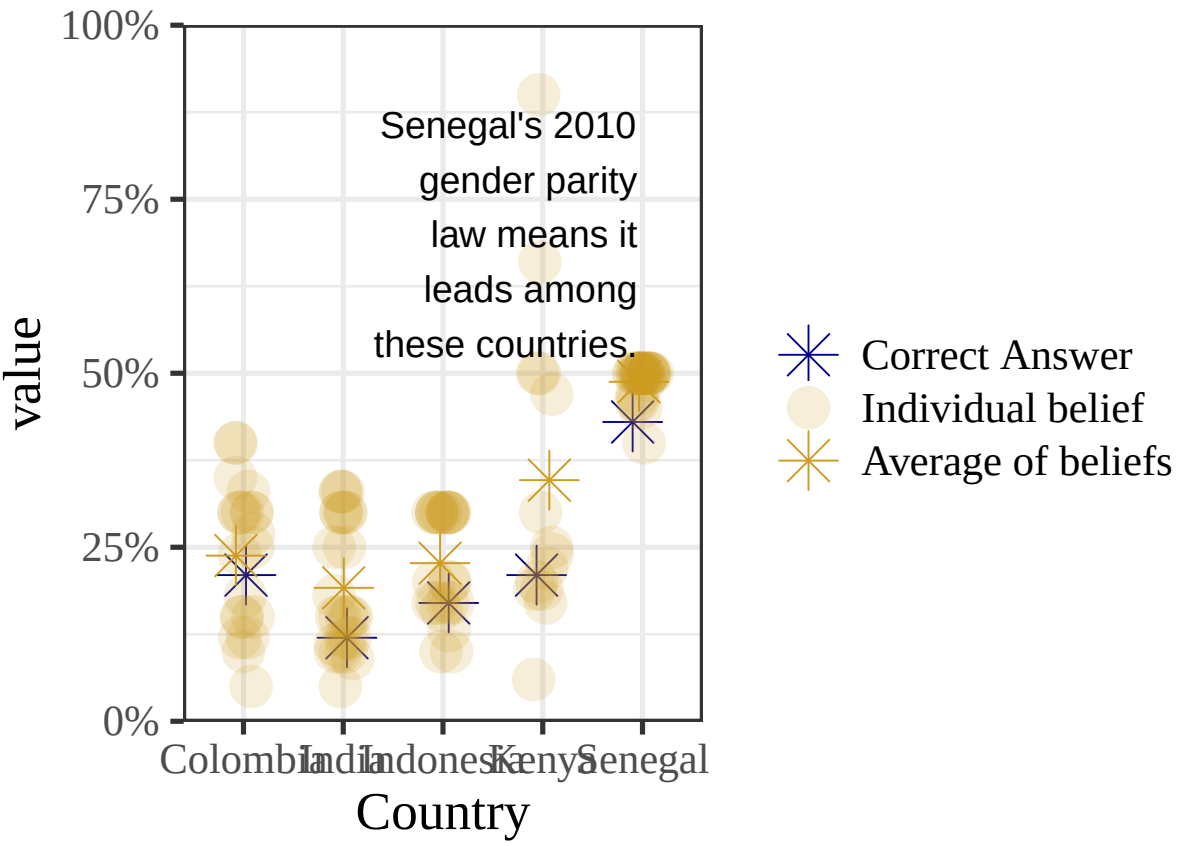
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times") +
  scale_y_continuous(limits = c(0, 1), expand = c(0, 0), labels = scales::percent) +
  scale_colour_manual(name = "", values = c("darkblue", "goldenrod3", "goldenrod3")) +
  scale_alpha_manual(name = "", values = c(1, .17, 1))
```



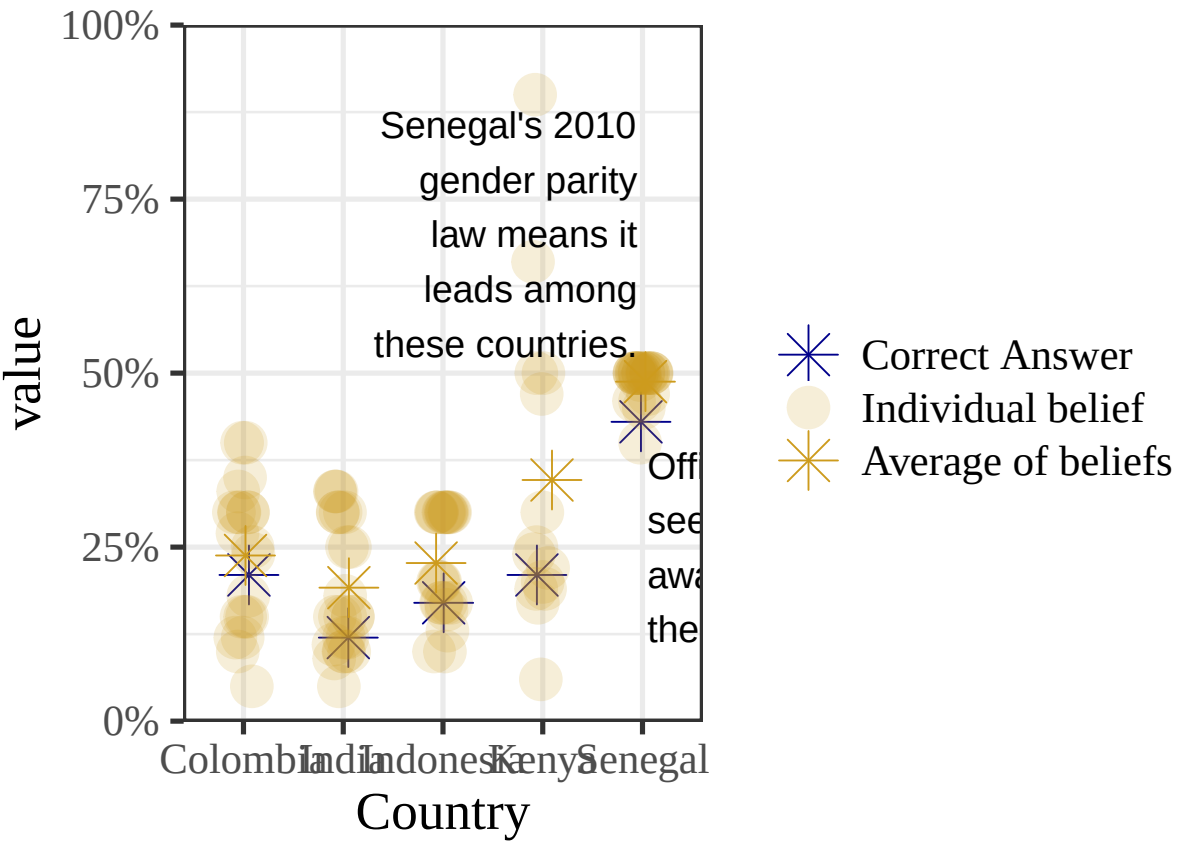
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times") +
  scale_y_continuous(limits = c(0, 1), expand = c(0, 0), labels = scales::percent) +
  scale_colour_manual(name = "", values = c("darkblue", "goldenrod3", "goldenrod3")) +
  scale_alpha_manual(name = "", values = c(1, .17, 1)) +
  scale_shape_manual(name = "", values = c(8, 19, 8))
```



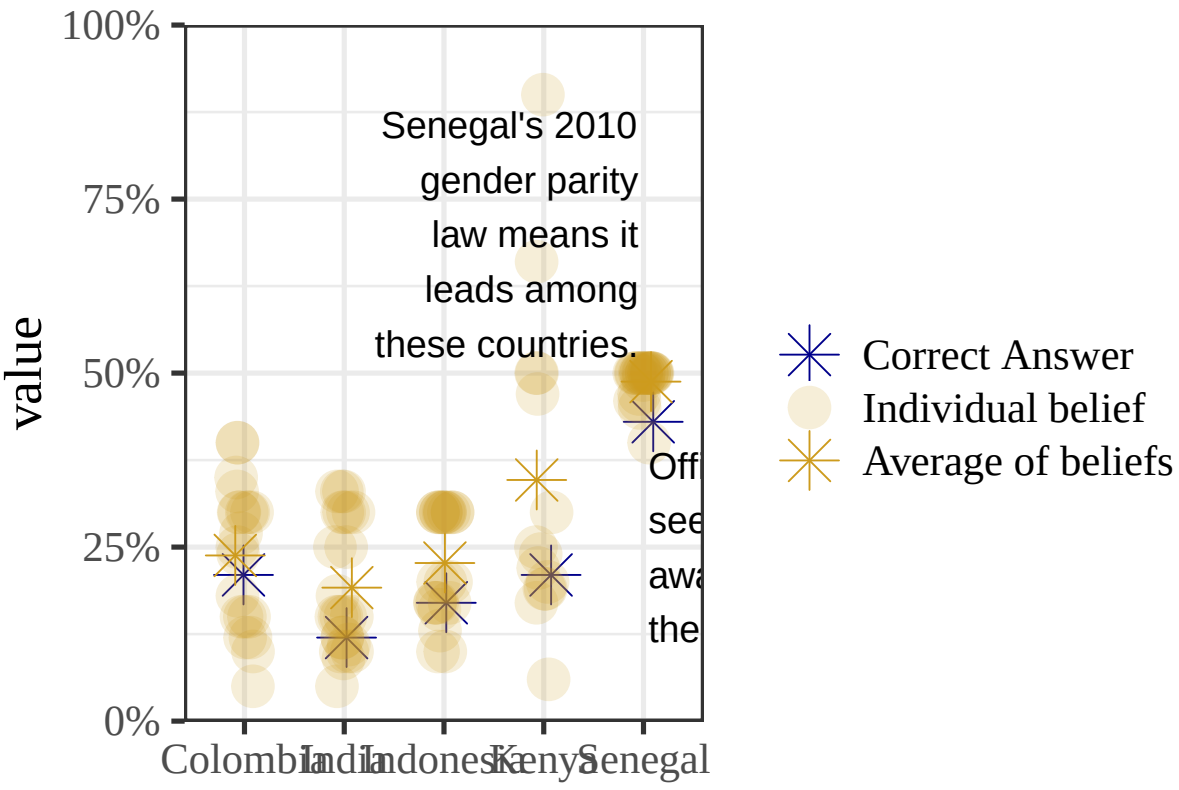
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times") +
  scale_y_continuous(limits = c(0, 1), expand = c(0, 0), labels = scales::percent) +
  scale_colour_manual(name = "", values = c("darkblue", "goldenrod3", "goldenrod3")) +
  scale_alpha_manual(name = "", values = c(1, .17, 1)) +
  scale_shape_manual(name = "", values = c(8, 19, 8)) +
  annotate(geom = "text", x = 4.95, y = .70, label = str_wrap("Senegal's 2010 gender parity law means i
```



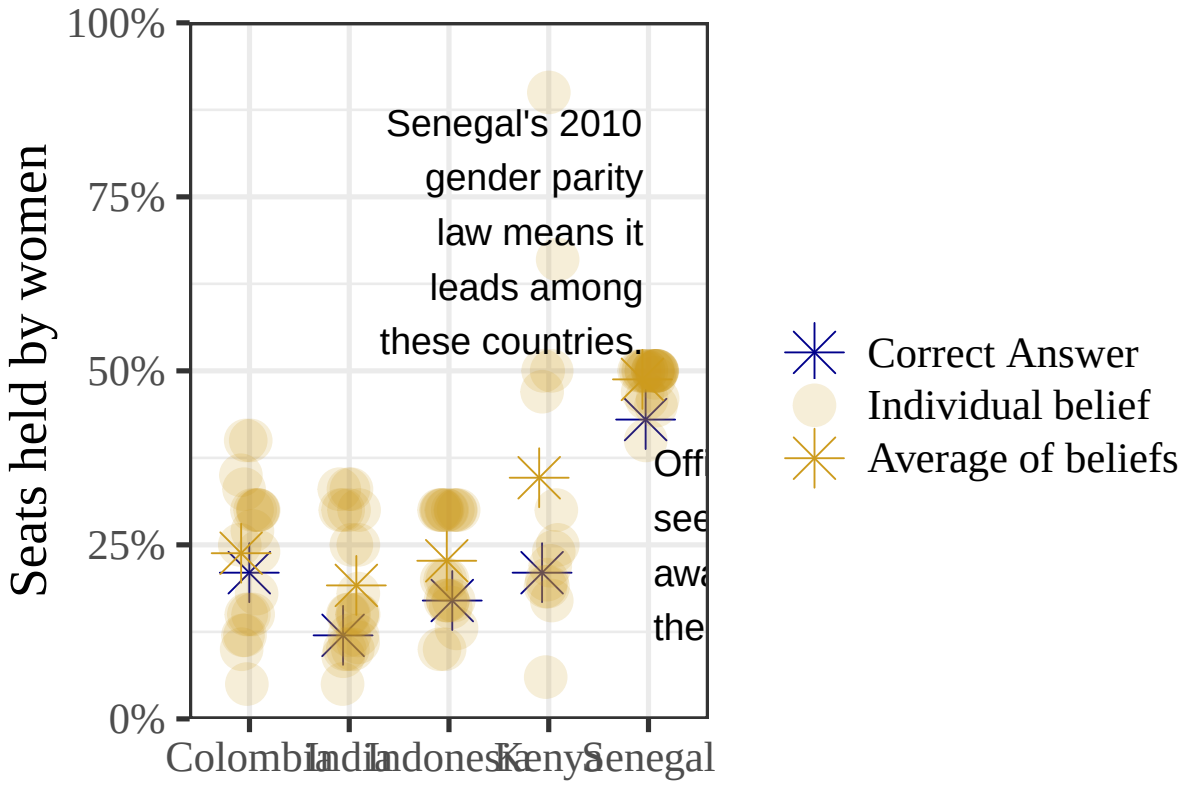
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times") +
  scale_y_continuous(limits = c(0, 1), expand = c(0, 0), labels = scales::percent) +
  scale_colour_manual(name = "", values = c("darkblue", "goldenrod3", "goldenrod3")) +
  scale_alpha_manual(name = "", values = c(1, .17, 1)) +
  scale_shape_manual(name = "", values = c(8, 19, 8)) +
  annotate(geom = "text", x = 4.95, y = .70, label = str_wrap("Senegal's 2010 gender parity law means i
  annotate(geom = "text", x = 5.05, y = .250, label = str_wrap("Officials seem to be aware of the law."
```



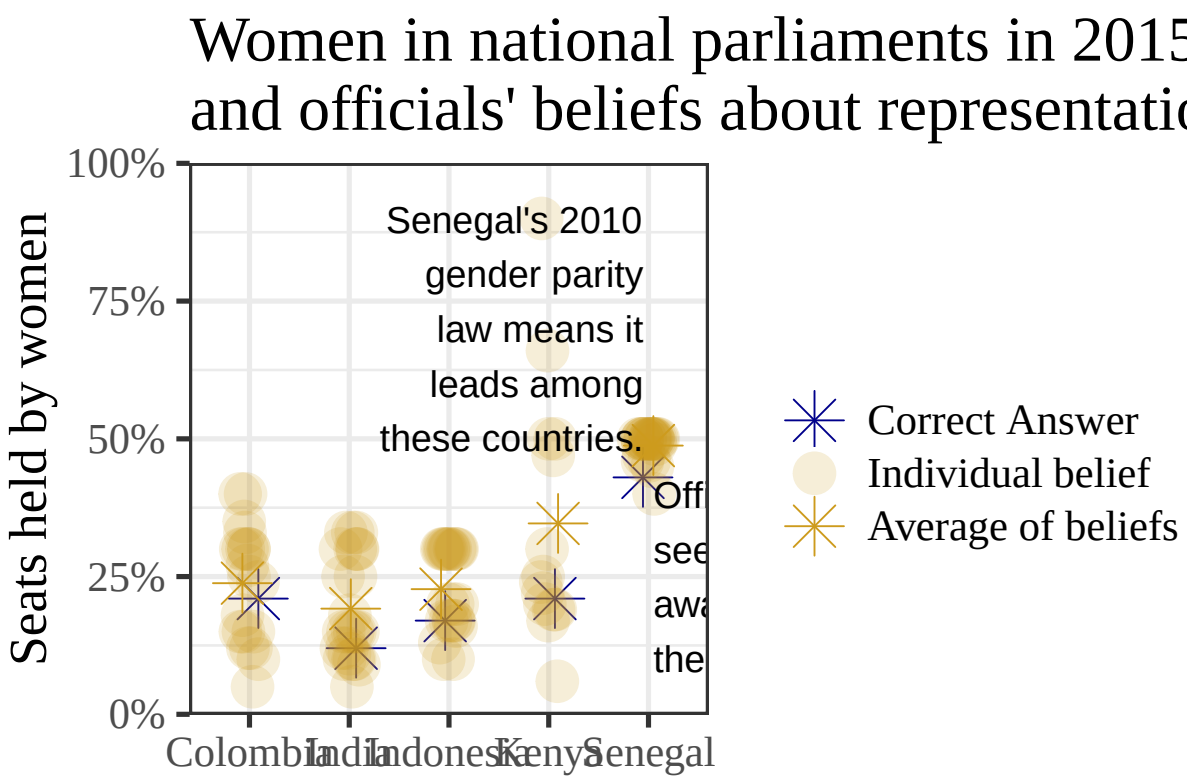
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times") +
  scale_y_continuous(limits = c(0, 1), expand = c(0, 0), labels = scales::percent) +
  scale_colour_manual(name = "", values = c("darkblue", "goldenrod3", "goldenrod3")) +
  scale_alpha_manual(name = "", values = c(1, .17, 1)) +
  scale_shape_manual(name = "", values = c(8, 19, 8)) +
  annotate(geom = "text", x = 4.95, y = .70, label = str_wrap("Senegal's 2010 gender parity law means it
  annotate(geom = "text", x = 5.05, y = .250, label = str_wrap("Officials seem to be aware of the law."
  labs(x = "")
```



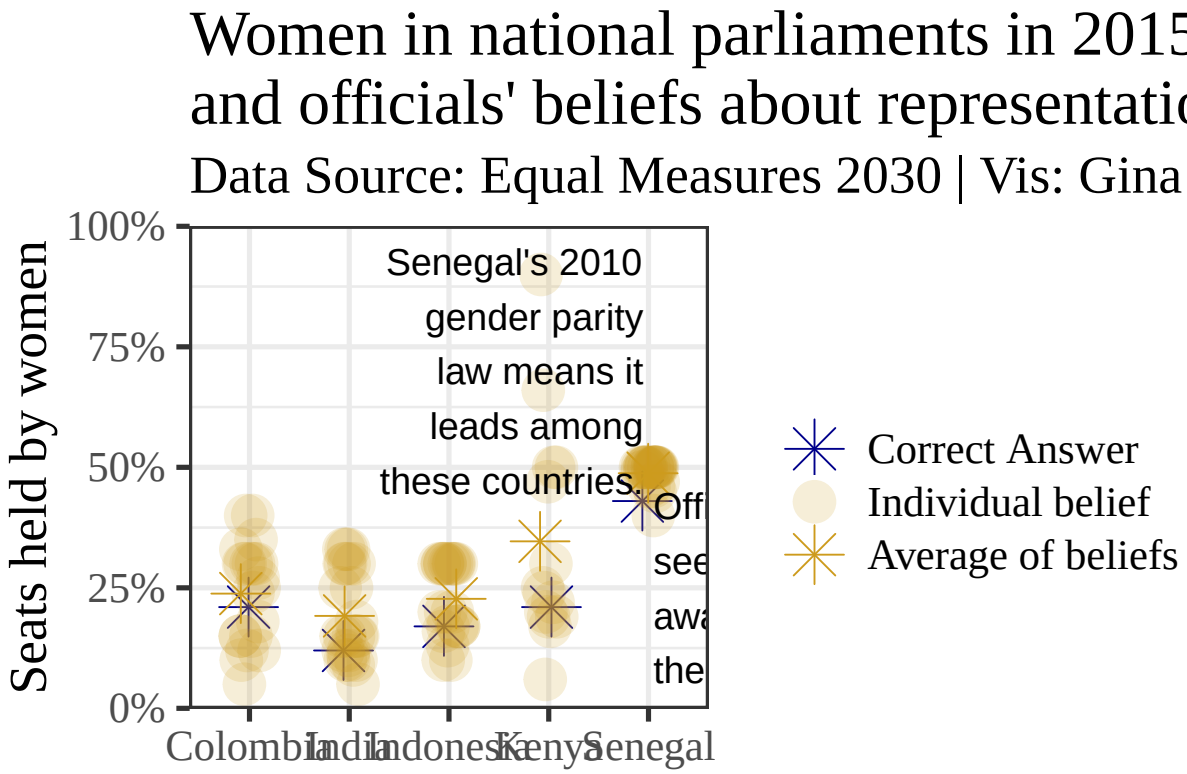
```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times") +
  scale_y_continuous(limits = c(0, 1), expand = c(0, 0), labels = scales::percent) +
  scale_colour_manual(name = "", values = c("darkblue", "goldenrod3", "goldenrod3")) +
  scale_alpha_manual(name = "", values = c(1, .17, 1)) +
  scale_shape_manual(name = "", values = c(8, 19, 8)) +
  annotate(geom = "text", x = 4.95, y = .70, label = str_wrap("Senegal's 2010 gender parity law means it
  annotate(geom = "text", x = 5.05, y = .250, label = str_wrap("Officials seem to be aware of the law."
  labs(x = "") +
  labs(y = "Seats held by women")
```



```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times") +
  scale_y_continuous(limits = c(0, 1), expand = c(0, 0), labels = scales::percent) +
  scale_colour_manual(name = "", values = c("darkblue", "goldenrod3", "goldenrod3")) +
  scale_alpha_manual(name = "", values = c(1, .17, 1)) +
  scale_shape_manual(name = "", values = c(8, 19, 8)) +
  annotate(geom = "text", x = 4.95, y = .70, label = str_wrap("Senegal's 2010 gender parity law means it
  annotate(geom = "text", x = 5.05, y = .250, label = str_wrap("Officials seem to be aware of the law."
  labs(x = "") +
  labs(y = "Seats held by women") +
  labs(title = "Women in national parliaments in 2015 in five countries \nand officials' beliefs about
```



```
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  geom_hline(yintercept = c(50), lty = 2, col = "grey") +
  theme_bw(base_size = 20, base_family = "Times") +
  scale_y_continuous(limits = c(0, 1), expand = c(0, 0), labels = scales::percent) +
  scale_colour_manual(name = "", values = c("darkblue", "goldenrod3", "goldenrod3")) +
  scale_alpha_manual(name = "", values = c(1, .17, 1)) +
  scale_shape_manual(name = "", values = c(8, 19, 8)) +
  annotate(geom = "text", x = 4.95, y = .70, label = str_wrap("Senegal's 2010 gender parity law means it
  annotate(geom = "text", x = 5.05, y = .250, label = str_wrap("Officials seem to be aware of the law."
  labs(x = "") +
  labs(y = "Seats held by women") +
  labs(title = "Women in national parliaments in 2015 in five countries \nand officials' beliefs about
  labs(subtitle = "Data Source: Equal Measures 2030 | Vis: Gina Reynolds @EvaMaeRey")
```



Chapter 5

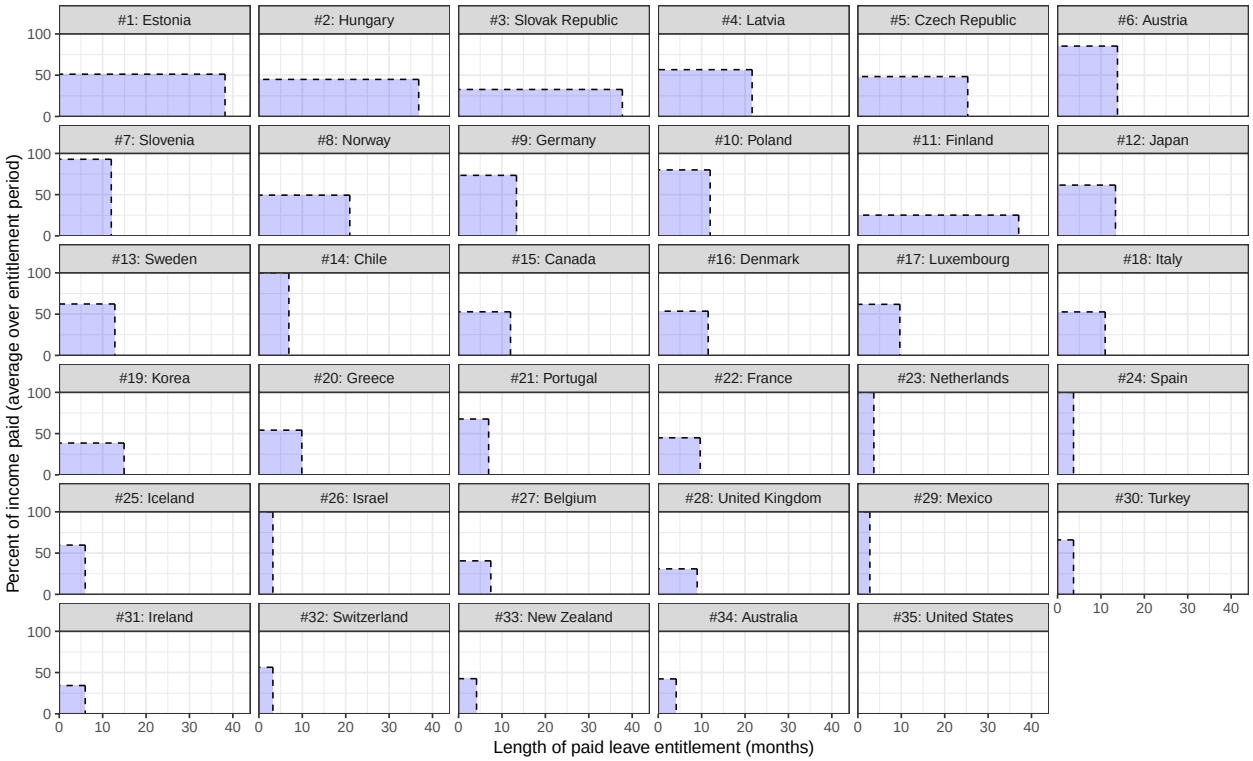
Maternal Leave

The OECD provides a comparative report on how much paid leave women are entitled to after childbirth. But leave takes different forms. In some places, the allowed leave is longer, but sometimes that means that the pay out compared to the regular salary is lower. To emphasize the different forms that law around paid leave take, I plotted the total payout available to mothers as areas of rectangles, where one side is the length of leave allowed, and the other side is the proportion of salary paid to the new mom.

A random sample from the data set:

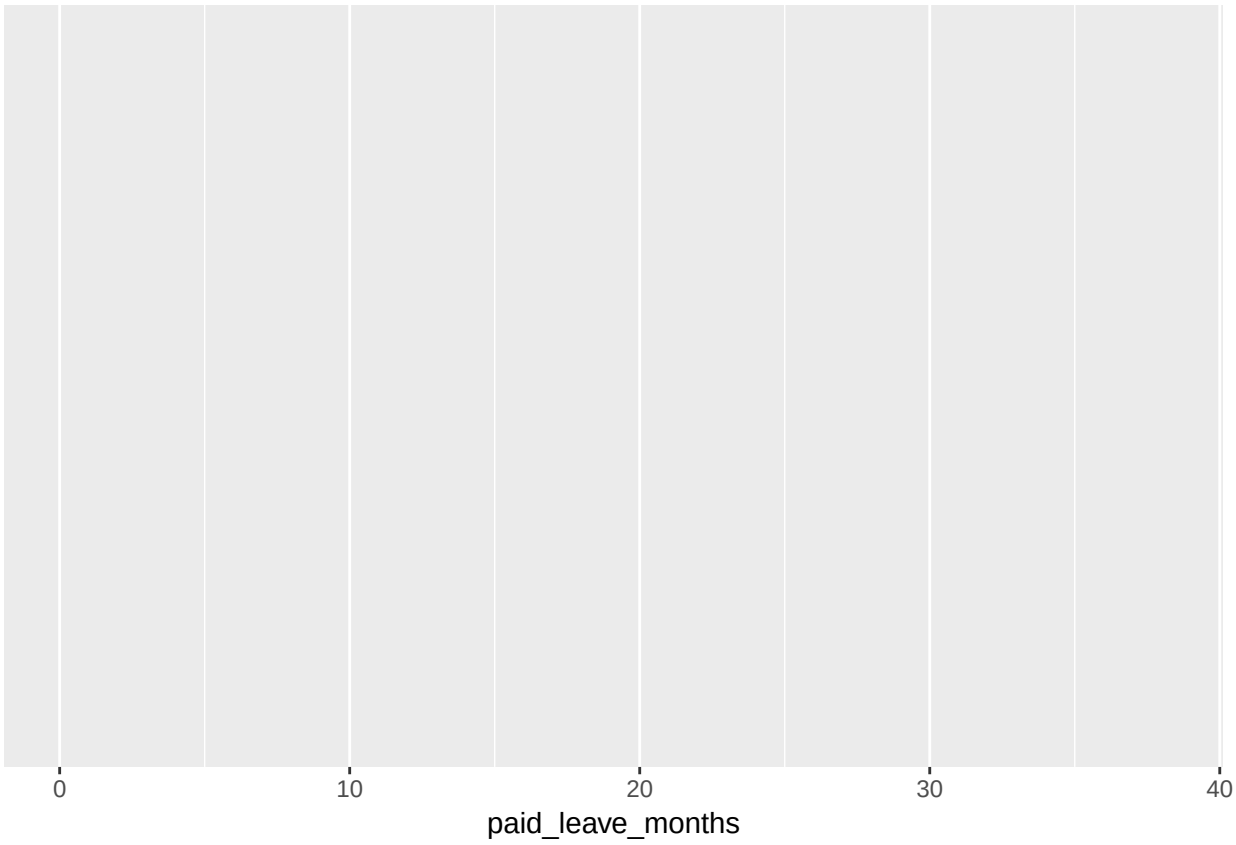
Country	Paid maternity leave avg payment rate (%)	Paid maternity leave full rate equivalent in weeks	Paid mate
Belgium	64.1	9.6	
Slovenia	100.0	15.0	
Chile	100.0	18.0	
Finland	74.4	13.0	
Switzerland	56.4	7.9	

```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0) +
  geom_segment(aes(yend = `Total paid leave avg payment rate (%)`), lty = "dashed") +
  geom_segment(aes(xend = paid_leave_months), lty = "dashed") +
  scale_y_continuous(limits = c(0, 100), expand = c(0, 0), breaks = c(0, 50, 100)) +
  scale_x_continuous(limits = c(0, 44), expand = c(0, 0)) +
  labs(x = "Length of paid leave entitlement (months)") +
  labs(y = "Percent of income paid (average over entitlement period)") +
  labs(title = "Total paid leave available to mothers in the OECD") +
  labs(subtitle = "Countries rank ordered by paid leave full rate equivalent (blue rectangular area)\nV
  theme_bw(base_size = 12)
```

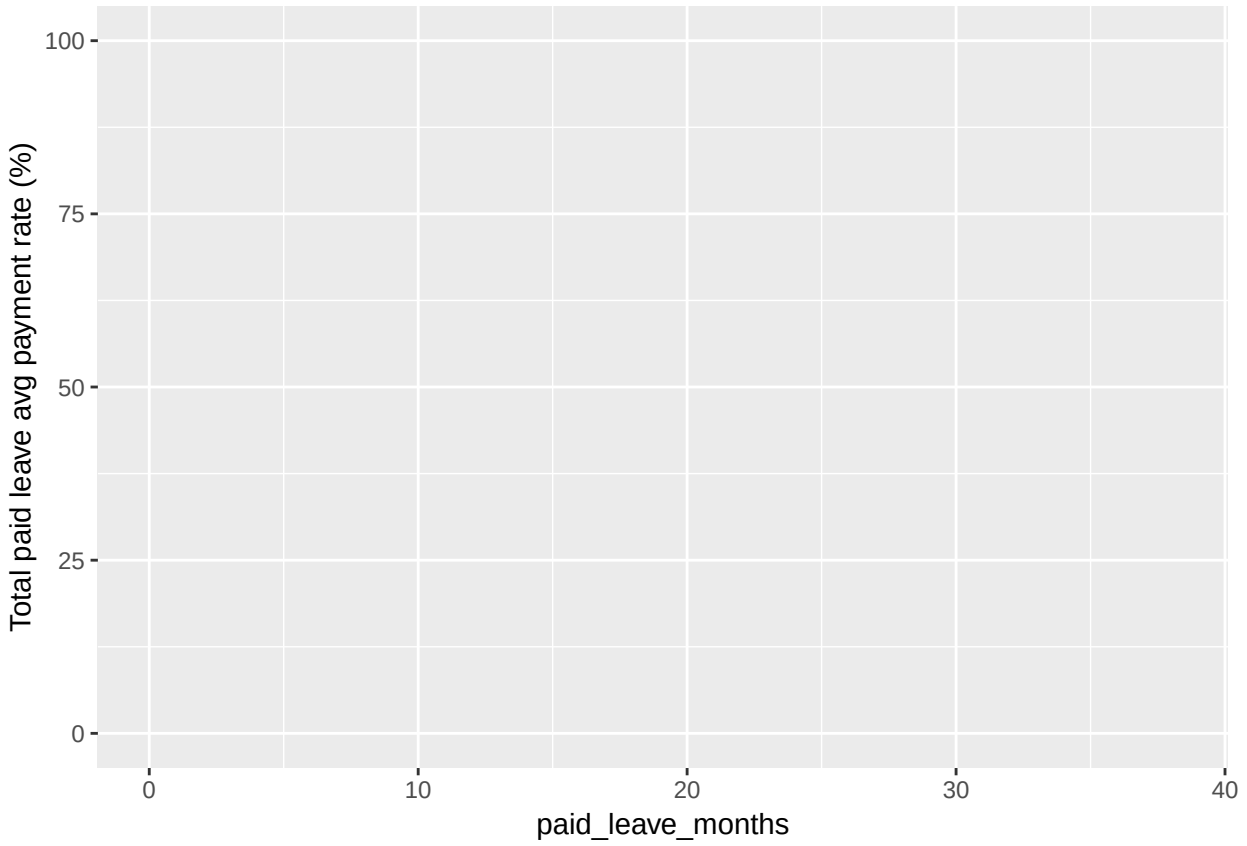

Total paid leave available to mothers in the OECD
Countries rank ordered by paid leave full rate equivalent (blue rectangular area)
Visualization: Gina Reynolds | Data source: OECD.org
#1: Estonia
#2: Hungary
#3: Slovak Republic
#4: Latvia
#5: Czech Republic
#6: Austria
#7: Slovenia
#8: Norway
#9: Germany
#10: Poland
#11: Finland
#12: Japan
#13: Sweden
#14: Chile
#15: Canada
#16: Denmark
#17: Luxembourg
#18: Italy
#19: Korea
#20: Greece
#21: Portugal
#22: France
#23: Netherlands
#24: Spain
#25: Iceland
#26: Israel
#27: Belgium
#28: United Kingdom
#29: Mexico
#30: Turkey
#31: Ireland
#32: Switzerland
#33: New Zealand
#34: Australia
#35: United States
Percent of income paid (average over entitlement period)
Length of paid leave entitlement (months)

```
ggplot(df)
```

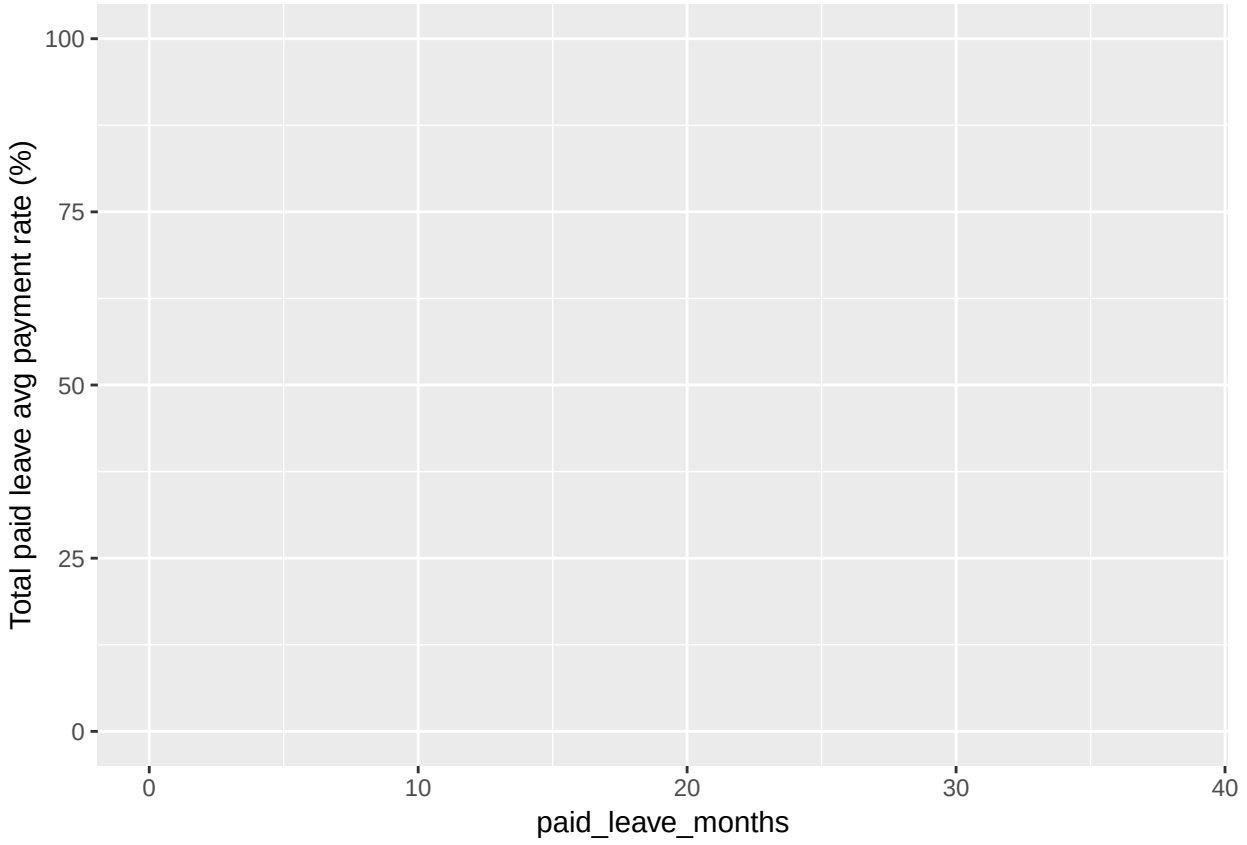
```
ggplot(df) +
  aes(x = paid_leave_months)
```



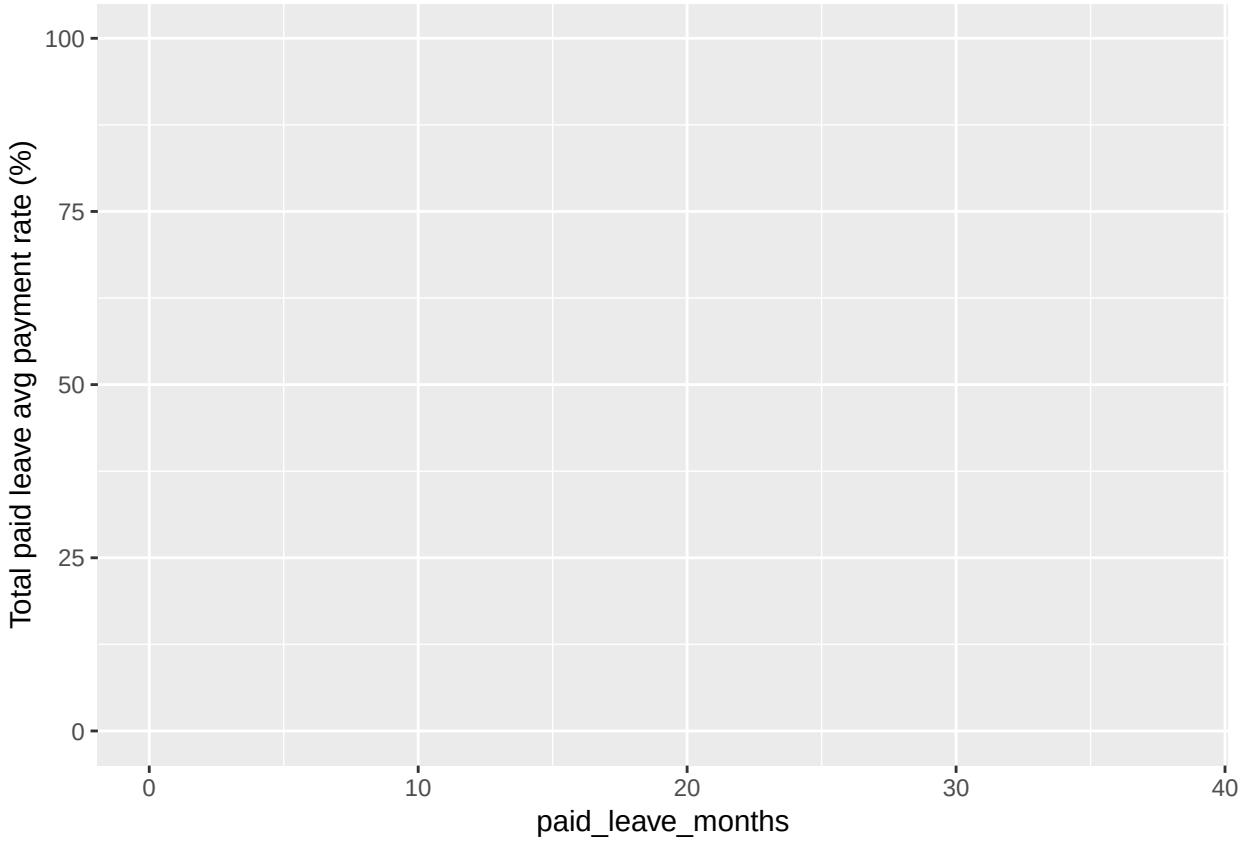
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`)
```



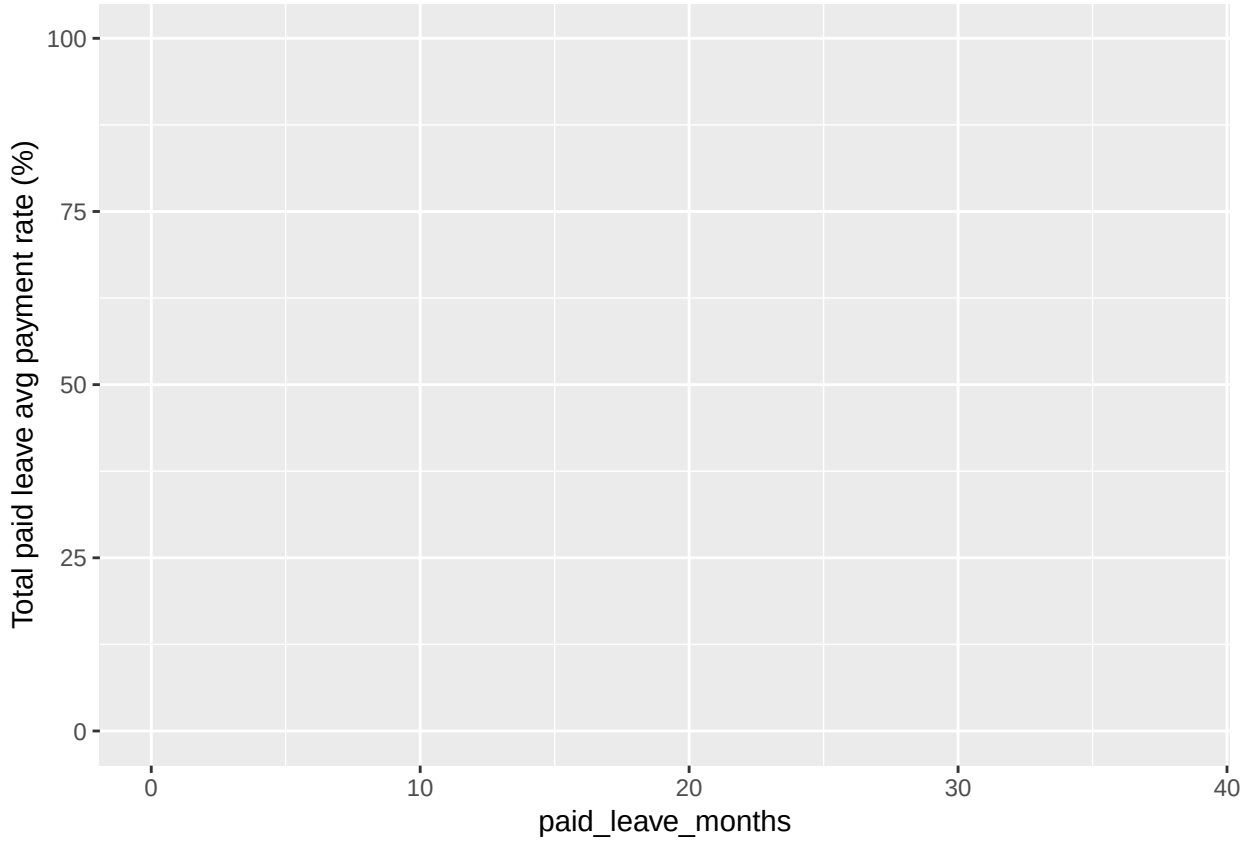
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0)
```



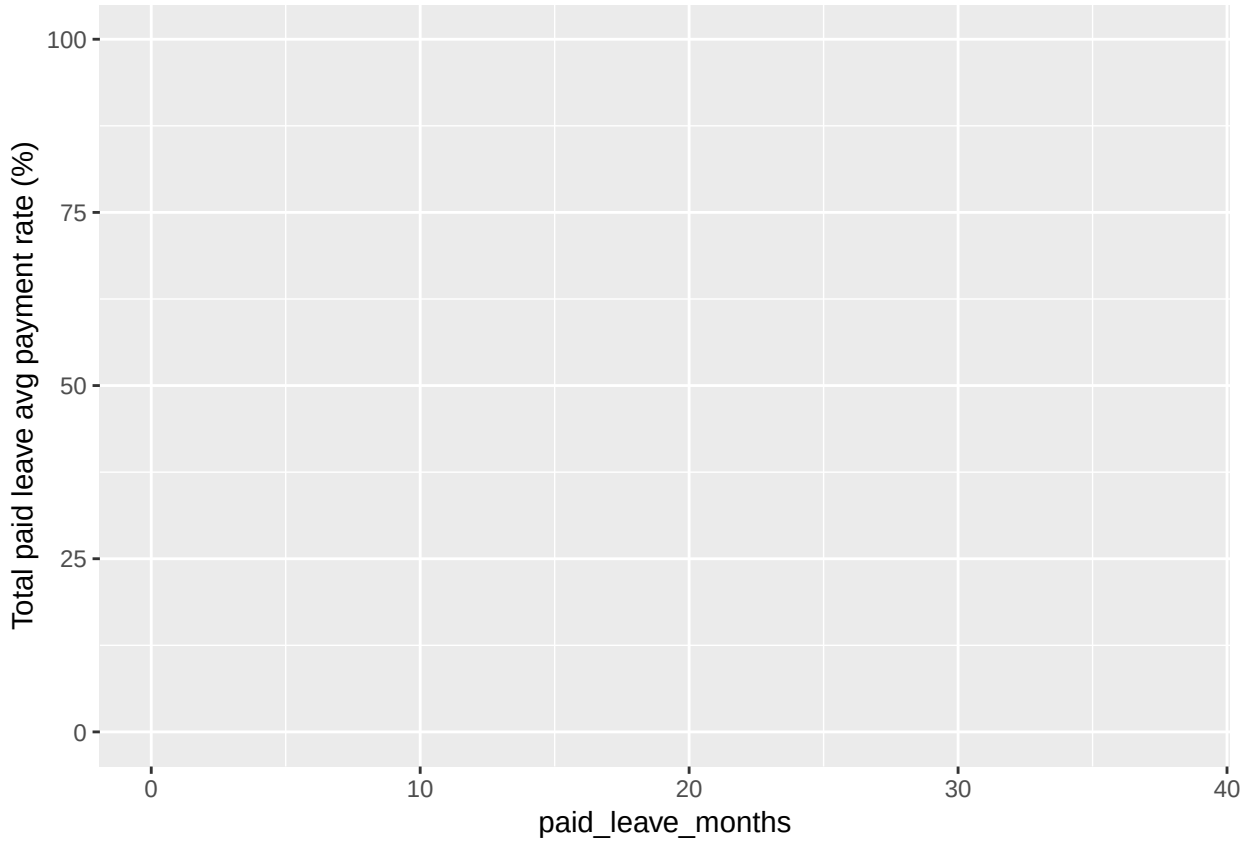
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months)
```



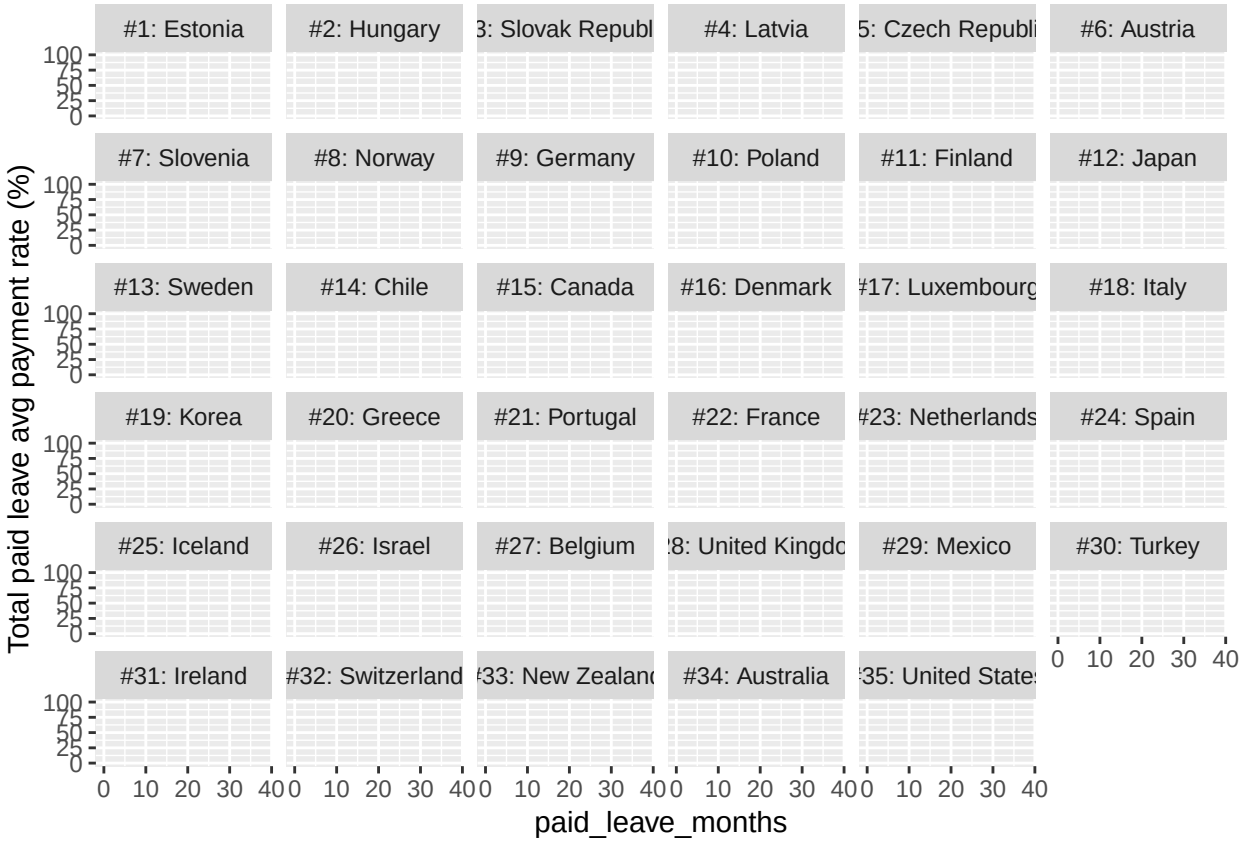
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0)
```



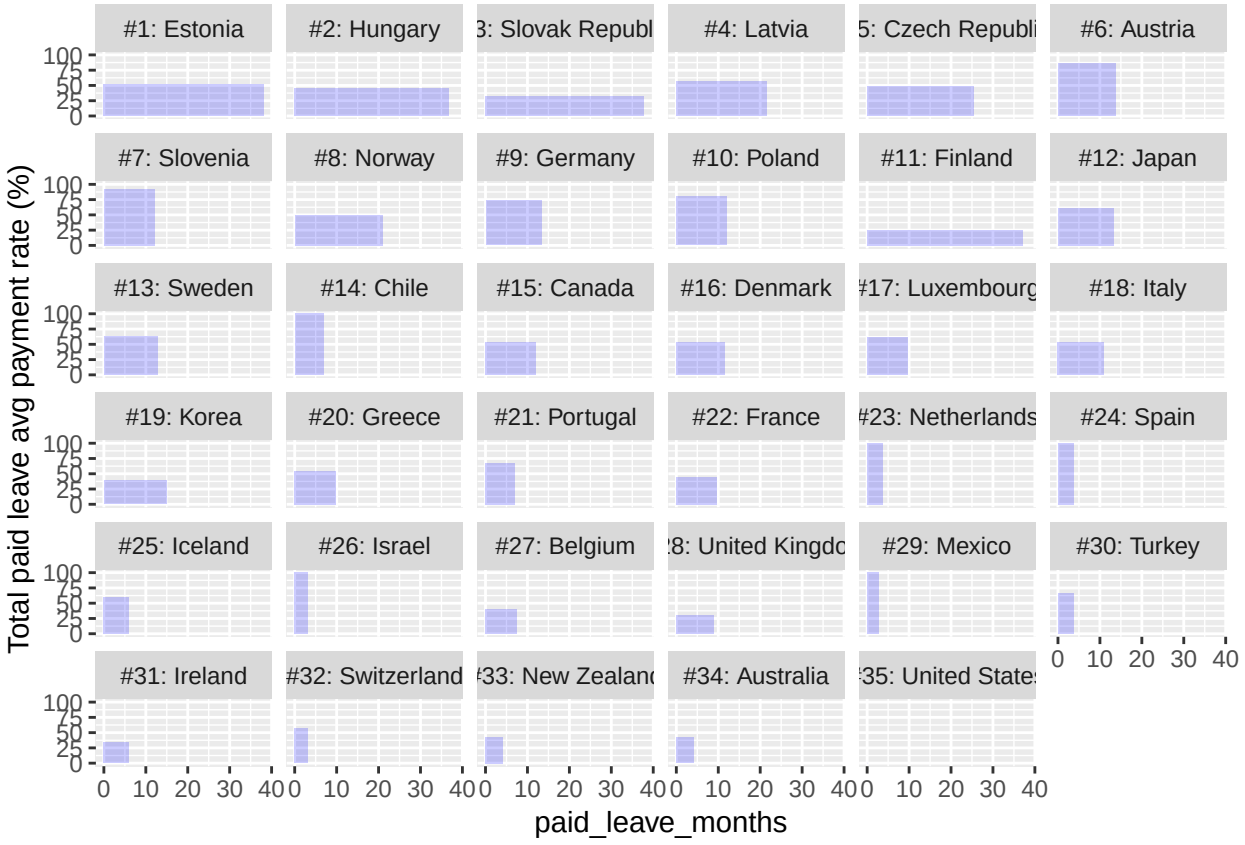
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`)
```



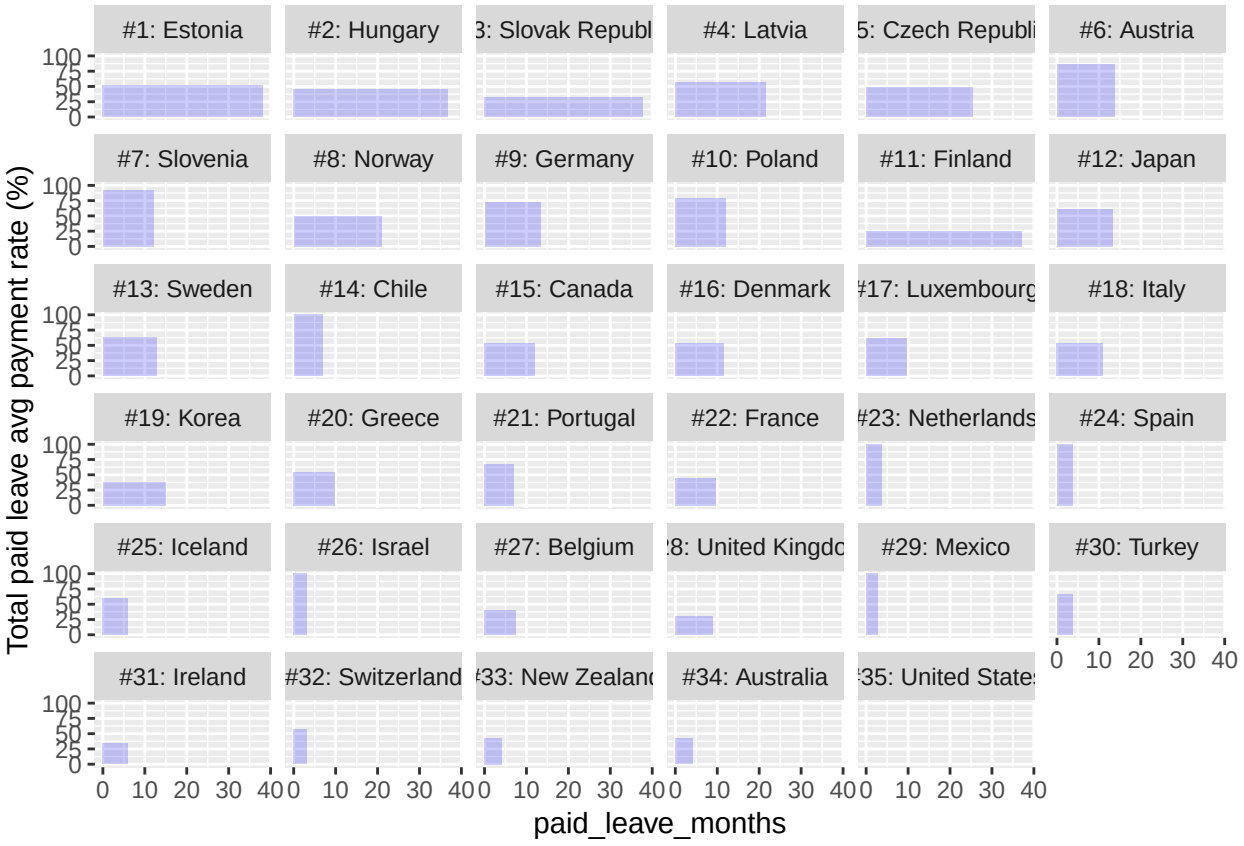
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .)
```



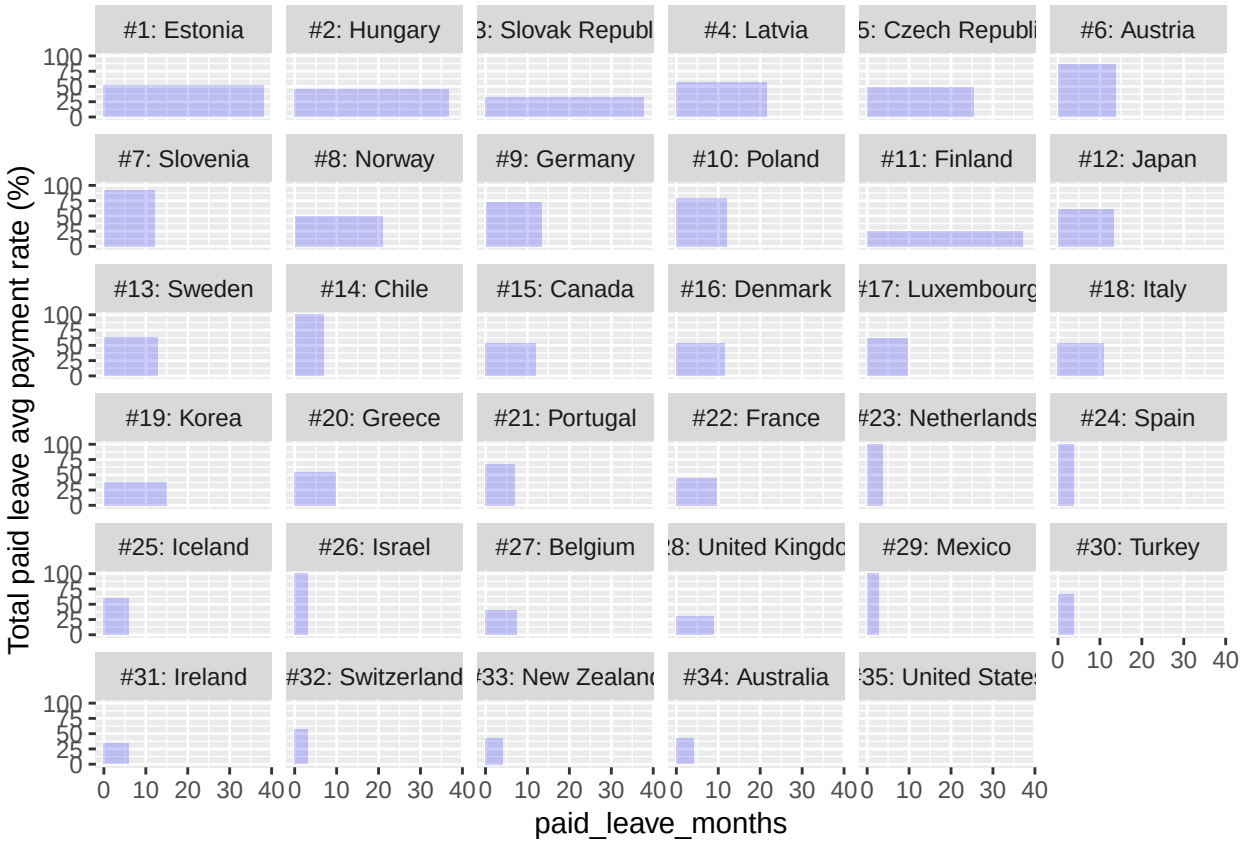
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2)
```



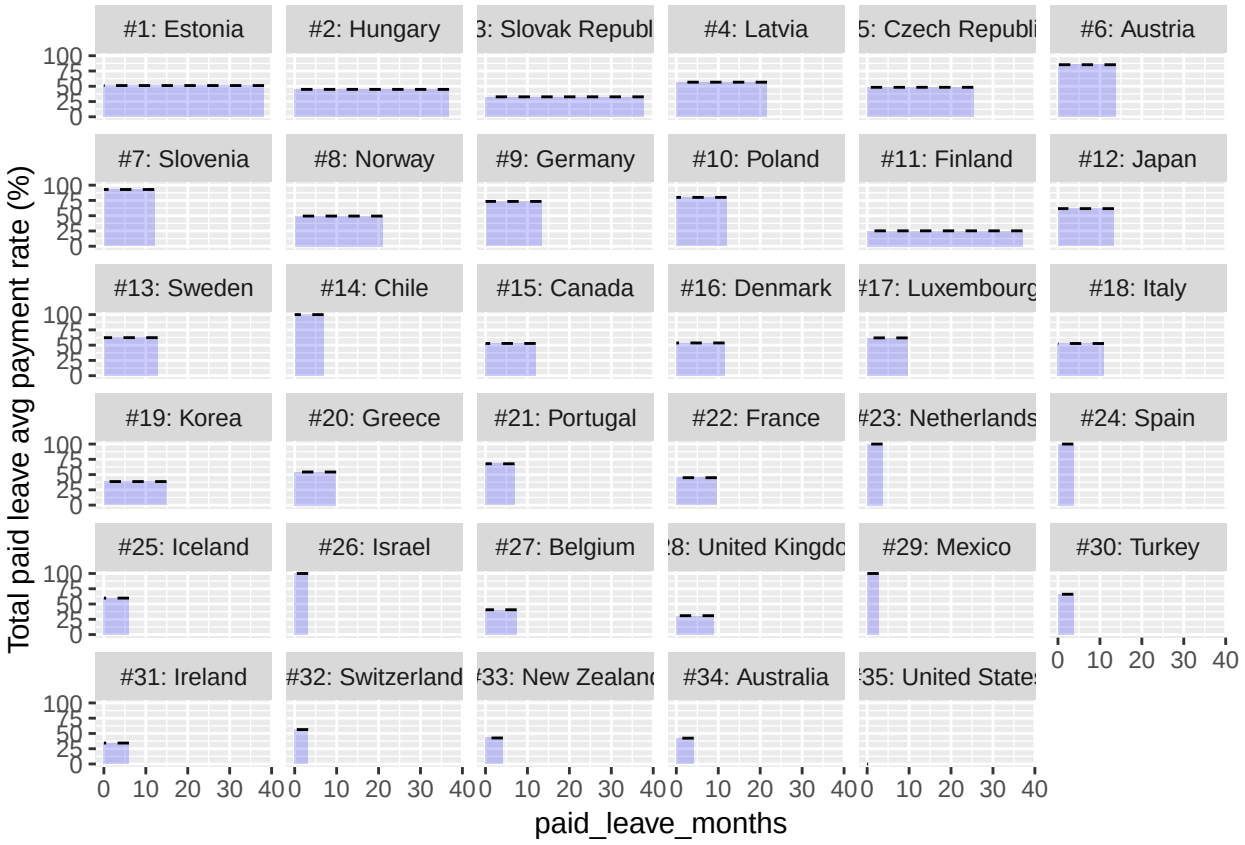
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0)
```



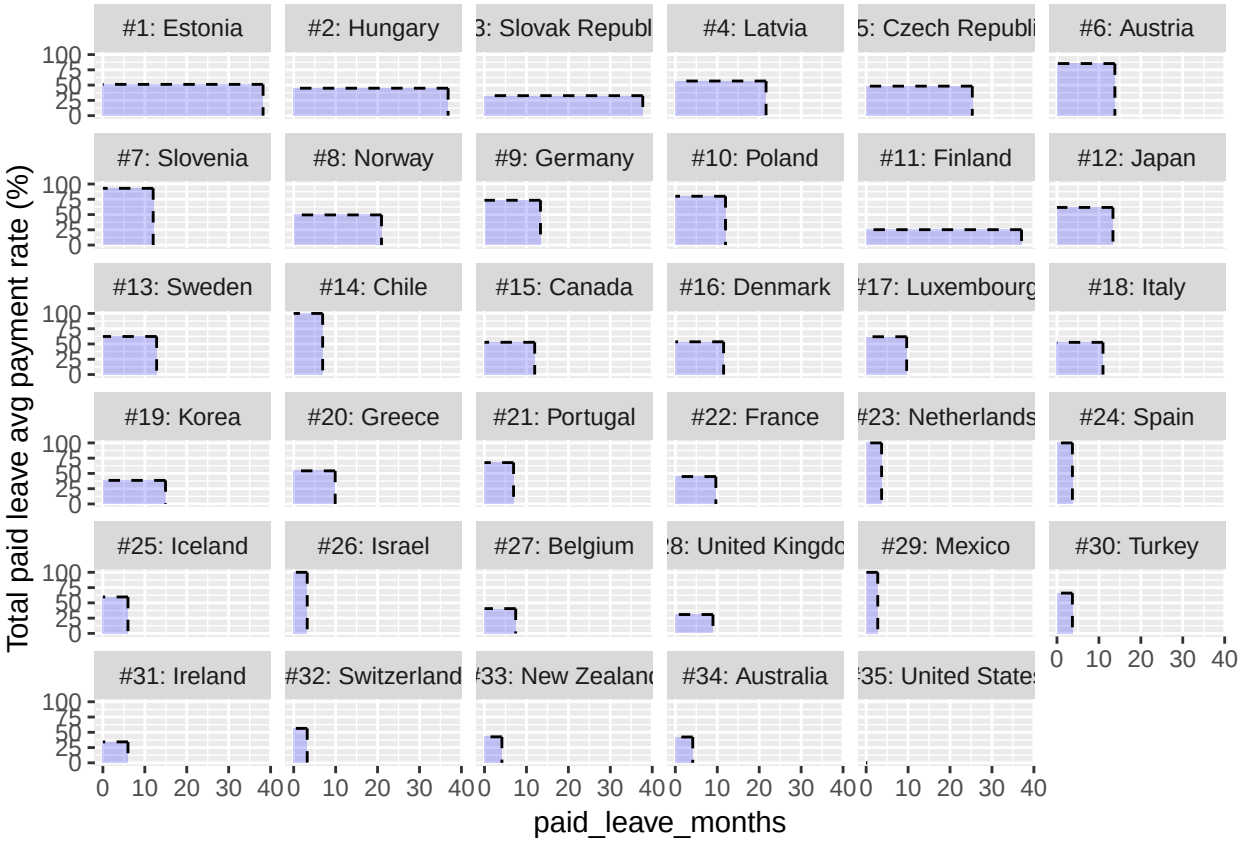
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0)
```



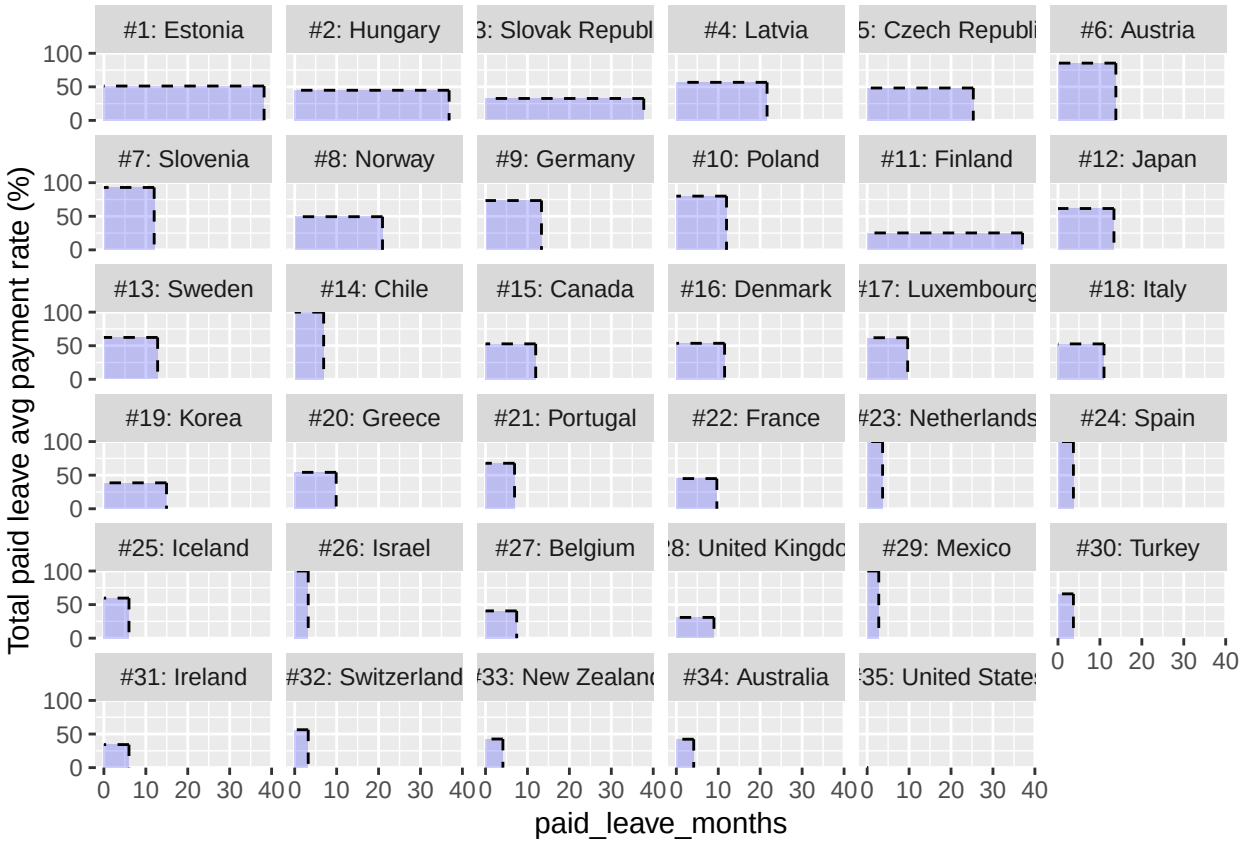
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0) +
  geom_segment(aes(yend = `Total paid leave avg payment rate (%)`), lty = "dashed")
```



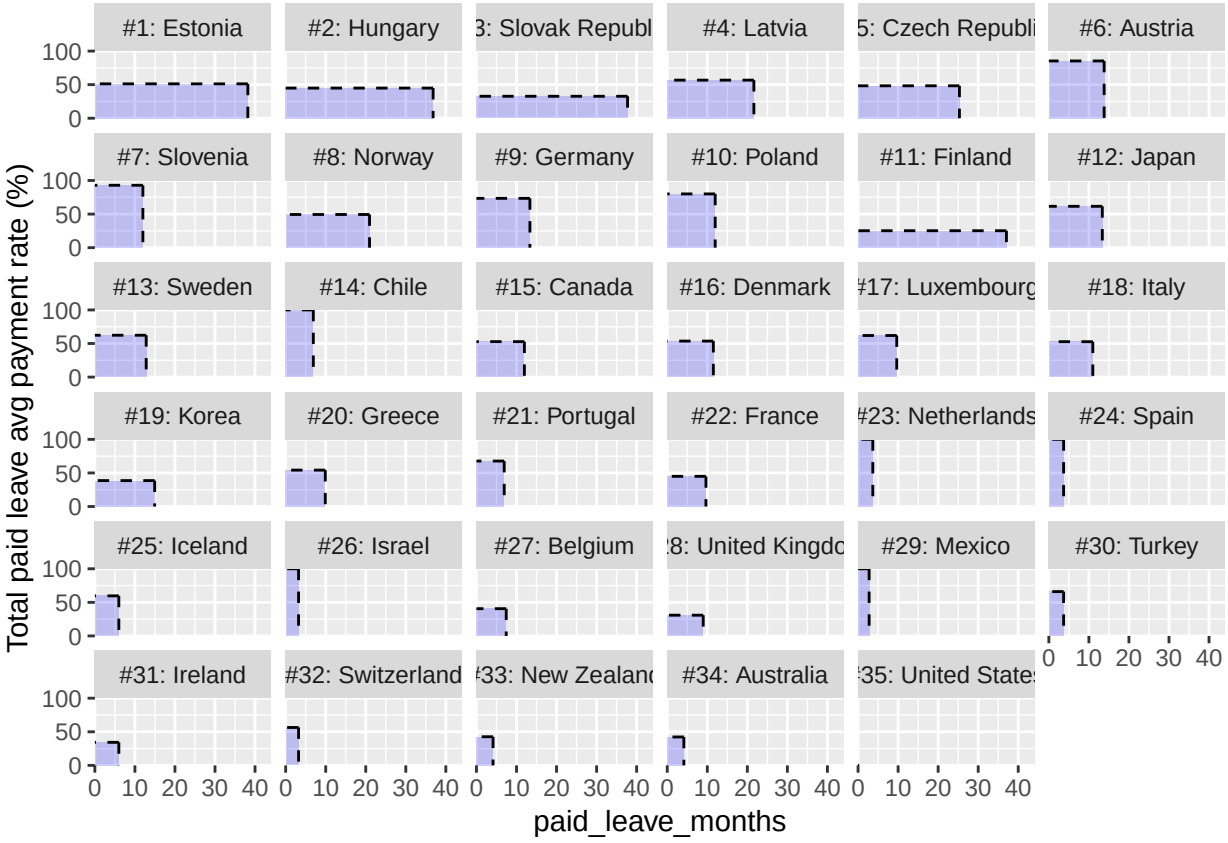
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0) +
  geom_segment(aes(yend = `Total paid leave avg payment rate (%)`), lty = "dashed") +
  geom_segment(aes(xend = paid_leave_months), lty = "dashed")
```



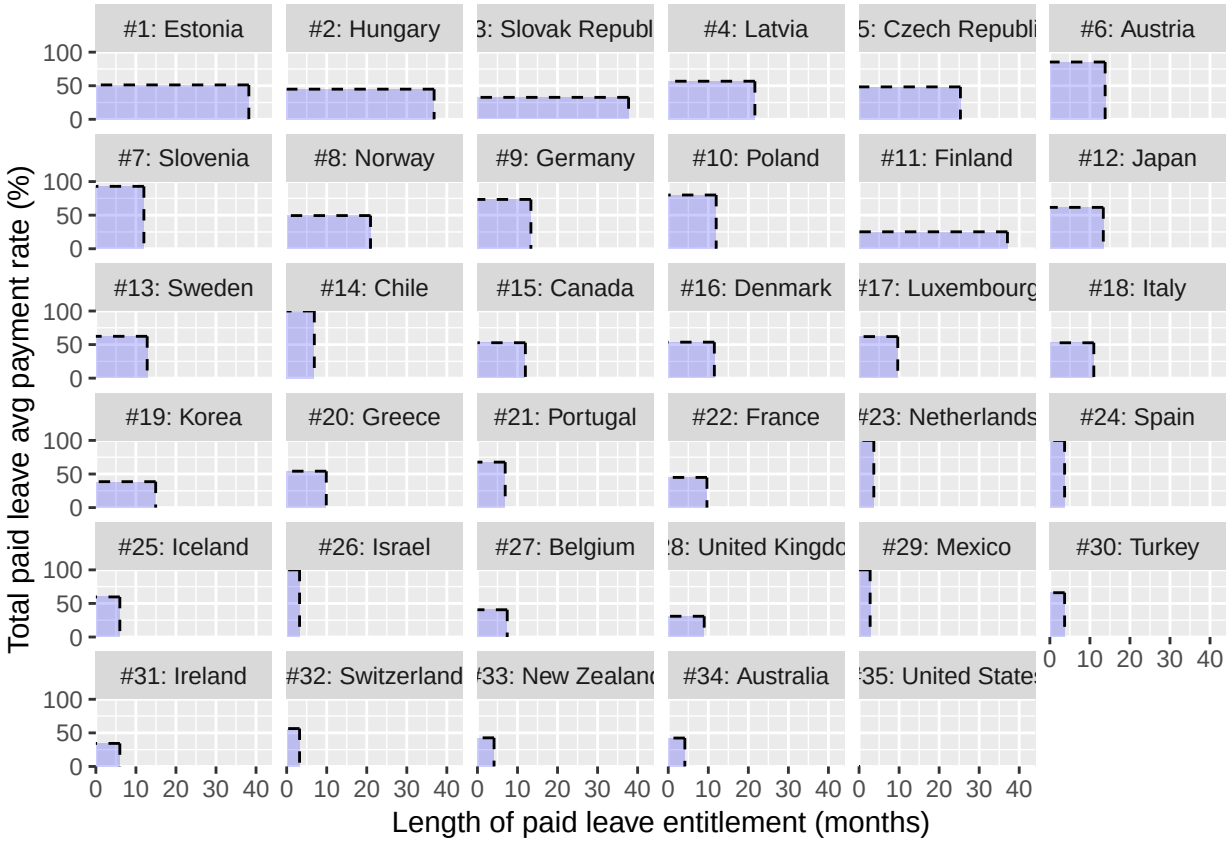
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0) +
  geom_segment(aes(yend = `Total paid leave avg payment rate (%)`), lty = "dashed") +
  geom_segment(aes(xend = paid_leave_months), lty = "dashed") +
  scale_y_continuous(limits = c(0, 100), expand = c(0, 0), breaks = c(0, 50, 100))
```



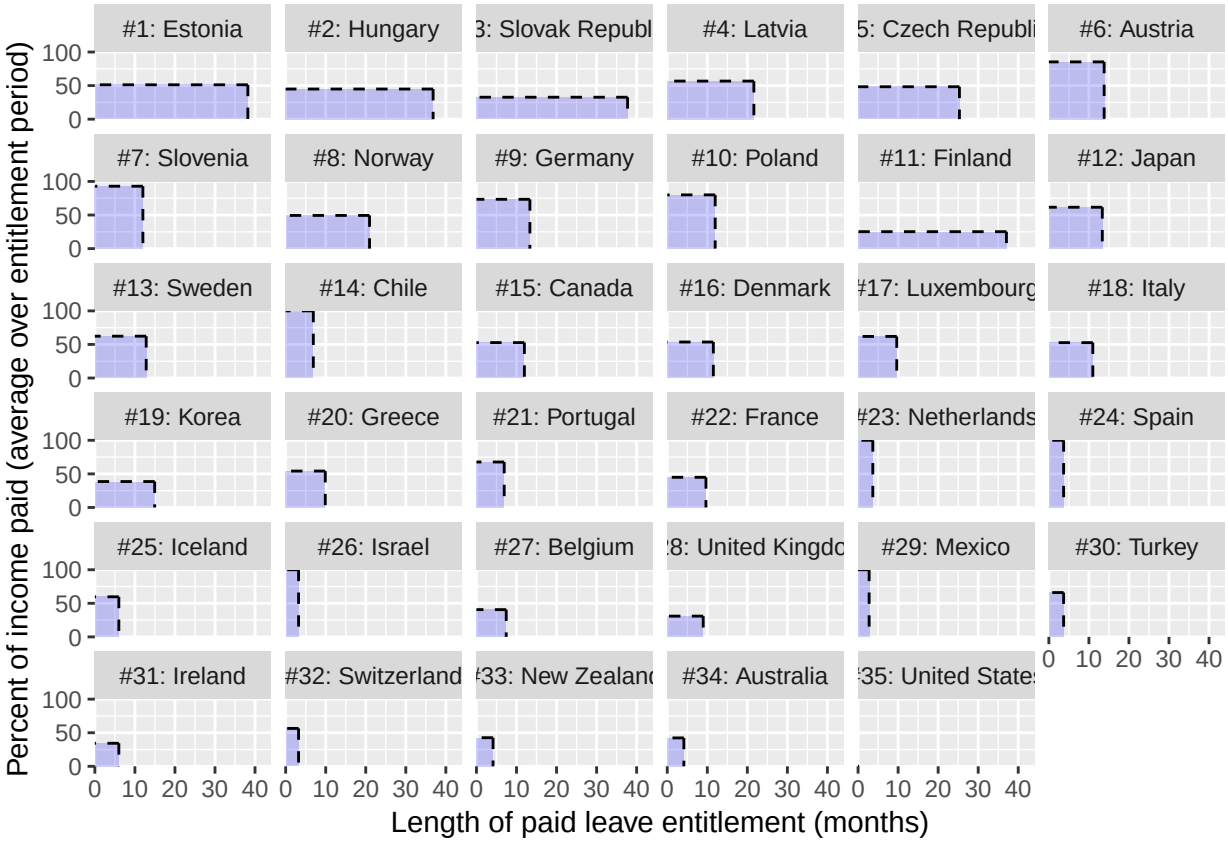
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0) +
  geom_segment(aes(yend = `Total paid leave avg payment rate (%)`), lty = "dashed") +
  geom_segment(aes(xend = paid_leave_months), lty = "dashed") +
  scale_y_continuous(limits = c(0, 100), expand = c(0, 0), breaks = c(0, 50, 100)) +
  scale_x_continuous(limits = c(0, 44), expand = c(0, 0))
```



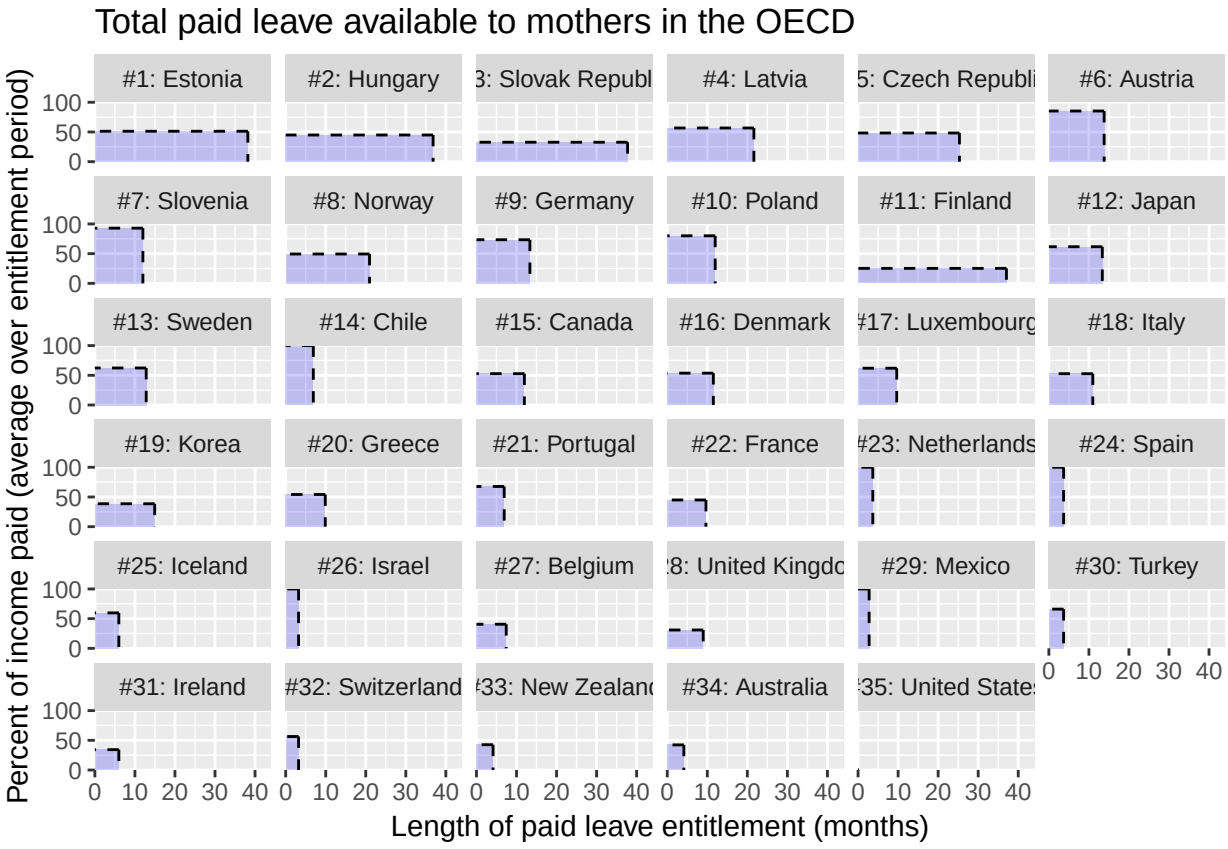
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0) +
  geom_segment(aes(yend = `Total paid leave avg payment rate (%)`), lty = "dashed") +
  geom_segment(aes(xend = paid_leave_months), lty = "dashed") +
  scale_y_continuous(limits = c(0, 100), expand = c(0, 0), breaks = c(0, 50, 100)) +
  scale_x_continuous(limits = c(0, 44), expand = c(0, 0)) +
  labs(x = "Length of paid leave entitlement (months)")
```



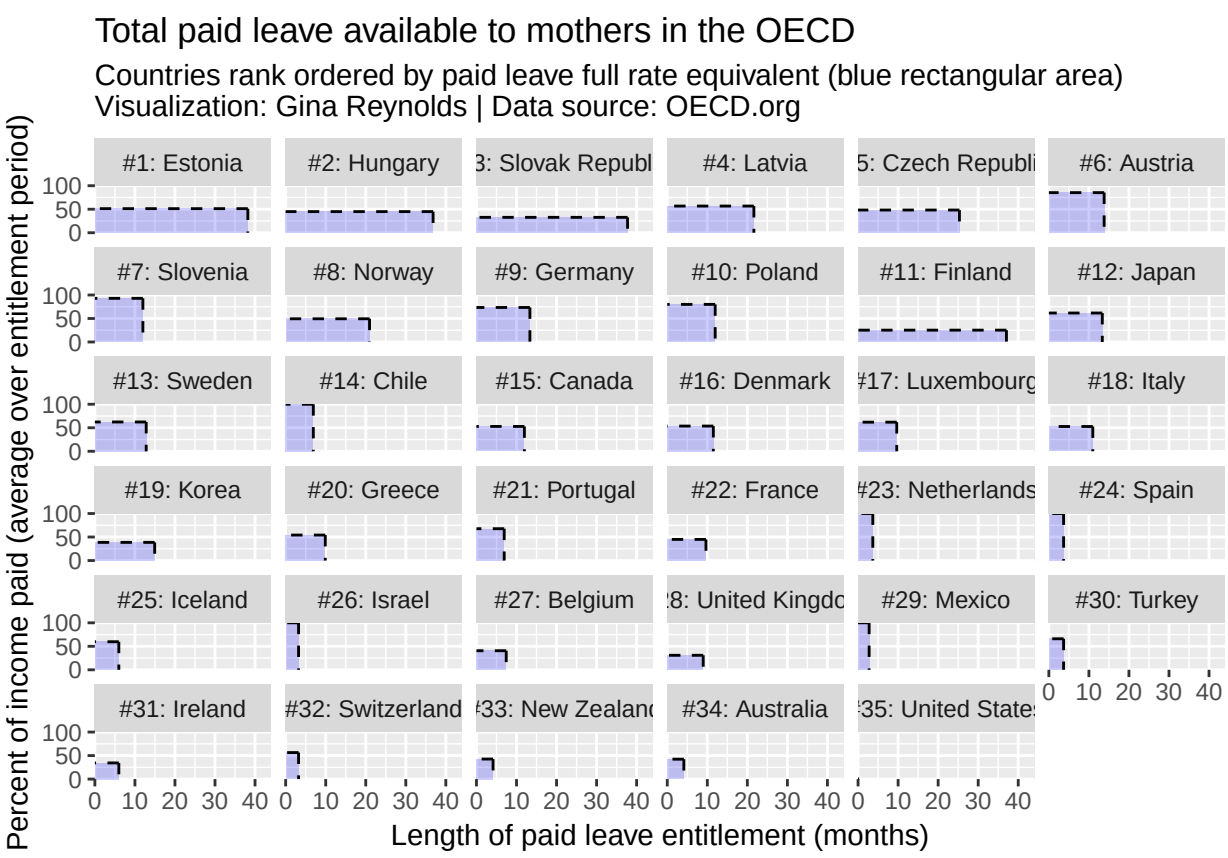
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0) +
  geom_segment(aes(yend = `Total paid leave avg payment rate (%)`), lty = "dashed") +
  geom_segment(aes(xend = paid_leave_months), lty = "dashed") +
  scale_y_continuous(limits = c(0, 100), expand = c(0, 0), breaks = c(0, 50, 100)) +
  scale_x_continuous(limits = c(0, 44), expand = c(0, 0)) +
  labs(x = "Length of paid leave entitlement (months)") +
  labs(y = "Percent of income paid (average over entitlement period)")
```



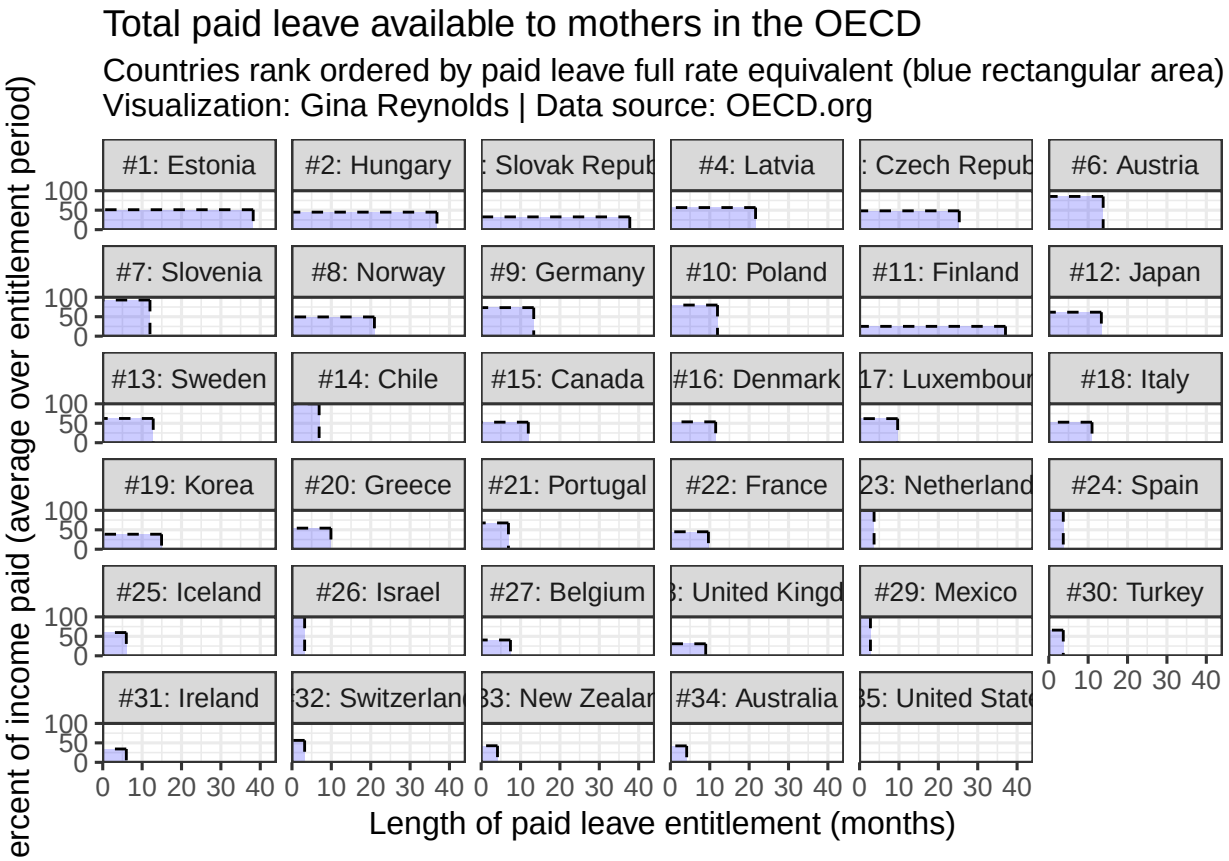
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0) +
  geom_segment(aes(yend = `Total paid leave avg payment rate (%)`), lty = "dashed") +
  geom_segment(aes(xend = paid_leave_months), lty = "dashed") +
  scale_y_continuous(limits = c(0, 100), expand = c(0, 0), breaks = c(0, 50, 100)) +
  scale_x_continuous(limits = c(0, 44), expand = c(0, 0)) +
  labs(x = "Length of paid leave entitlement (months)") +
  labs(y = "Percent of income paid (average over entitlement period)") +
  labs(title = "Total paid leave available to mothers in the OECD")
```



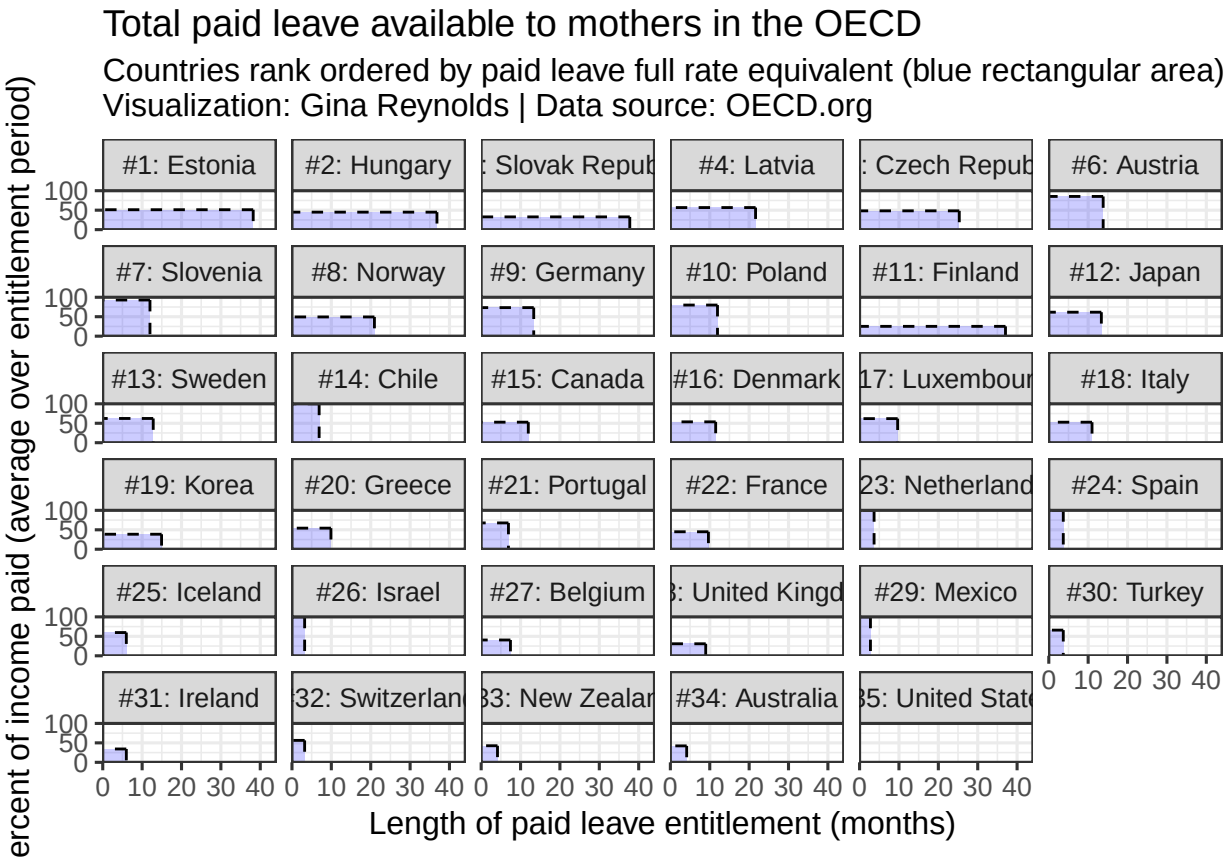
```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0) +
  geom_segment(aes(yend = `Total paid leave avg payment rate (%)`), lty = "dashed") +
  geom_segment(aes(xend = paid_leave_months), lty = "dashed") +
  scale_y_continuous(limits = c(0, 100), expand = c(0, 0), breaks = c(0, 50, 100)) +
  scale_x_continuous(limits = c(0, 44), expand = c(0, 0)) +
  labs(x = "Length of paid leave entitlement (months)") +
  labs(y = "Percent of income paid (average over entitlement period)") +
  labs(title = "Total paid leave available to mothers in the OECD") +
  labs(subtitle = "Countries rank ordered by paid leave full rate equivalent (blue rectangular area)\nVi
```



```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0) +
  geom_segment(aes(yend = `Total paid leave avg payment rate (%)`), lty = "dashed") +
  geom_segment(aes(xend = paid_leave_months), lty = "dashed") +
  scale_y_continuous(limits = c(0, 100), expand = c(0, 0), breaks = c(0, 50, 100)) +
  scale_x_continuous(limits = c(0, 44), expand = c(0, 0)) +
  labs(x = "Length of paid leave entitlement (months)") +
  labs(y = "Percent of income paid (average over entitlement period)") +
  labs(title = "Total paid leave available to mothers in the OECD") +
  labs(subtitle = "Countries rank ordered by paid leave full rate equivalent (blue rectangular area)\nV
  theme_bw(base_size = 12)
```



```
ggplot(df) +
  aes(x = paid_leave_months) +
  aes(y = `Total paid leave avg payment rate (%)`) +
  aes(xmin = 0) +
  aes(xmax = paid_leave_months) +
  aes(ymin = 0) +
  aes(ymax = `Total paid leave avg payment rate (%)`) +
  facet_wrap(fct_inorder(rank_name) ~ .) +
  geom_rect(fill = "blue", alpha = .2) +
  aes(yend = 0) +
  aes(xend = 0) +
  geom_segment(aes(yend = `Total paid leave avg payment rate (%)`), lty = "dashed") +
  geom_segment(aes(xend = paid_leave_months), lty = "dashed") +
  scale_y_continuous(limits = c(0, 100), expand = c(0, 0), breaks = c(0, 50, 100)) +
  scale_x_continuous(limits = c(0, 44), expand = c(0, 0)) +
  labs(x = "Length of paid leave entitlement (months)") +
  labs(y = "Percent of income paid (average over entitlement period)") +
  labs(title = "Total paid leave available to mothers in the OECD") +
  labs(subtitle = "Countries rank ordered by paid leave full rate equivalent (blue rectangular area)\nVi
  theme_bw(base_size = 12)
```



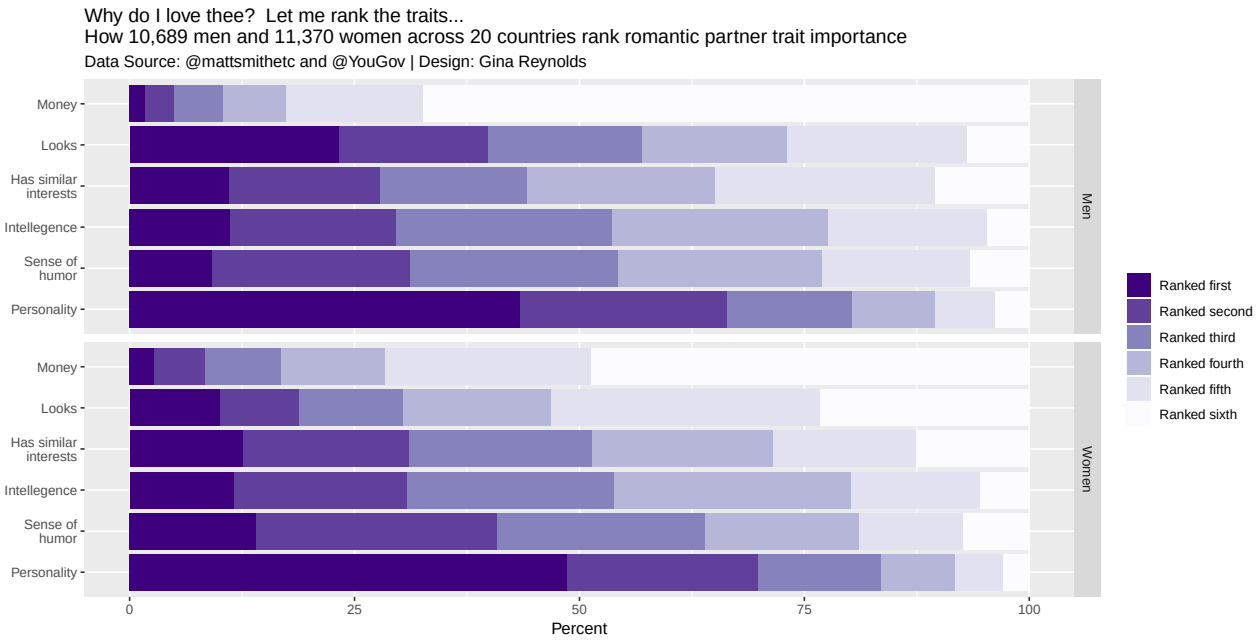
Chapter 6

Traits

A random sample from the data set:

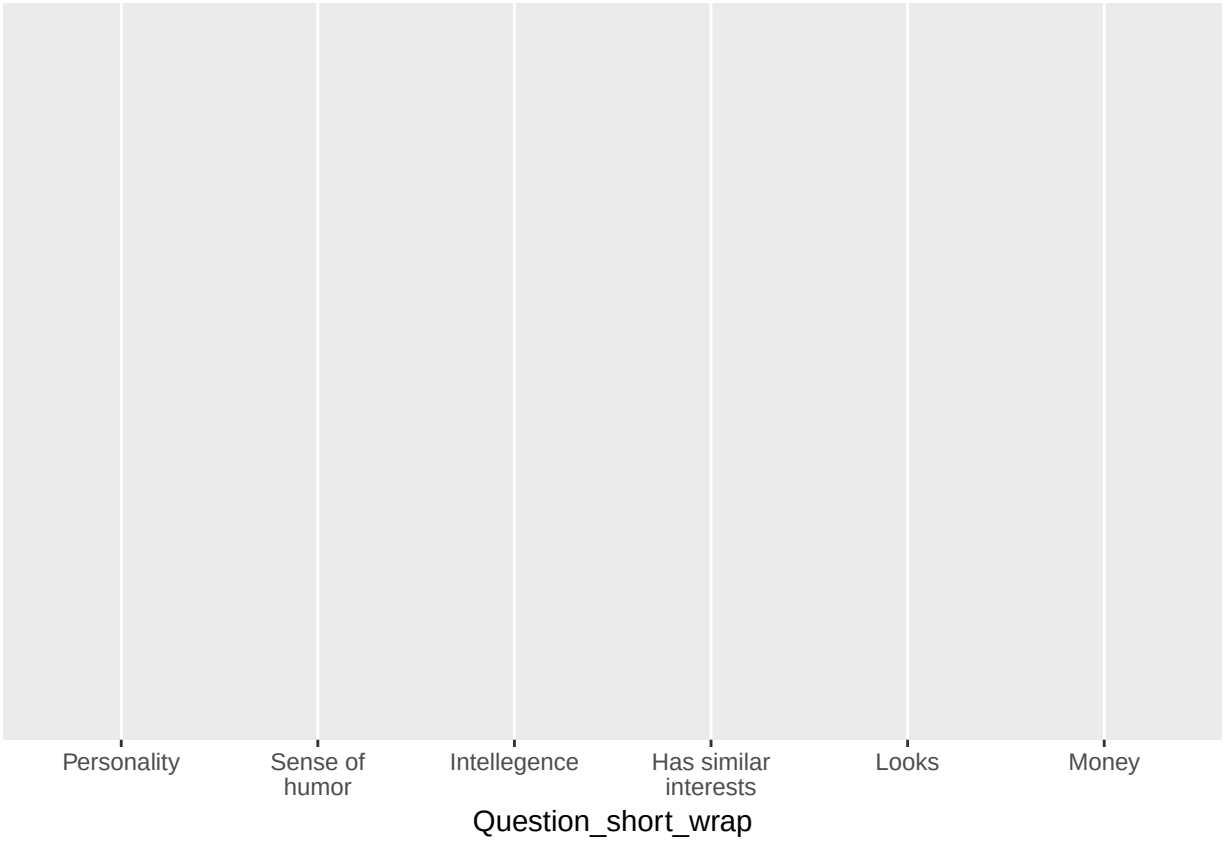
Gender	Question_short	Rank (text)	Rank (number)	n	Percent
Men	Money	Ranked fourth	4	754.62	7.071332
Women	Personality	Ranked second	2	2425.69	21.269083
Women	Has similar interests	Ranked first	1	1442.49	12.655053
Men	Personality	Ranked fifth	5	701.54	6.562005
Women	Looks	Ranked fifth	5	3403.09	29.893938

```
ggplot(data = world) +
  aes(x = Question_short_wrap) +
  aes(y = Percent) +
  aes(fill = `Rank (text)`) +
  facet_grid(Gender ~ .) +
  geom_col() +
  coord_flip() +
  scale_fill_manual(values = colorRampPalette(RColorBrewer::brewer.pal(9, "Purples"))(6)[1:6], guide = g
  labs(fill = "") +
  xlab("") +
  labs(title = "Why do I love thee?  Let me rank the traits... \nHow 10,689 men and 11,370 women across
  labs(subtitle = "Data Source: @mattsmithetc and @YouGov | Design: Gina Reynolds")
```

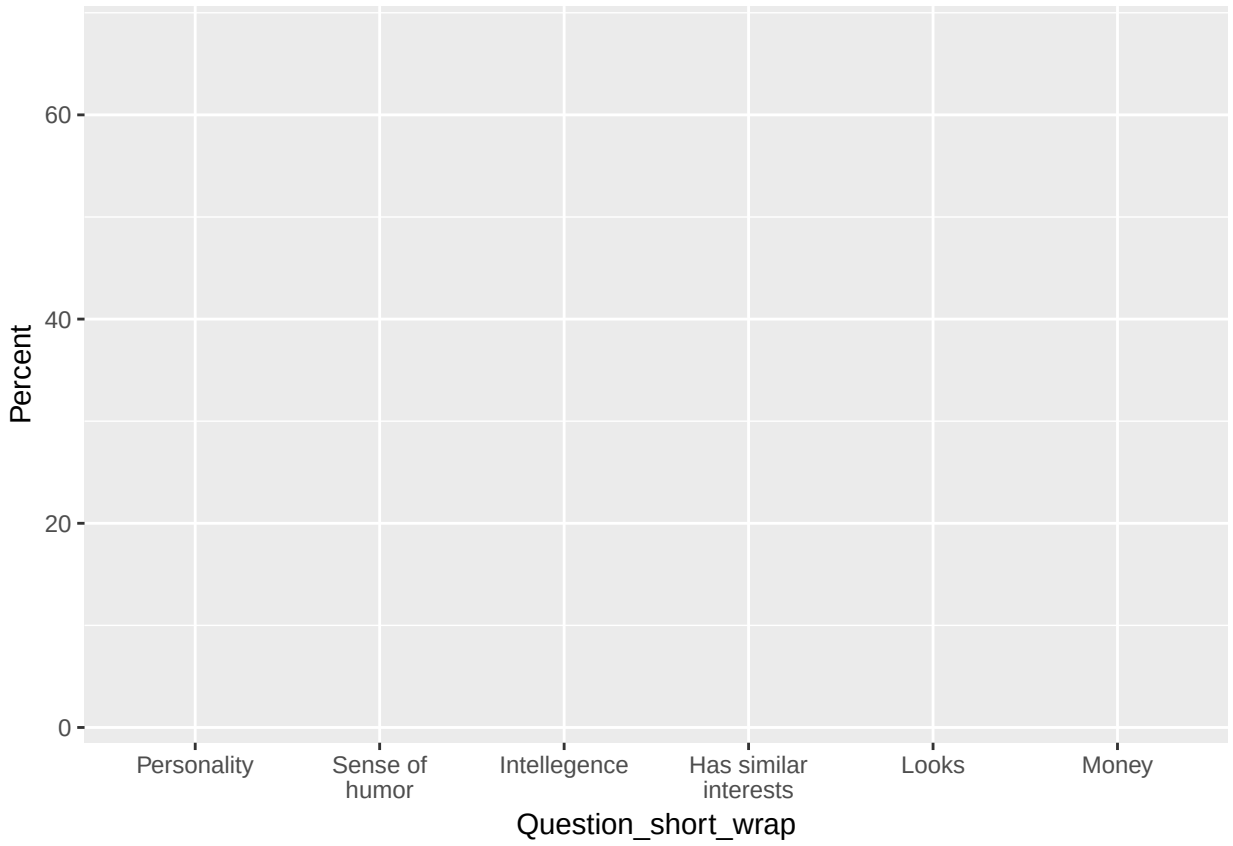

Why do I love thee? Let me rank the traits...
How 10,689 men and 11,370 women across 20 countries rank romantic partner trait importance
Data Source: @mattsmithetc and @YouGov | Design: Gina Reynolds
Money
Looks
Has similar interests
Intellegence
Sense of humor
Personality
Men
Women
0
25
50
75
100
Percent
Ranked first
Ranked second
Ranked third
Ranked fourth
Ranked fifth
Ranked sixth

```
ggplot(data = world)
```

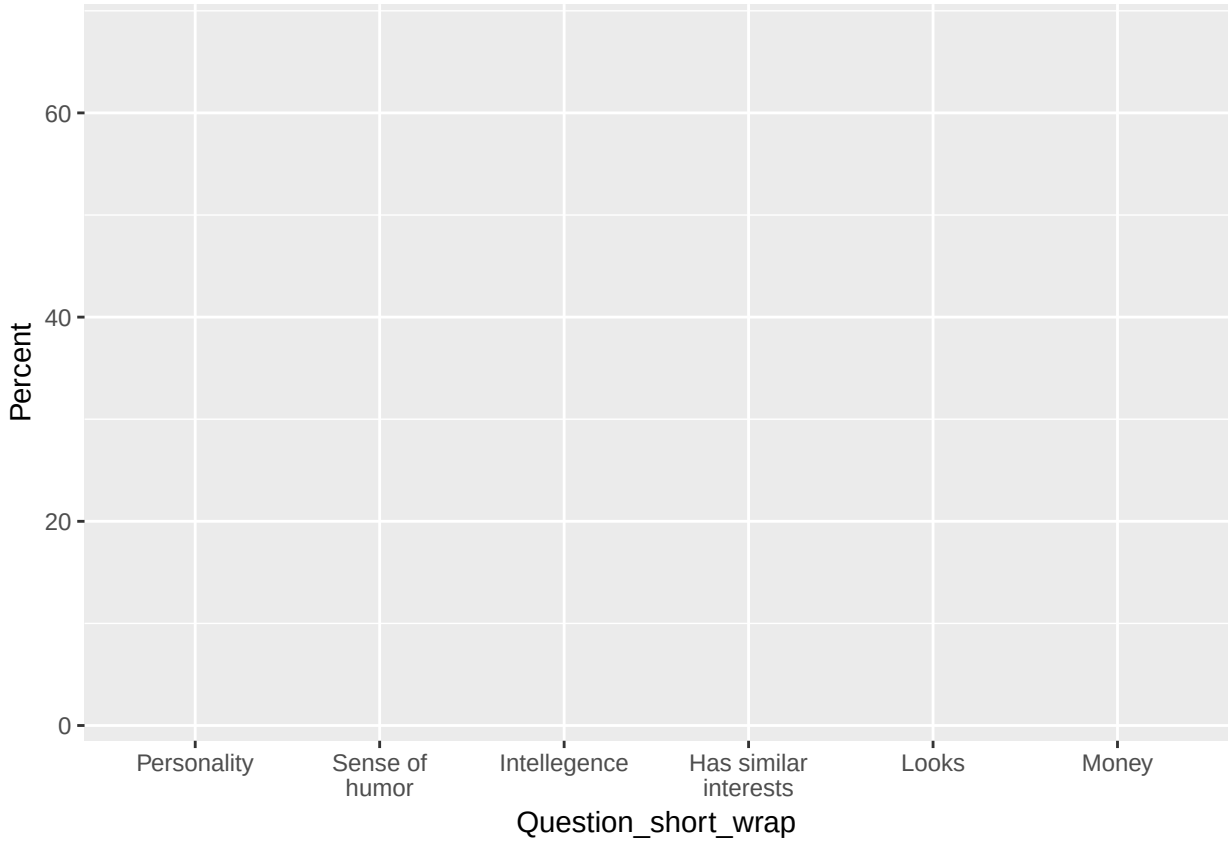
```
ggplot(data = world) +
  aes(x = Question_short_wrap)
```



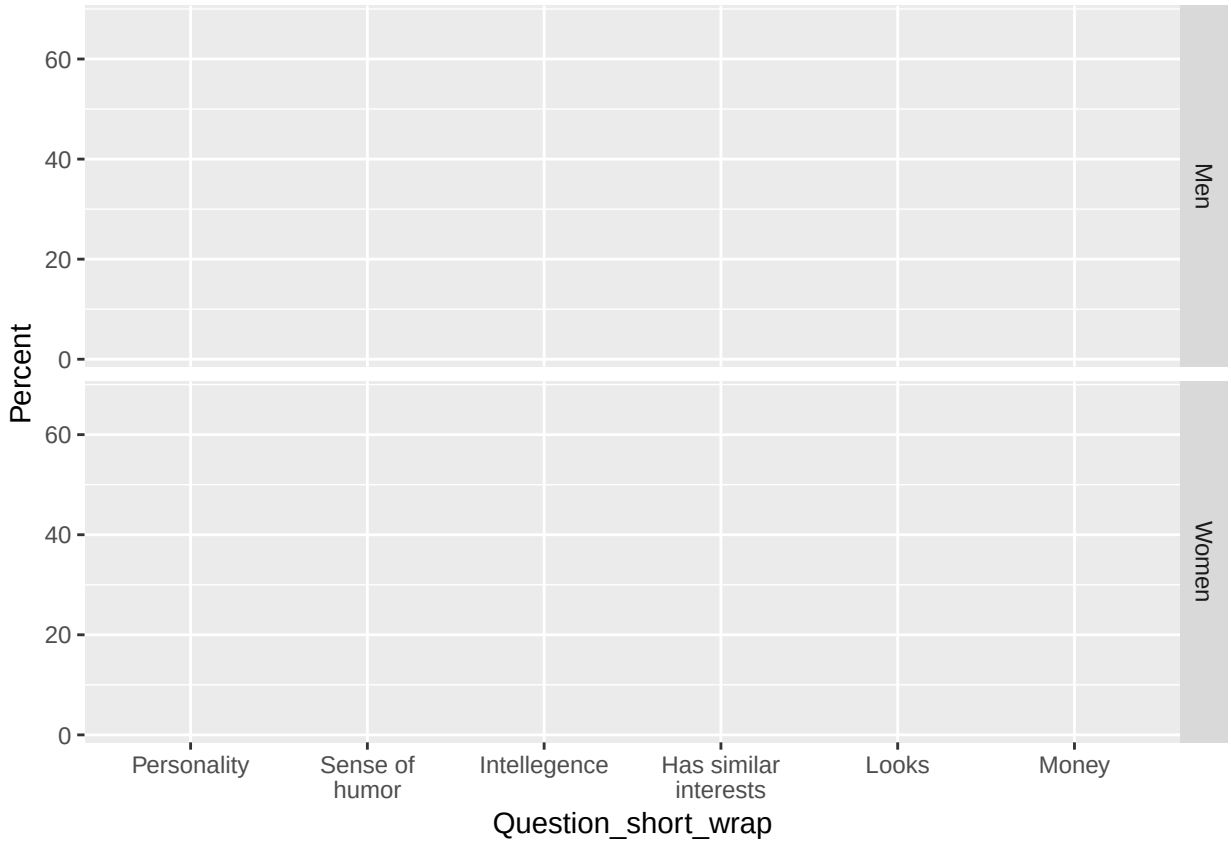
```
ggplot(data = world) +
  aes(x = Question_short_wrap) +
  aes(y = Percent)
```



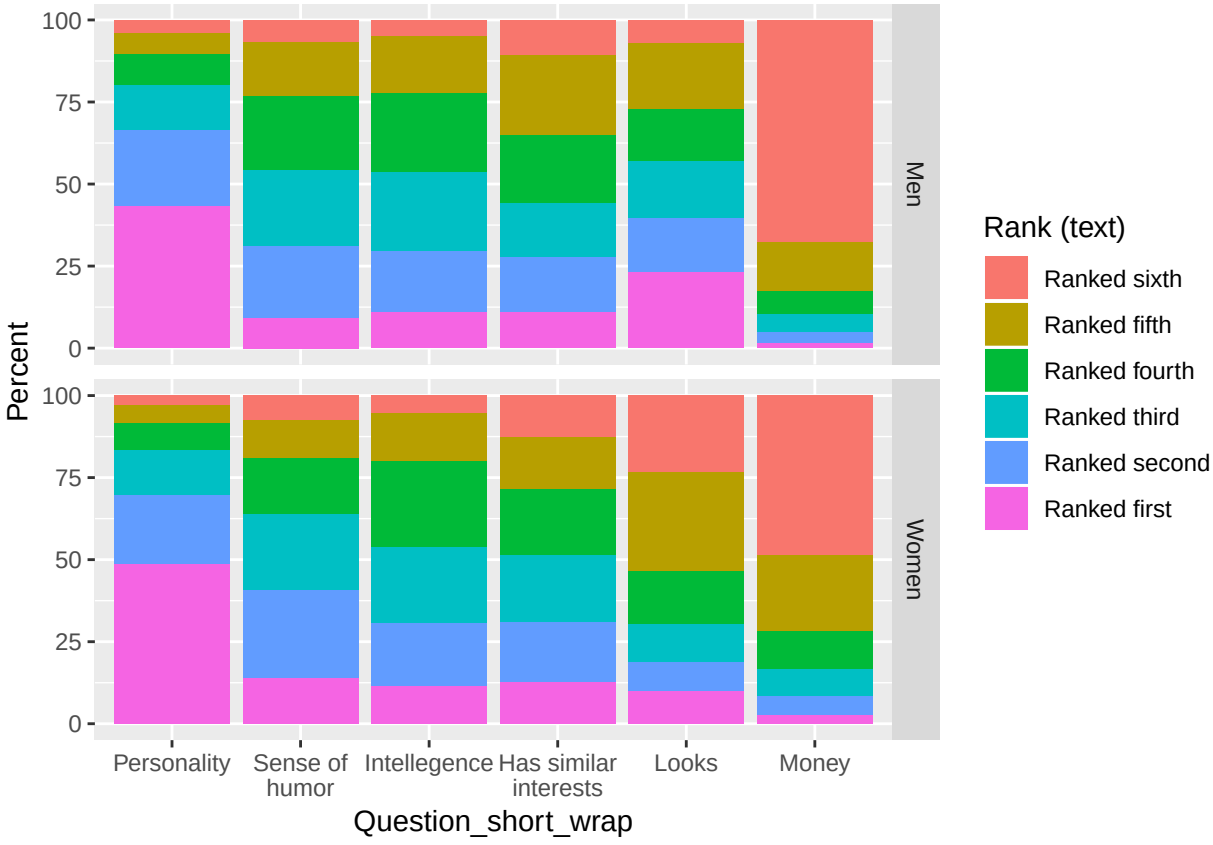
```
ggplot(data = world) +
  aes(x = Question_short_wrap) +
  aes(y = Percent) +
  aes(fill = `Rank (text)`)
```



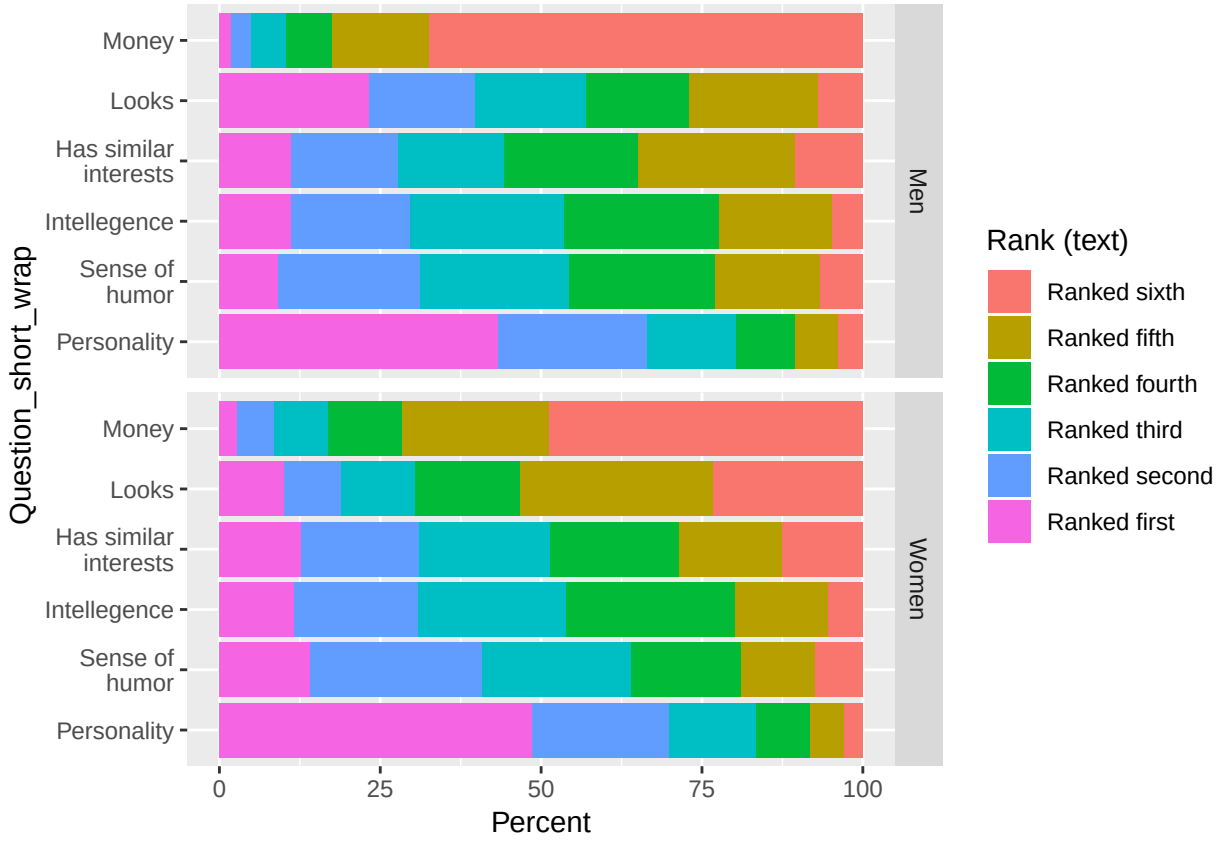
```
ggplot(data = world) +
  aes(x = Question_short_wrap) +
  aes(y = Percent) +
  aes(fill = `Rank (text)`) +
  facet_grid(Gender ~ .)
```



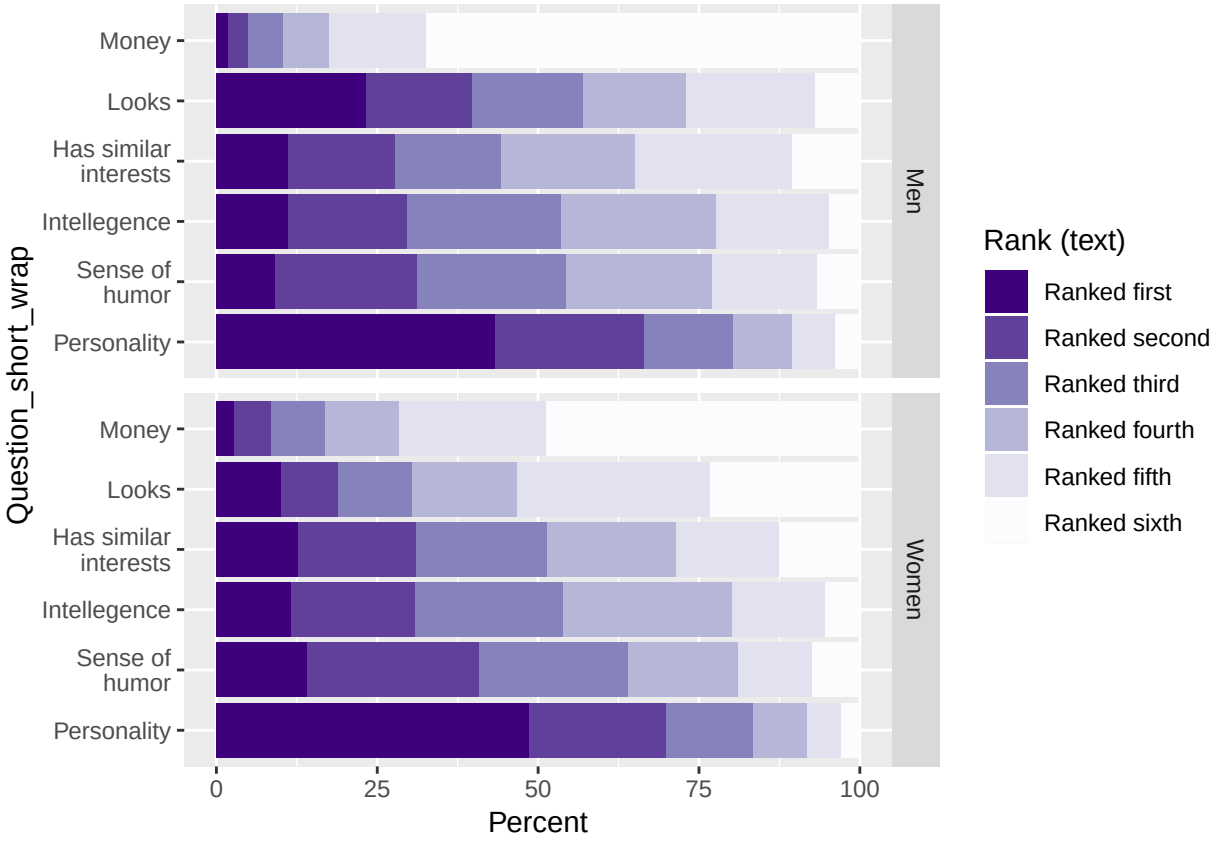
```
ggplot(data = world) +
  aes(x = Question_short_wrap) +
  aes(y = Percent) +
  aes(fill = `Rank (text)`) +
  facet_grid(Gender ~ .) +
  geom_col()
```



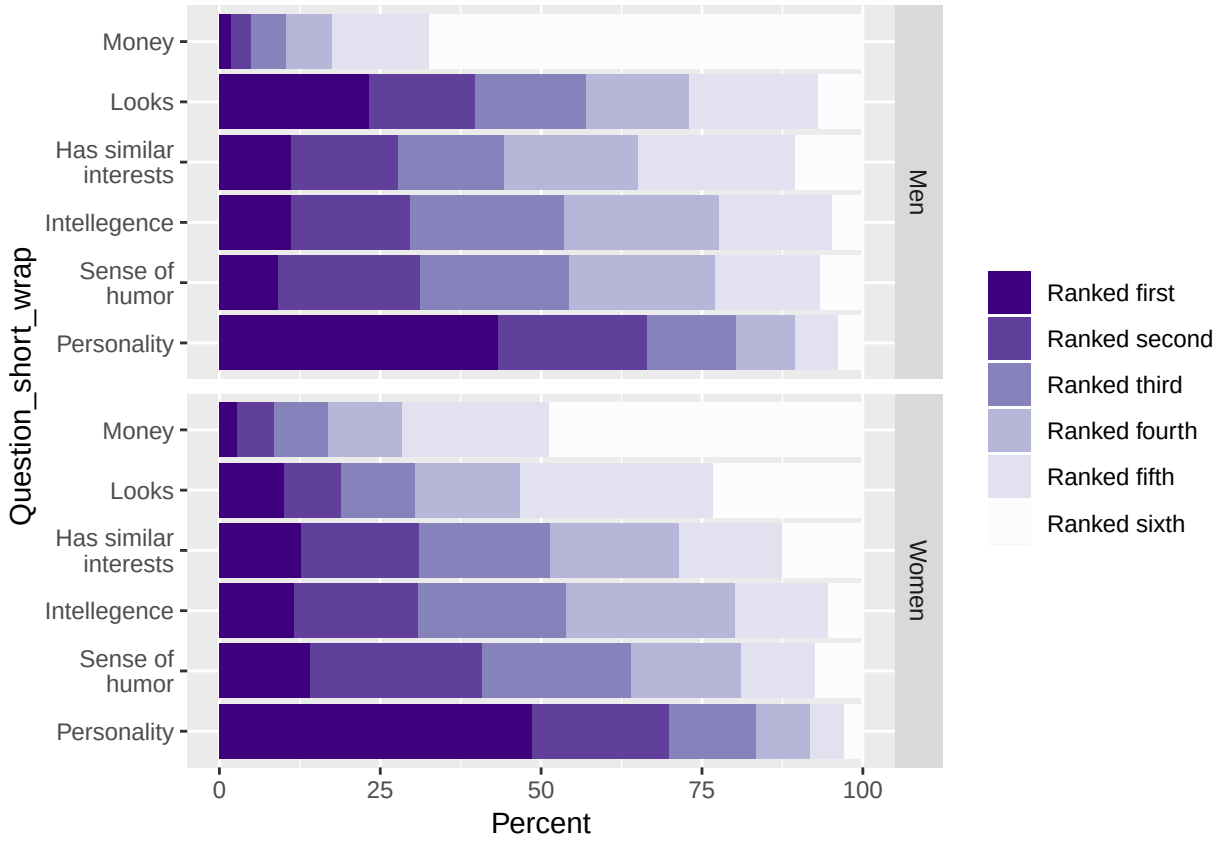
```
ggplot(data = world) +
  aes(x = Question_short_wrap) +
  aes(y = Percent) +
  aes(fill = `Rank (text)`) +
  facet_grid(Gender ~ .) +
  geom_col() +
  coord_flip()
```



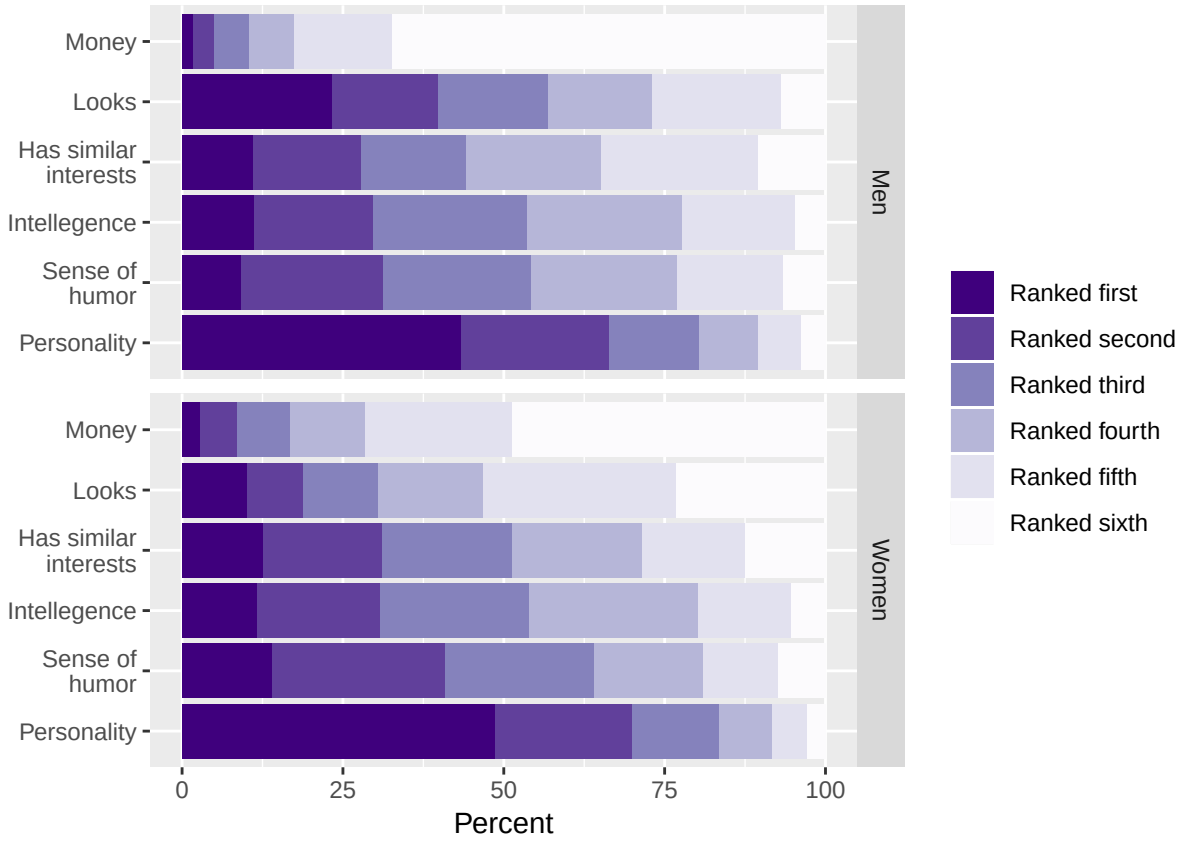
```
ggplot(data = world) +
  aes(x = Question_short_wrap) +
  aes(y = Percent) +
  aes(fill = `Rank (text)`) +
  facet_grid(Gender ~ .) +
  geom_col() +
  coord_flip() +
  scale_fill_manual(values = colorRampPalette(RColorBrewer::brewer.pal(9, "Purples"))(6)[1:6], guide = g
```



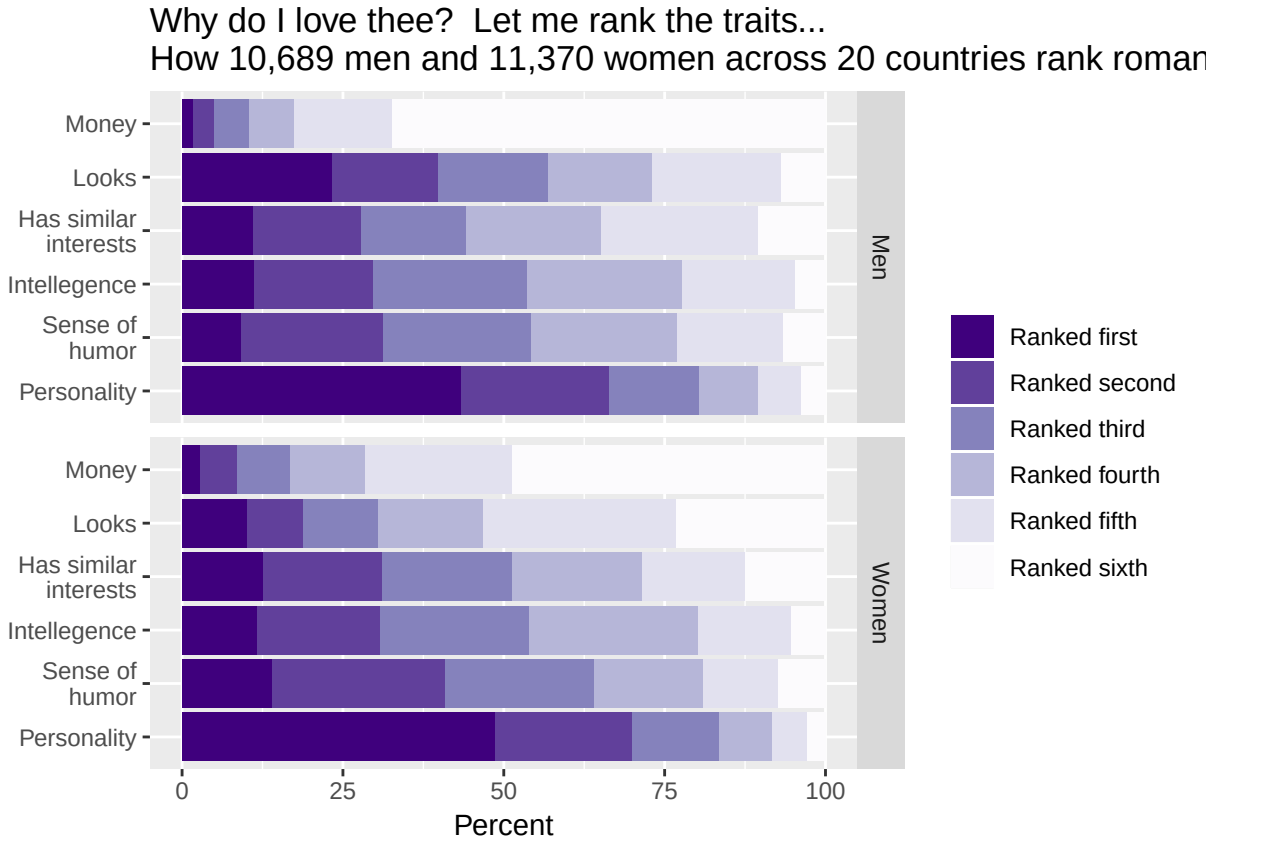
```
ggplot(data = world) +
  aes(x = Question_short_wrap) +
  aes(y = Percent) +
  aes(fill = `Rank (text)`) +
  facet_grid(Gender ~ .) +
  geom_col() +
  coord_flip() +
  scale_fill_manual(values = colorRampPalette(RColorBrewer::brewer.pal(9, "Purples"))(6)[1:6], guide = g
  labs(fill = "")
```



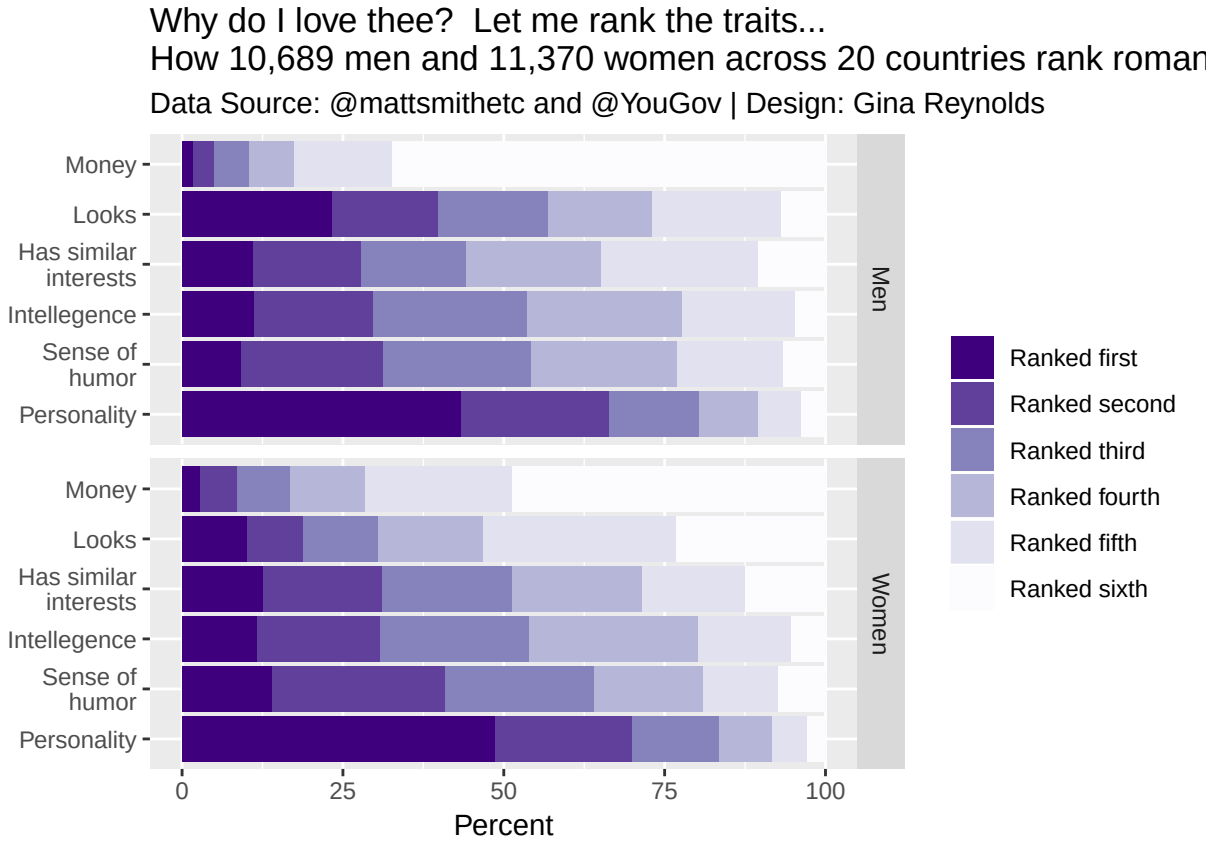
```
ggplot(data = world) +
  aes(x = Question_short_wrap) +
  aes(y = Percent) +
  aes(fill = `Rank (text)`) +
  facet_grid(Gender ~ .) +
  geom_col() +
  coord_flip() +
  scale_fill_manual(values = colorRampPalette(RColorBrewer::brewer.pal(9, "Purples"))(6)[1:6], guide = g
  labs(fill = "") +
  xlab("")
```



```
ggplot(data = world) +
  aes(x = Question_short_wrap) +
  aes(y = Percent) +
  aes(fill = `Rank (text)`) +
  facet_grid(Gender ~ .) +
  geom_col() +
  coord_flip() +
  scale_fill_manual(values = colorRampPalette(RColorBrewer::brewer.pal(9, "Purples"))(6)[1:6], guide = g
  labs(fill = "") +
  xlab("") +
  labs(title = "Why do I love thee?  Let me rank the traits... \nHow 10,689 men and 11,370 women across
```



```
ggplot(data = world) +
  aes(x = Question_short_wrap) +
  aes(y = Percent) +
  aes(fill = `Rank (text)`) +
  facet_grid(Gender ~ .) +
  geom_col() +
  coord_flip() +
  scale_fill_manual(values = colorRampPalette(RColorBrewer::brewer.pal(9, "Purples"))(6)[1:6], guide = g
  labs(fill = "") +
  xlab("") +
  labs(title = "Why do I love thee?  Let me rank the traits... \nHow 10,689 men and 11,370 women across
  labs(subtitle = "Data Source: @mattsmithetc and @YouGov | Design: Gina Reynolds")
```



Chapter 7

Salarys of Trump and Obama White House Employees

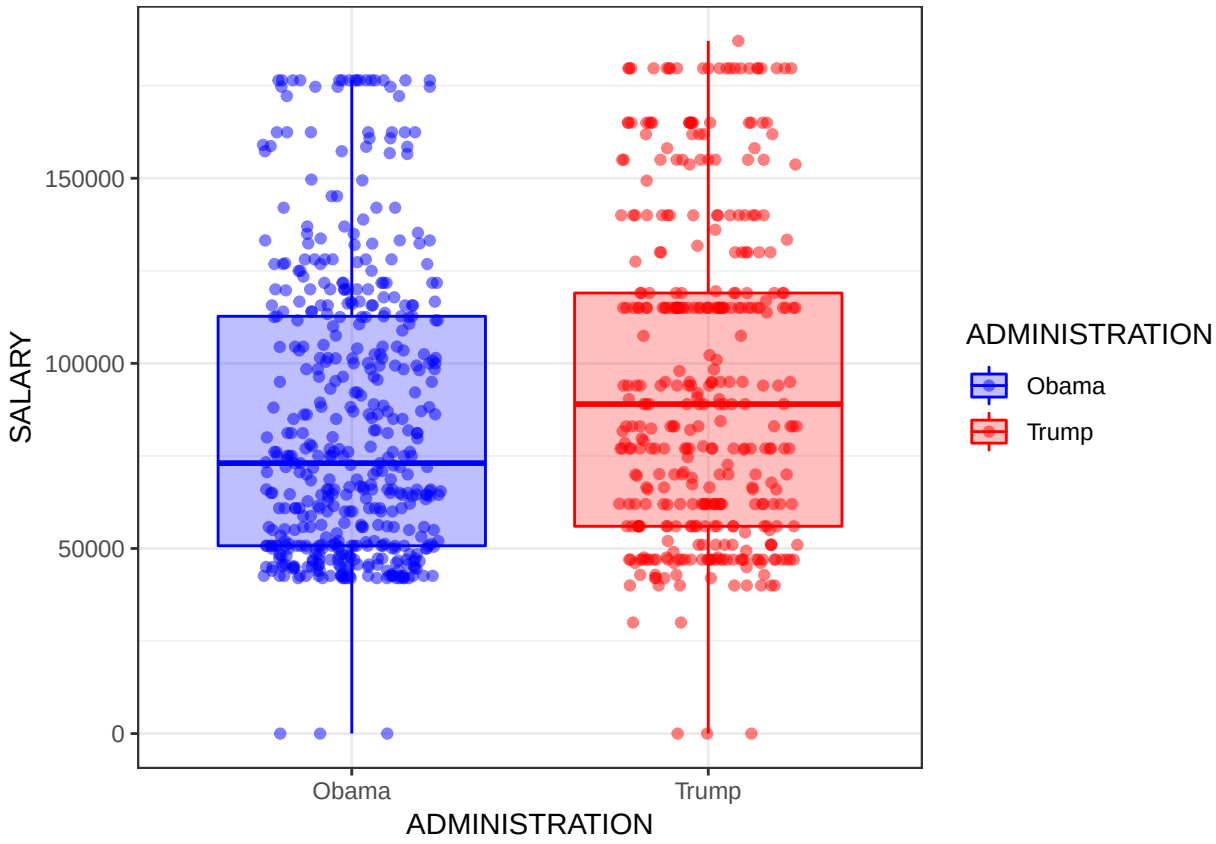
The data set, originally reported on in an NPR article, shows the difference in the distribution of salaries for the Obama and early Trump White House.

First I plot a histogram of each administration. Then I also contrast boxplots for each administration; the data points are overlayed, jittered to the widths of the boxplots. Plotly is used to make the graph interactive; mousing over will allow you to see who the point represents, their job description and exactly how much they are paid.

A random sample from the data set:

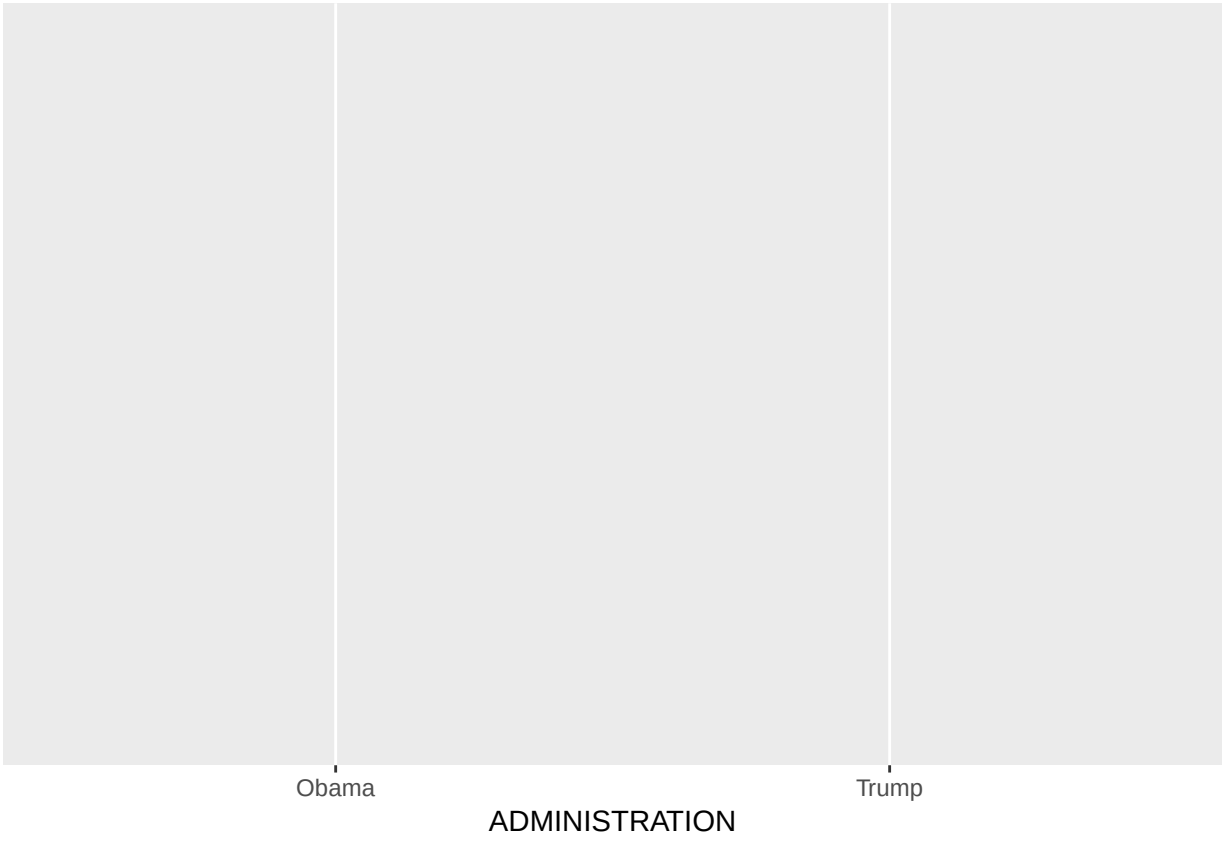
ADMINISTRATION	NAME	STATUS	SALARY	PAY BASIS	POSITION TITLE
Obama	Vahlsing, Candace M.	Employee	75000	Per Annum	SENIOR POLICY ADVISOR
Obama	Krupin, Stephen A.	Employee	85000	Per Annum	SENIOR PRESIDENTIAL SPEEC
Obama	Gonzalez, Ximena	Employee	65000	Per Annum	DEPUTY CHIEF OF STAFF FOI
Obama	Earnest, Joshua R.	Employee	176461	Per Annum	ASSISTANT TO THE PRESIDEN
Trump	Leighton, Rosalyn A.	Employee	89000	Per Annum	REGIONAL DIRECTOR

```
ggplot(both_data) +
  aes(x = ADMINISTRATION) +
  aes(y = SALARY) +
  geom_jitter(alpha = .5, height = 0, width = .25) +
  aes(col = ADMINISTRATION) +
  geom_boxplot(alpha = .25) +
  aes(fill = ADMINISTRATION) +
  scale_colour_manual(values = c("blue", "red")) +
  scale_fill_manual(values = c("blue", "red")) +
  theme_bw()
```

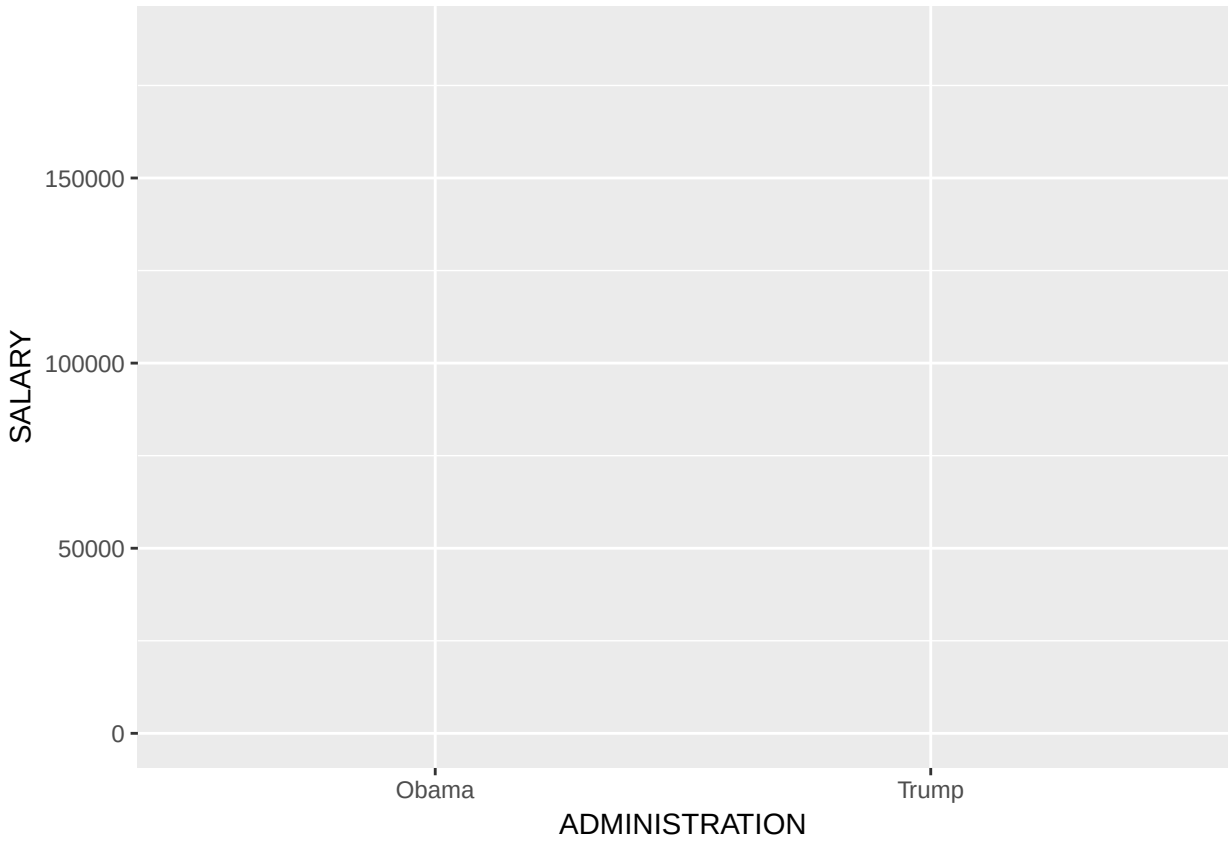

SALARY
150000
100000
50000
0
Obama
Trump
ADMINISTRATION
ADMINISTRATION
Obama
Trump

```
ggplot(both_data)
```

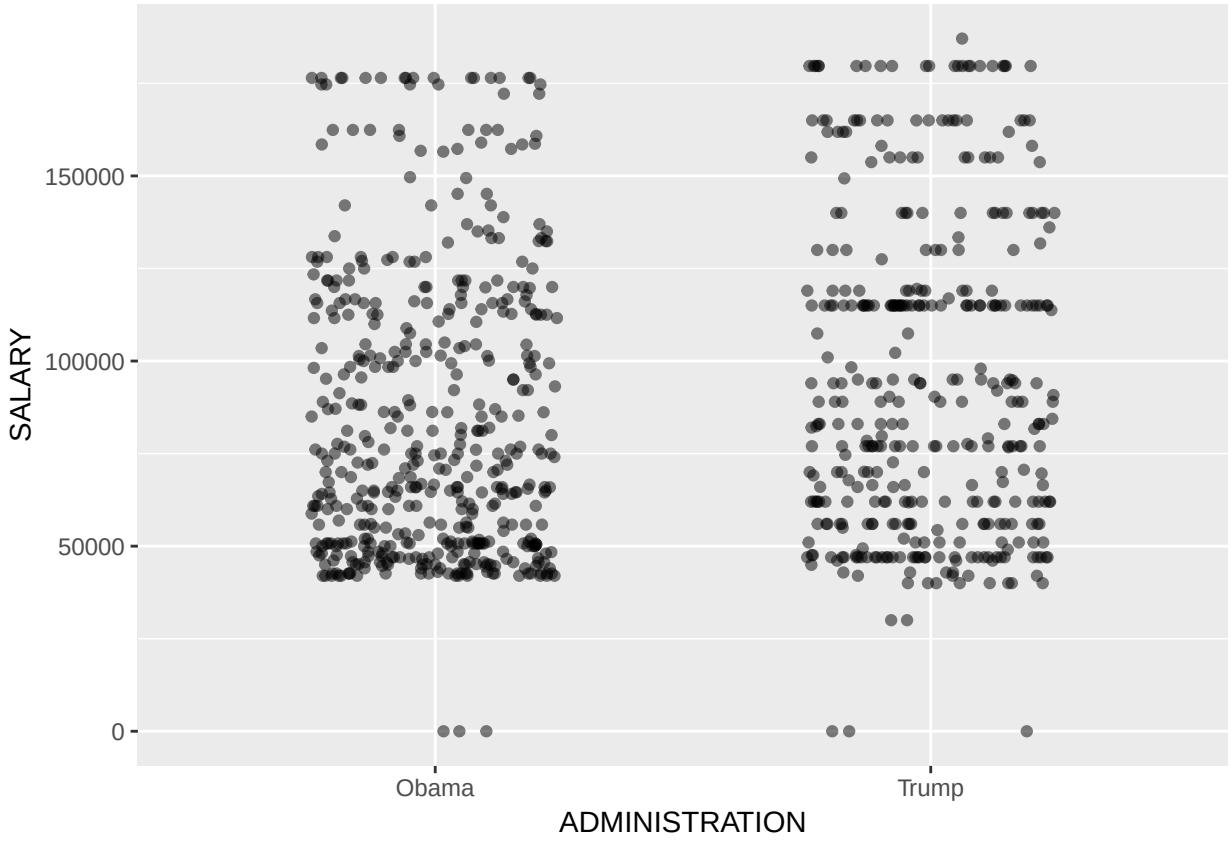
```
ggplot(both_data) +
  aes(x = ADMINISTRATION)
```



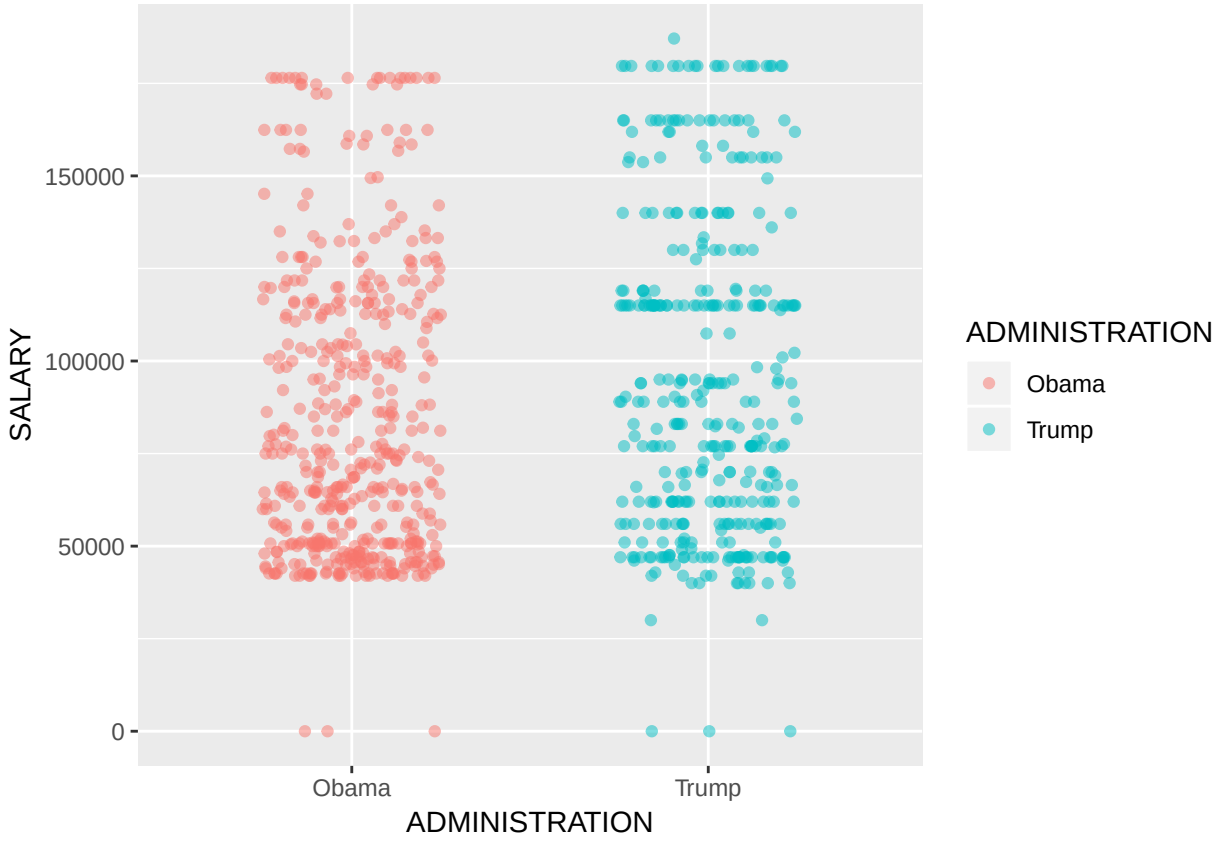
```
ggplot(both_data) +
  aes(x = ADMINISTRATION) +
  aes(y = SALARY)
```



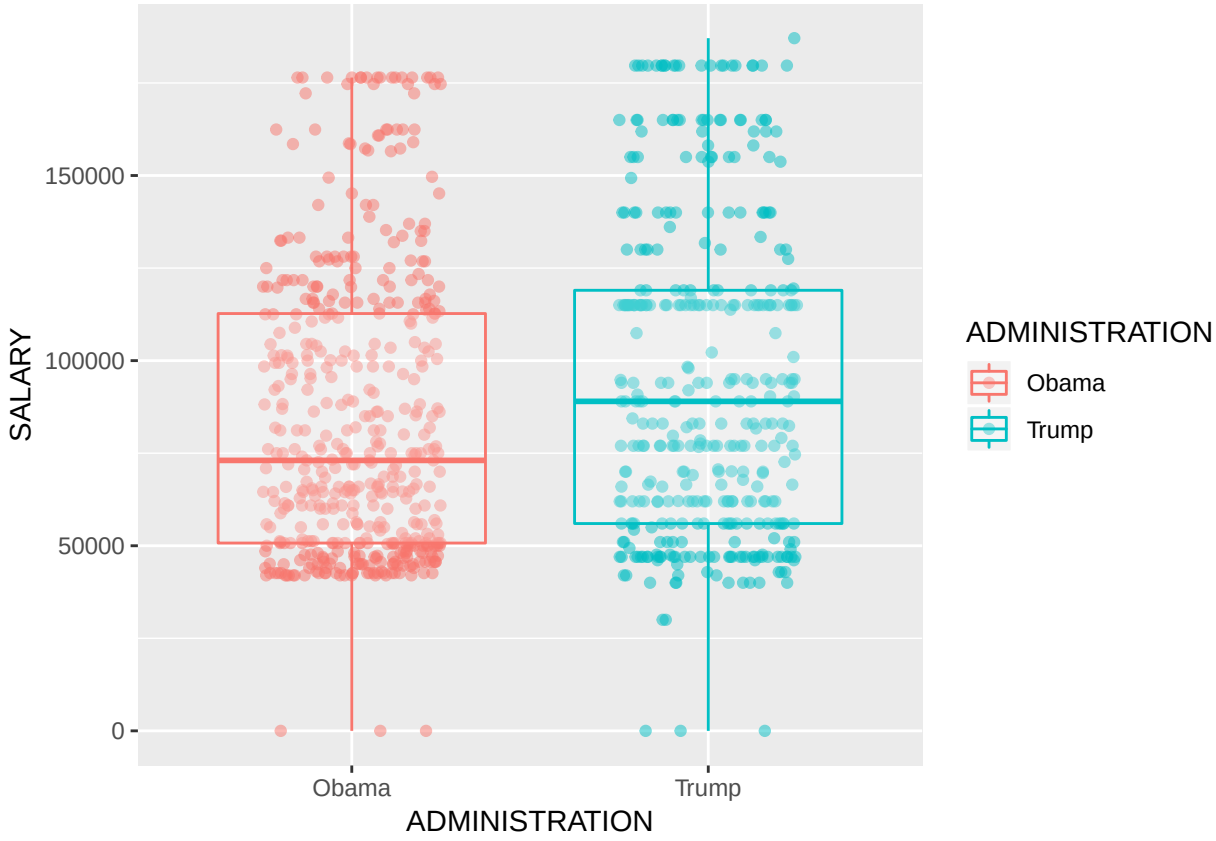
```
ggplot(both_data) +
  aes(x = ADMINISTRATION) +
  aes(y = SALARY) +
  geom_jitter(alpha = .5, height = 0, width = .25)
```



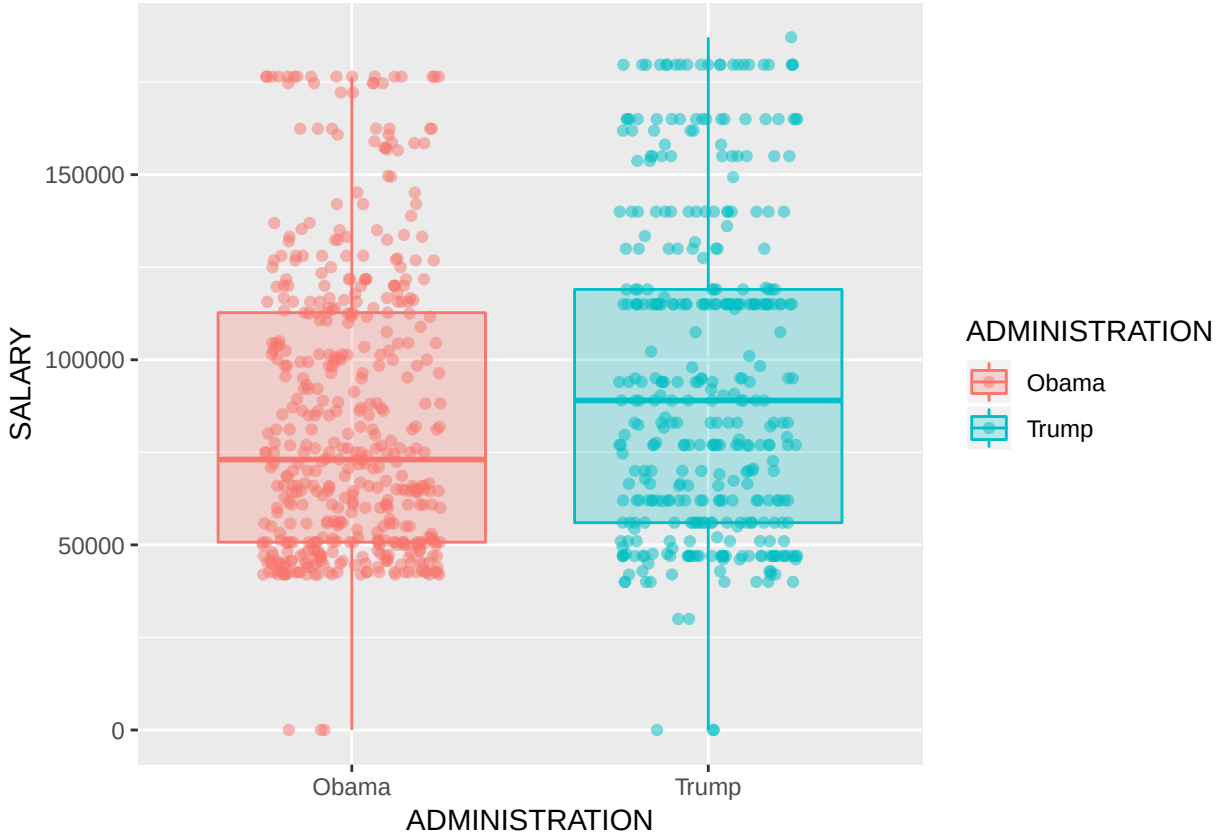
```
ggplot(both_data) +
  aes(x = ADMINISTRATION) +
  aes(y = SALARY) +
  geom_jitter(alpha = .5, height = 0, width = .25) +
  aes(col = ADMINISTRATION)
```



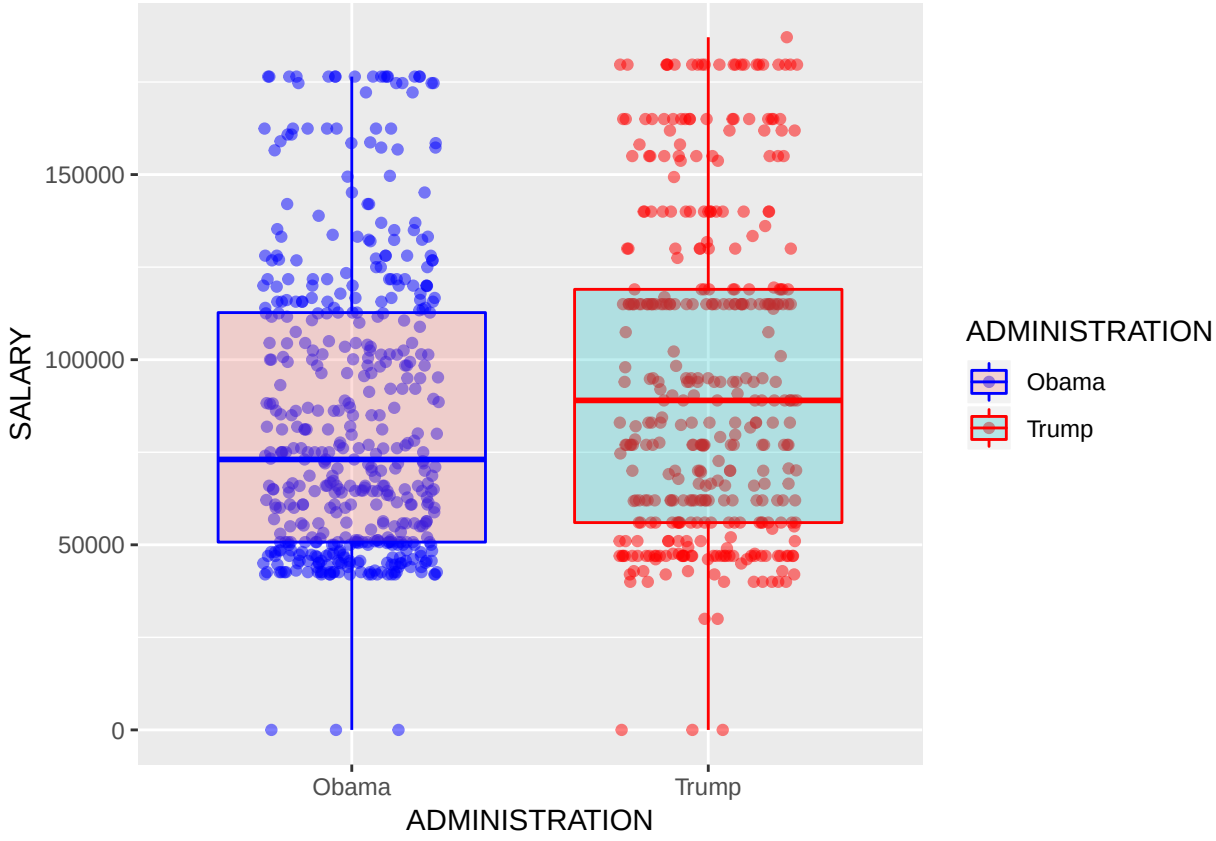
```
ggplot(both_data) +
  aes(x = ADMINISTRATION) +
  aes(y = SALARY) +
  geom_jitter(alpha = .5, height = 0, width = .25) +
  aes(col = ADMINISTRATION) +
  geom_boxplot(alpha = .25)
```



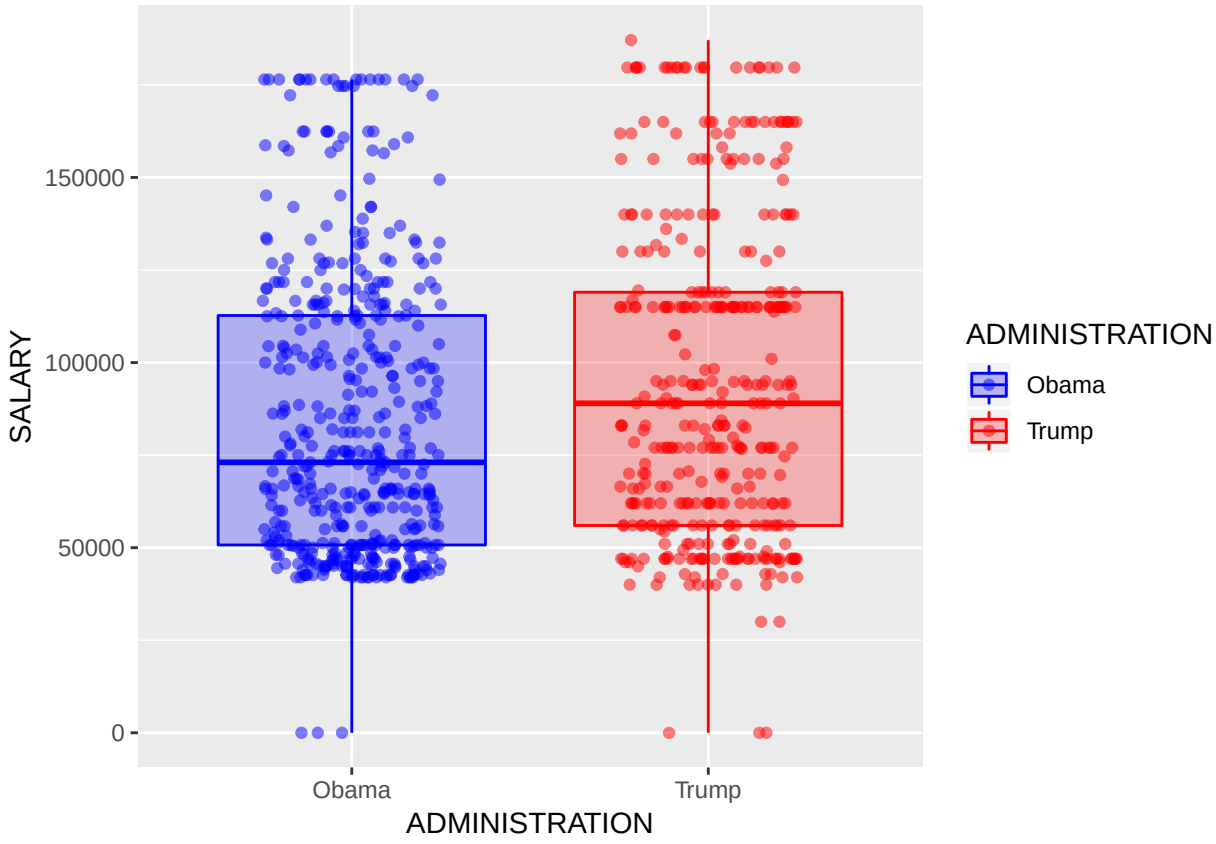
```
ggplot(both_data) +
  aes(x = ADMINISTRATION) +
  aes(y = SALARY) +
  geom_jitter(alpha = .5, height = 0, width = .25) +
  aes(col = ADMINISTRATION) +
  geom_boxplot(alpha = .25) +
  aes(fill = ADMINISTRATION)
```



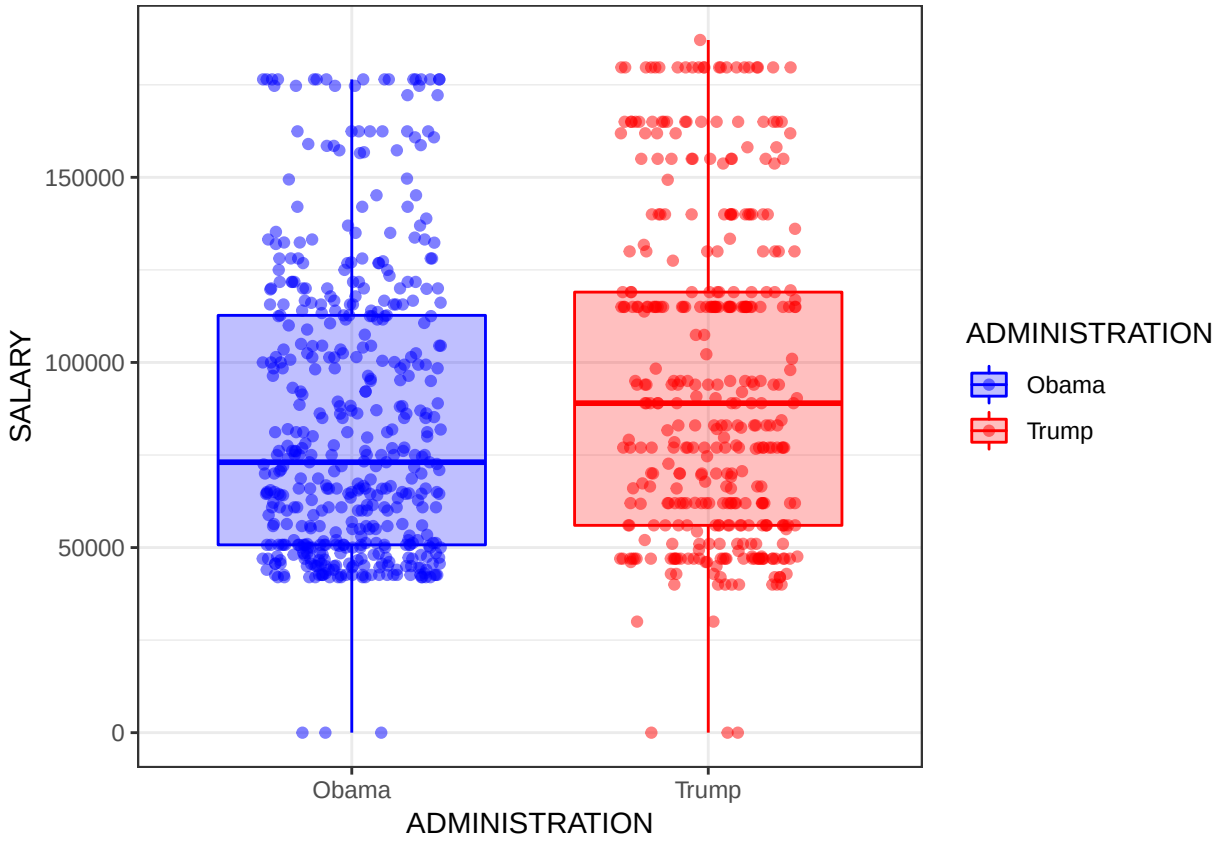
```
ggplot(both_data) +
  aes(x = ADMINISTRATION) +
  aes(y = SALARY) +
  geom_jitter(alpha = .5, height = 0, width = .25) +
  aes(col = ADMINISTRATION) +
  geom_boxplot(alpha = .25) +
  aes(fill = ADMINISTRATION) +
  scale_colour_manual(values = c("blue", "red"))
```



```
ggplot(both_data) +
  aes(x = ADMINISTRATION) +
  aes(y = SALARY) +
  geom_jitter(alpha = .5, height = 0, width = .25) +
  aes(col = ADMINISTRATION) +
  geom_boxplot(alpha = .25) +
  aes(fill = ADMINISTRATION) +
  scale_colour_manual(values = c("blue", "red")) +
  scale_fill_manual(values = c("blue", "red"))
```



```
ggplot(both_data) +
  aes(x = ADMINISTRATION) +
  aes(y = SALARY) +
  geom_jitter(alpha = .5, height = 0, width = .25) +
  aes(col = ADMINISTRATION) +
  geom_boxplot(alpha = .25) +
  aes(fill = ADMINISTRATION) +
  scale_colour_manual(values = c("blue", "red")) +
  scale_fill_manual(values = c("blue", "red")) +
  theme_bw()
```



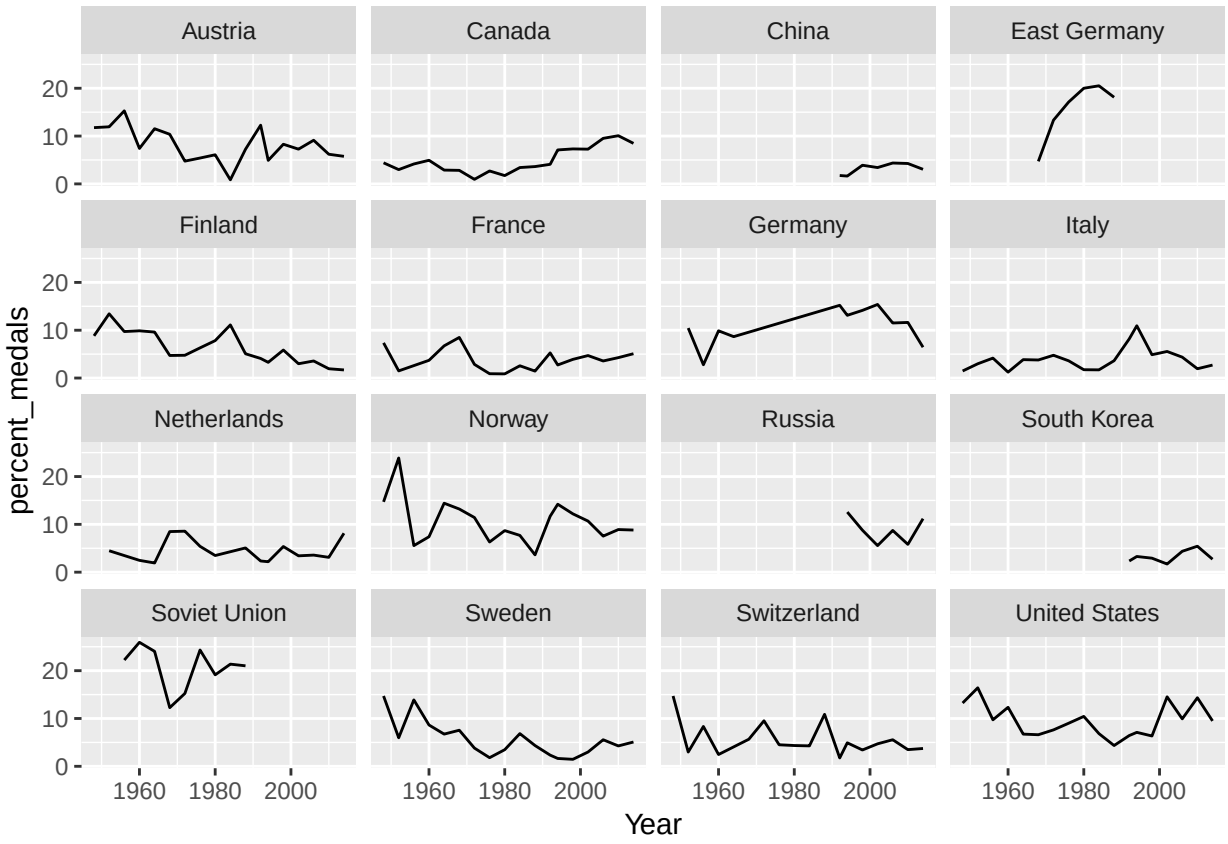
Chapter 8

Winter Games

A random sample from the data set:

Year	Sport	Event	Country	Gender	Medal Rank	Medal	Name of Athl
1994	Nordic Combined	Men's Team	Switzerland	Men	3	bronze	Switzerland
2014	Snowboarding	Women's Halfpipe	United States	Women	1	gold	Kaitlyn Farri
1968	Cross-Country Skiing	Women's 5 Kilometers	Sweden	Women	1	gold	Toini Gustafs
1972	Speedskating	Men's 10,000 Meters	Norway	Men	3	bronze	Sten Stensen
1968	Cross-Country Skiing	Women's 10 Kilometers	Norway	Women	3	bronze	Inger Aufles

```
ggplot(data = dta) +
  aes(x = Year) +
  aes(y = percent_medals) +
  geom_line() +
  facet_wrap(~ Country)
```



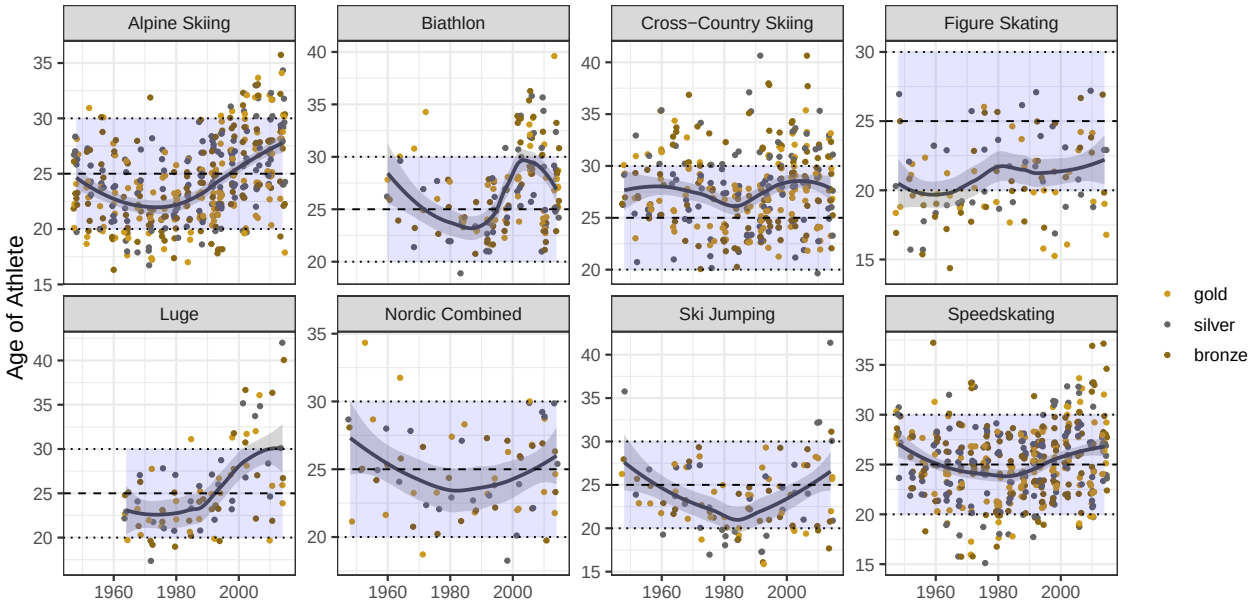
A random sample from the data set:

Year	Sport	Event	Country	Gender	Medal Rank	Medal	Name of Athlete or
2010	Alpine Skiing	Women's Giant Slalom	Slovenia	Women	2	silver	Tina Maze
1972	Luge	Women's Singles	East Germany	Women	2	silver	Ute RÃ¼hrold
1998	Luge	Men's Singles	Germany	Men	3	bronze	Jens MÃ¼ller
1976	Figure Skating	Mixed Ice Dancing	Soviet Union	Mixed	2	silver	Soviet Union-2
2010	Luge	Women's Singles	Germany	Women	3	bronze	Natalie Geisenberger

```
ggplot(dta) +
  aes(x = Year) +
  aes(y = `Age of Athlete`) +
  facet_wrap(~ Sport, scales = "free_y", nrow = 2) +
  geom_jitter(size = 1, mapping = aes(col = fct_inorder(Medal))) +
  geom_smooth(col = "grey30") +
  geom_ribbon(ymin = 20, ymax = 30, alpha = .1, fill = "blue") +
  geom_hline(yintercept = c(20, 30), lty = "dotted") +
  geom_hline(yintercept = c(25), lty = "dashed") +
  scale_color_manual(values = c("goldenrod3", "grey40", "goldenrod4"), name = "") +
  labs(x = "") +
  labs(title = "Young and old at the Winter Olympics: medalists' declared ages have risen in recent yea
  labs(subtitle = "Includes individual sports that have been in Olympic since 1965") +
  labs(caption = "Source: Sports-Reference.com | Vis: Gina Reynolds @EvaMaeRey \nValues 'jittered' to r
  theme_bw(base_size = 13)
```

Young and old at the Winter Olympics: medalists' declared ages have risen in recent years

Includes individual sports that have been in Olympic since 1965



Source: Sports−Reference.com | Vis: Gina Reynolds @EvaMaeRey
Values 'jittered' to reduce overplotting

Chapter 9

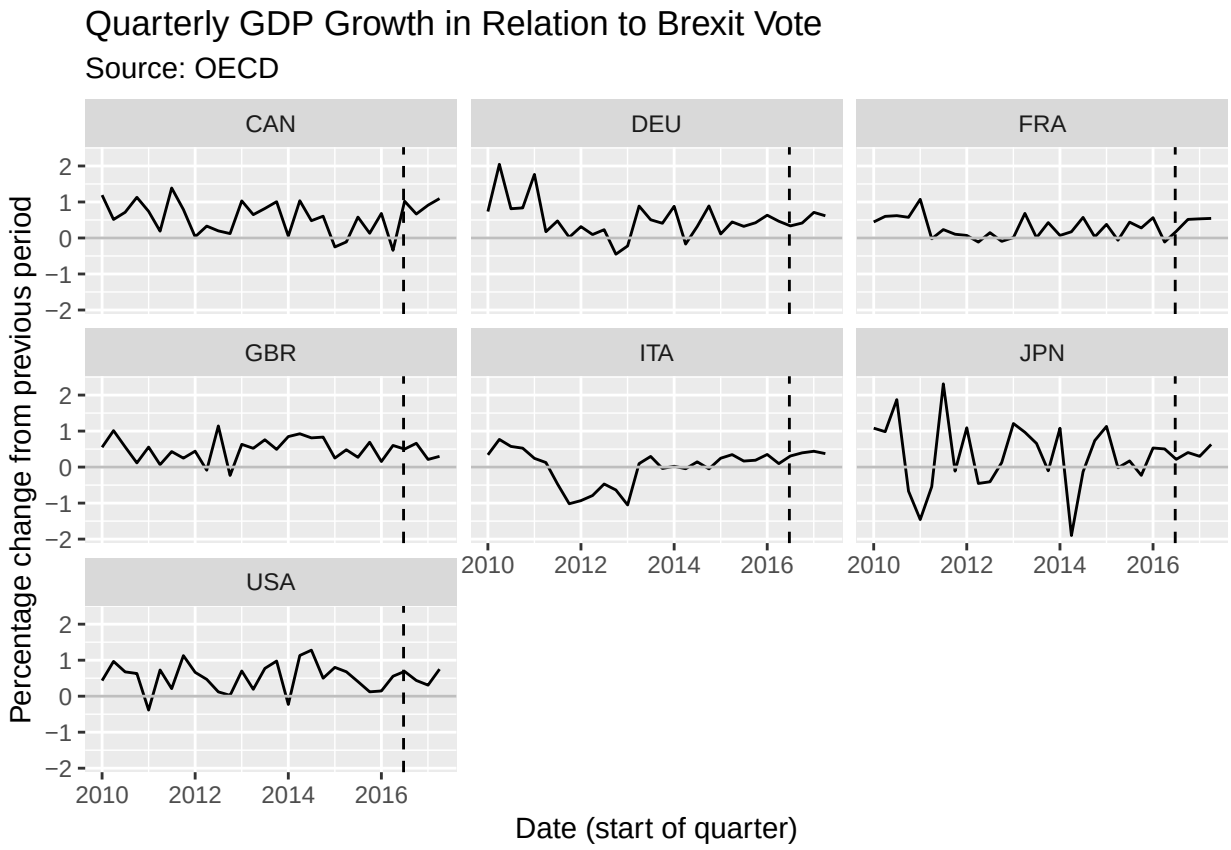
Brexit

This visualization challenge was a proposed makeover for a Financial Times visualization focusing on relative economic growth in G7 countries, with an emphasis on growth in the UK, focusing especially since Brexit. The visualization I present here is not what I created at the time of the challenge; instead it is inspired by Alan Smith a data journalist at the Financial Times, who created a really compelling visualization a couple of months after MakeoverMonday's treatment. I try to recreate his plot - which uses a ribbon to contain all G7 countries, and plot the UK's stats thereover. This declutters the graph, and makes you focus on where the UK falls among other countries, without being needlessly specific about those countries; the data story isn't about them anyway, might be Smith's thinking. My graph actually lightly traces economic growth in other countries, but deemphasizes their importance, like Smith.

A random sample from the data set:

Country	Year	Quarter	Date (start of quarter)	Percentage change from previous period	Date (start o quarter)
DEU	2014	1	2014-01-01	0.880174	2014-01-01
ITA	2010	1	2010-01-01	0.339939	2010-01-01
USA	2011	1	2011-01-01	-0.386237	2011-01-01
CAN	2013	2	2013-04-01	0.648283	2013-04-01
GBR	2014	2	2014-04-01	0.925355	2014-04-01

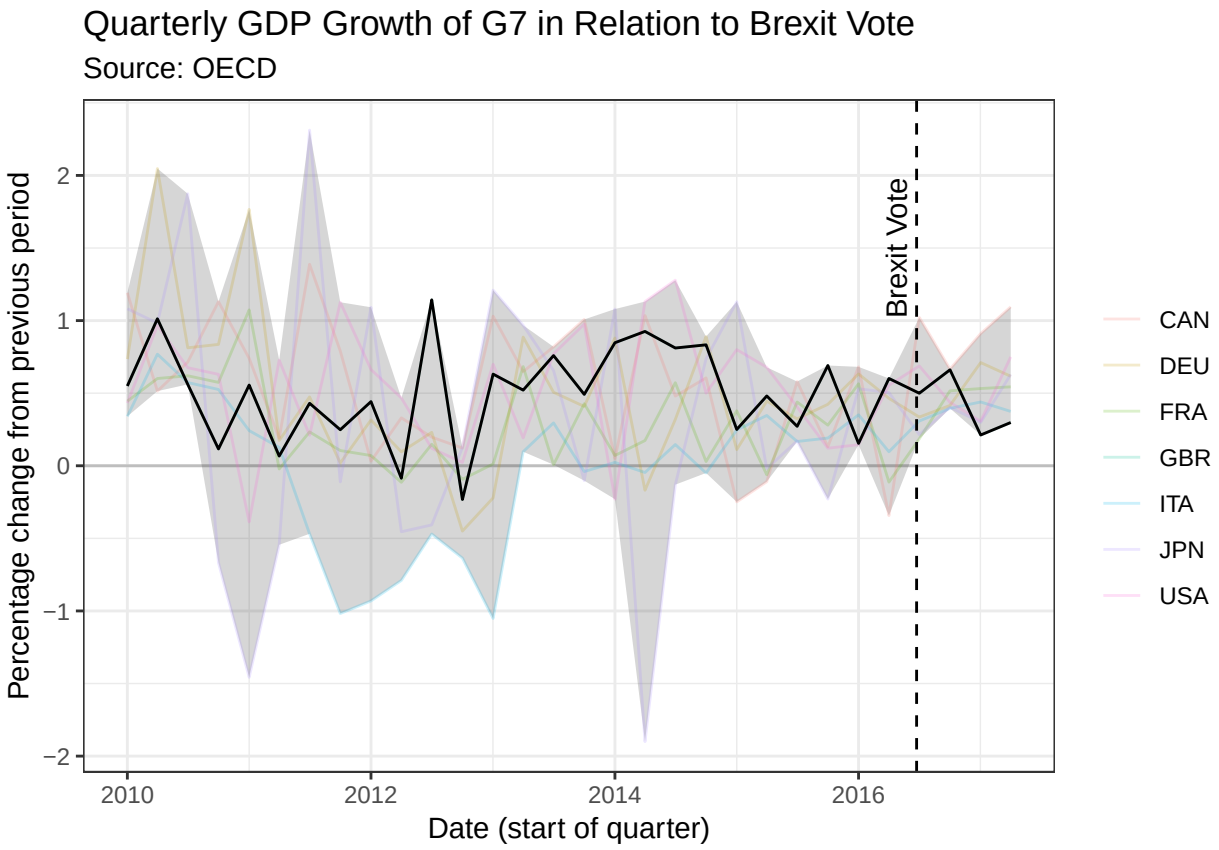
```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  facet_wrap(~ Country) +
  geom_line() +
  geom_hline(yintercept = 0, col = "grey") +
  geom_vline(xintercept = as.numeric(as.POSIXct("2016-06-23")), lty = "dashed") +
  labs(title = "Quarterly GDP Growth in Relation to Brexit Vote") +
  labs(subtitle = "Source: OECD")
```



A random sample from the data set:

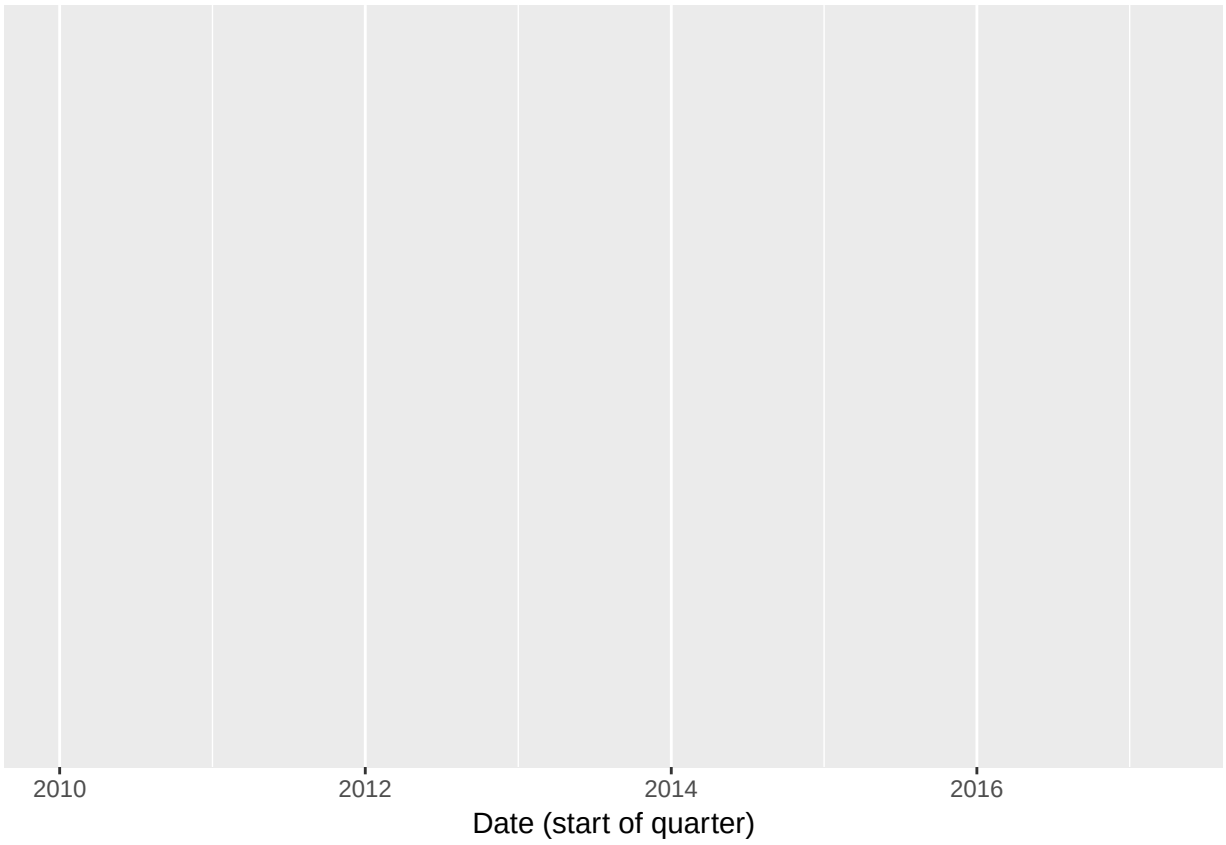
Country	Year	Quarter	Date (start of quarter)	Percentage change from previous period	Date (start o quarter)	
USA	2011	4	2011-10-01	1.126416	2011-10-01	-
FRA	2017	1	2017-01-01	0.531857	2017-01-01	
CAN	2015	4	2015-10-01	0.131436	2015-10-01	-
CAN	2012	2	2012-04-01	0.330094	2012-04-01	-
FRA	2017	2	2017-04-01	0.543614	2017-04-01	

```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_) +
  aes(ymax = max_) +
  geom_hline(yintercept = 0, col = "grey") +
  geom_ribbon(alpha = .2) +
  geom_line(aes(col = Country), alpha = .2) +
  geom_line(data = data_gbr, col = "black") +
  geom_vline(xintercept = as.numeric(as.POSIXct("2016-06-23")), lty = 2) +
  annotate(geom = "text", x = as.POSIXct("2016-04-23"), y = 1.5, label = "Brexit Vote", angle = 90) +
  labs(title = "Quarterly GDP Growth of G7 in Relation to Brexit Vote", subtitle = "Source: OECD", col =
  theme_bw()
```

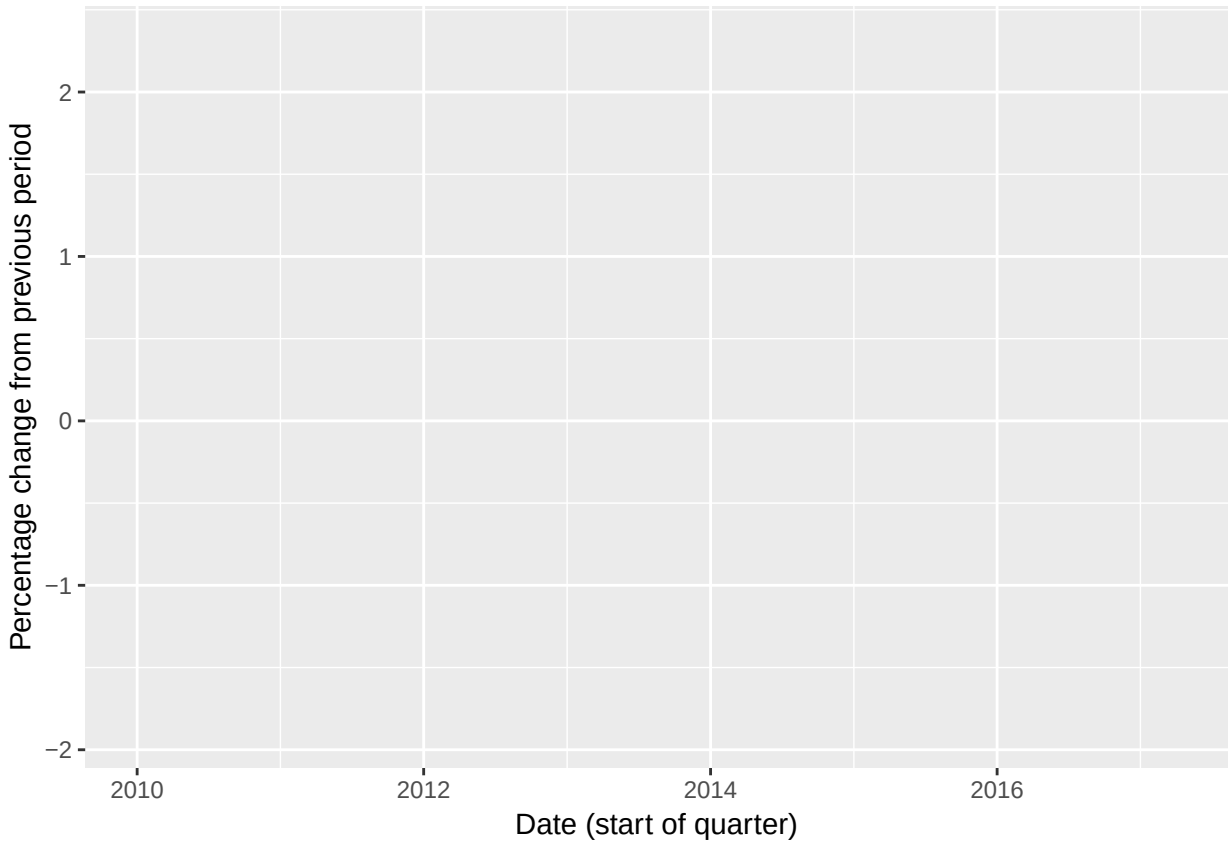

Quarterly GDP Growth of G7 in Relation to Brexit Vote
Source: OECD
Percentage change from previous period
2
1
0
−1
−2
Brexit Vote
2010
2012
2014
2016
Date (start of quarter)
CAN
DEU
FRA
GBR
ITA
JPN
USA

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

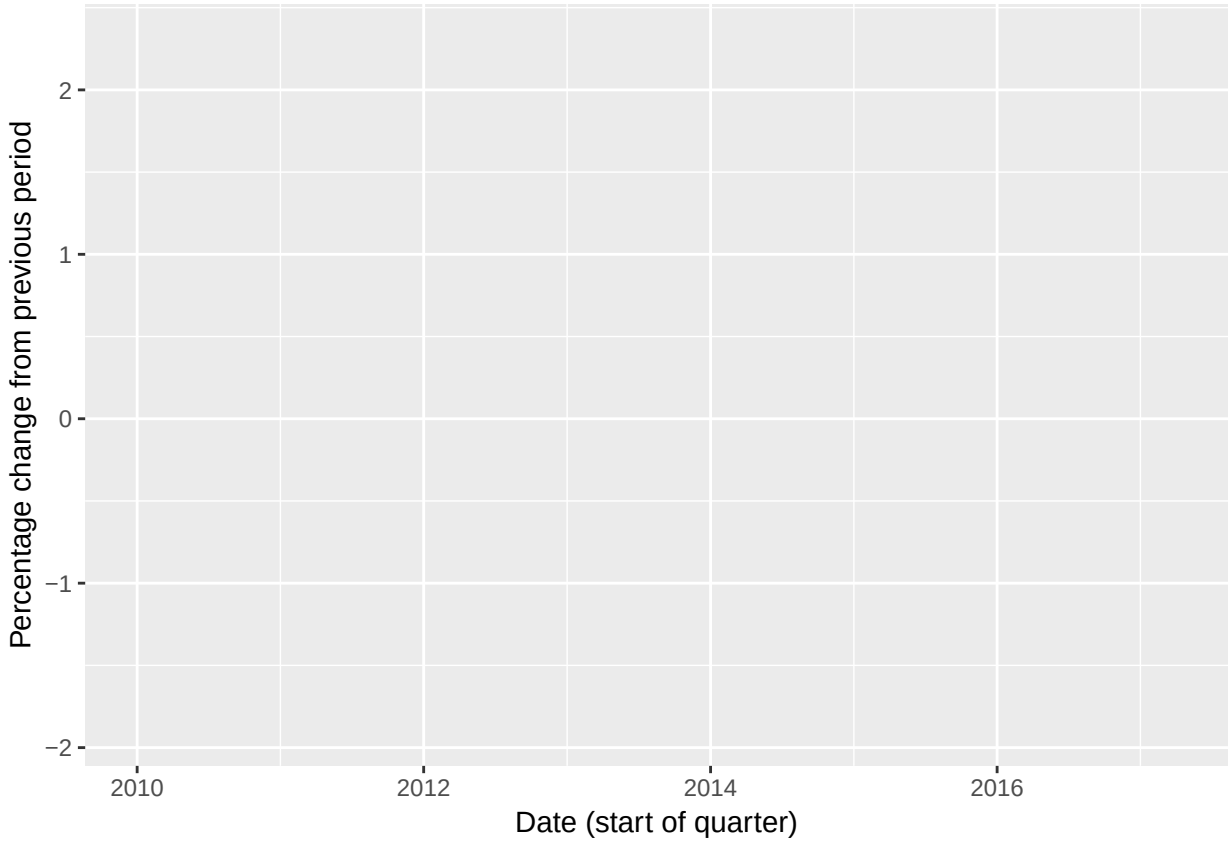
```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`)
```



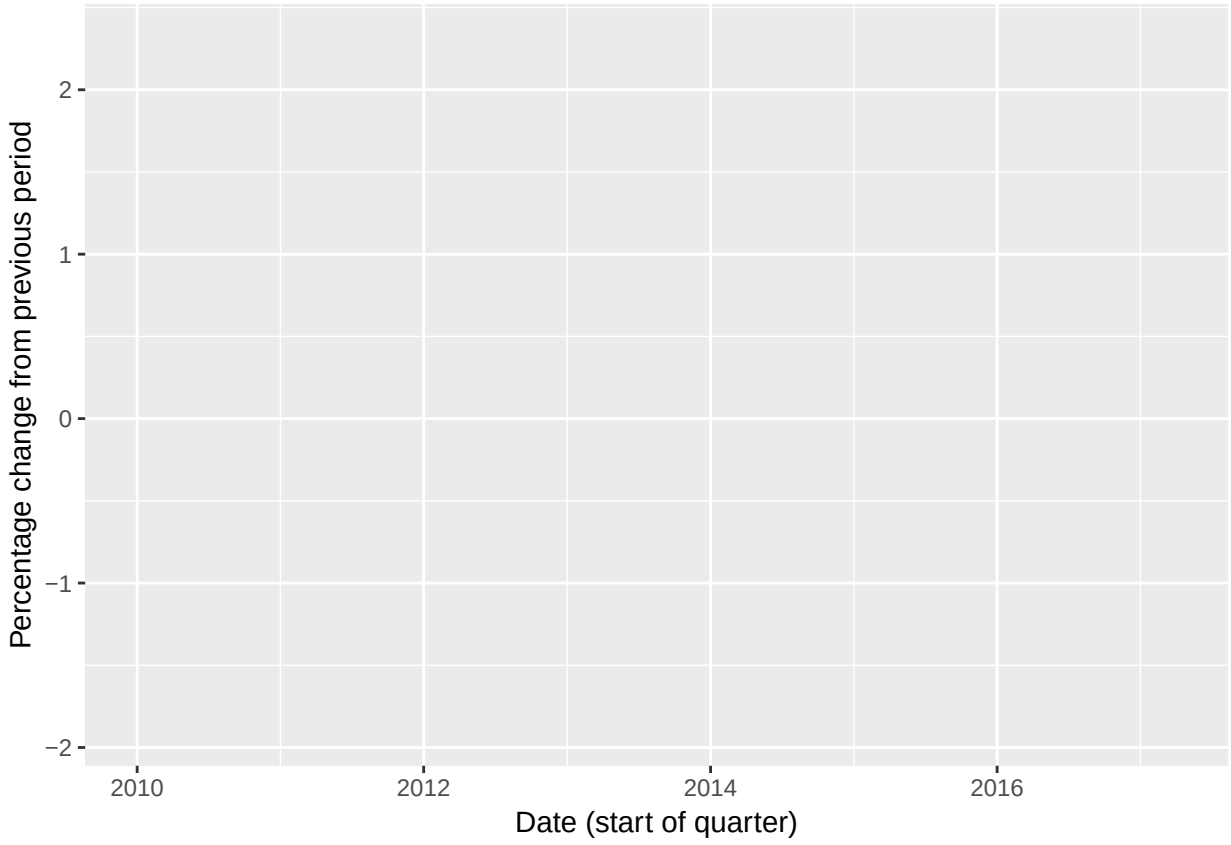
```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`)
```



```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_)
```

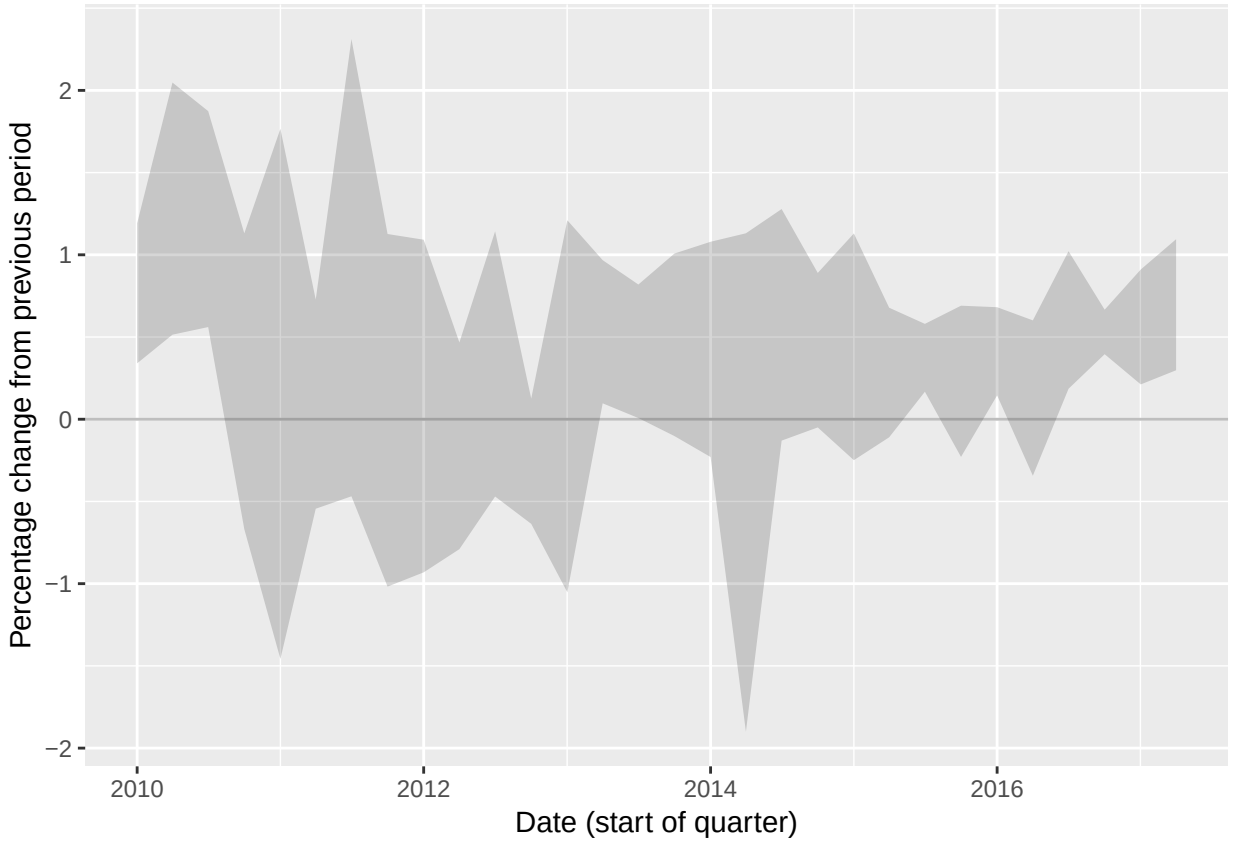


```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_) +
  aes(ymax = max_)
```



```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_) +
  aes(ymax = max_) +
  geom_hline(yintercept = 0, col = "grey")
```

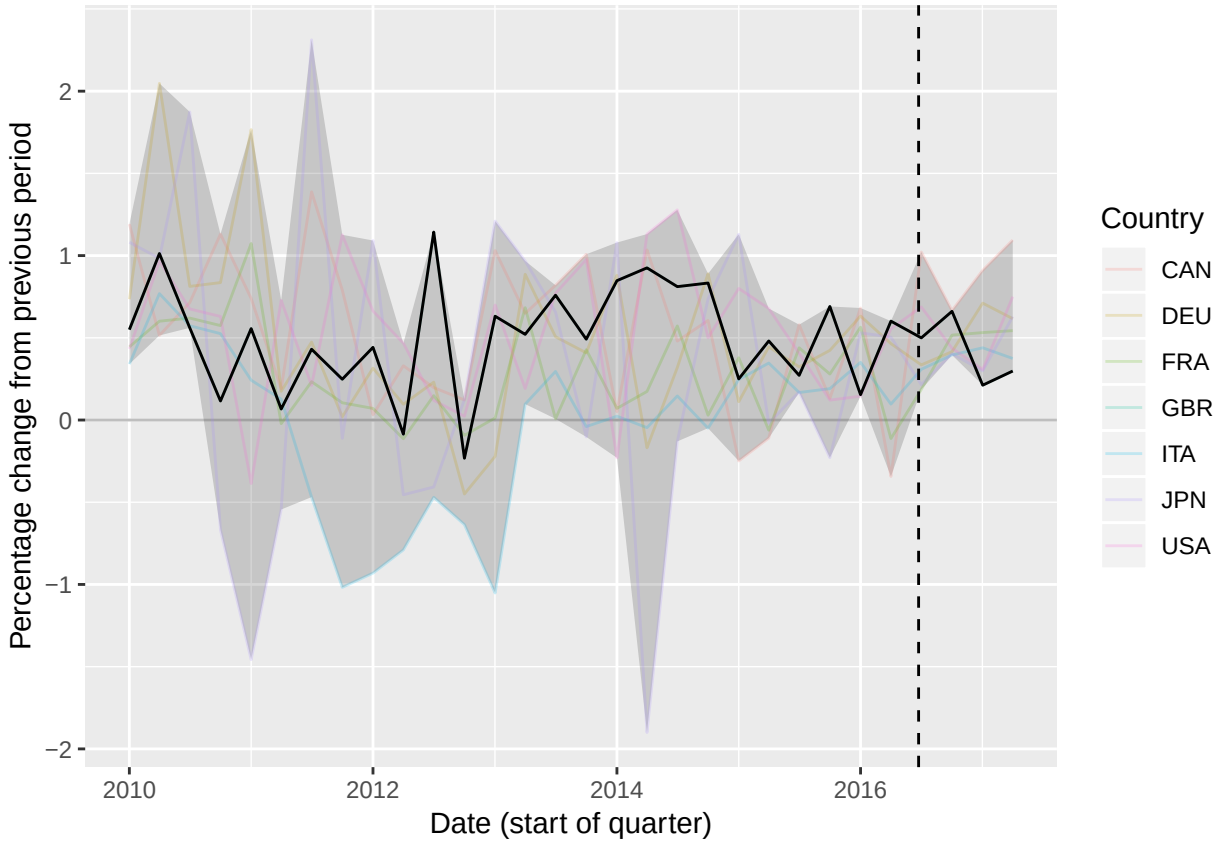
```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_) +
  aes(ymax = max_) +
  geom_hline(yintercept = 0, col = "grey") +
  geom_ribbon(alpha = .2)
```



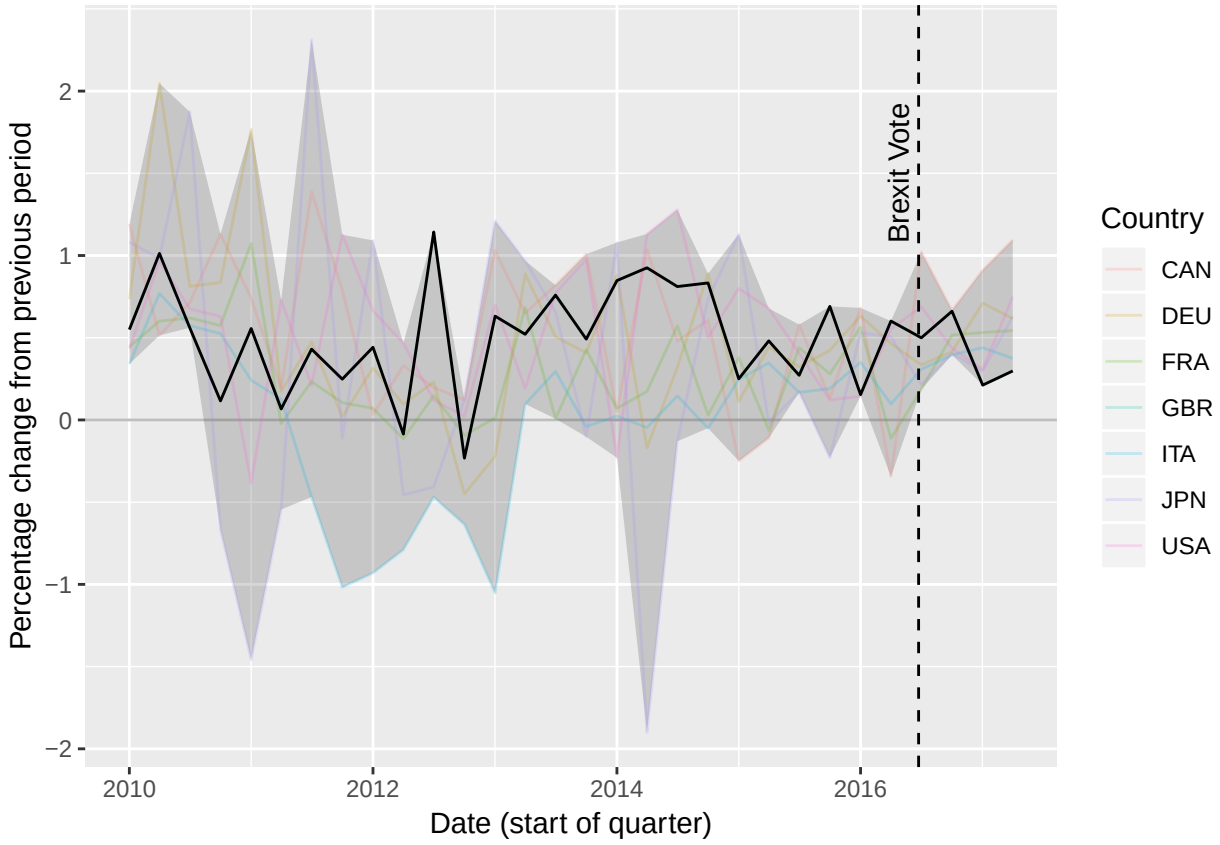
```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_) +
  aes(ymax = max_) +
  geom_hline(yintercept = 0, col = "grey") +
  geom_ribbon(alpha = .2) +
  geom_line(aes(col = Country), alpha = .2)
```

```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_) +
  aes(ymax = max_) +
  geom_hline(yintercept = 0, col = "grey") +
  geom_ribbon(alpha = .2) +
  geom_line(aes(col = Country), alpha = .2) +
  geom_line(data = data_gbr, col = "black")
```

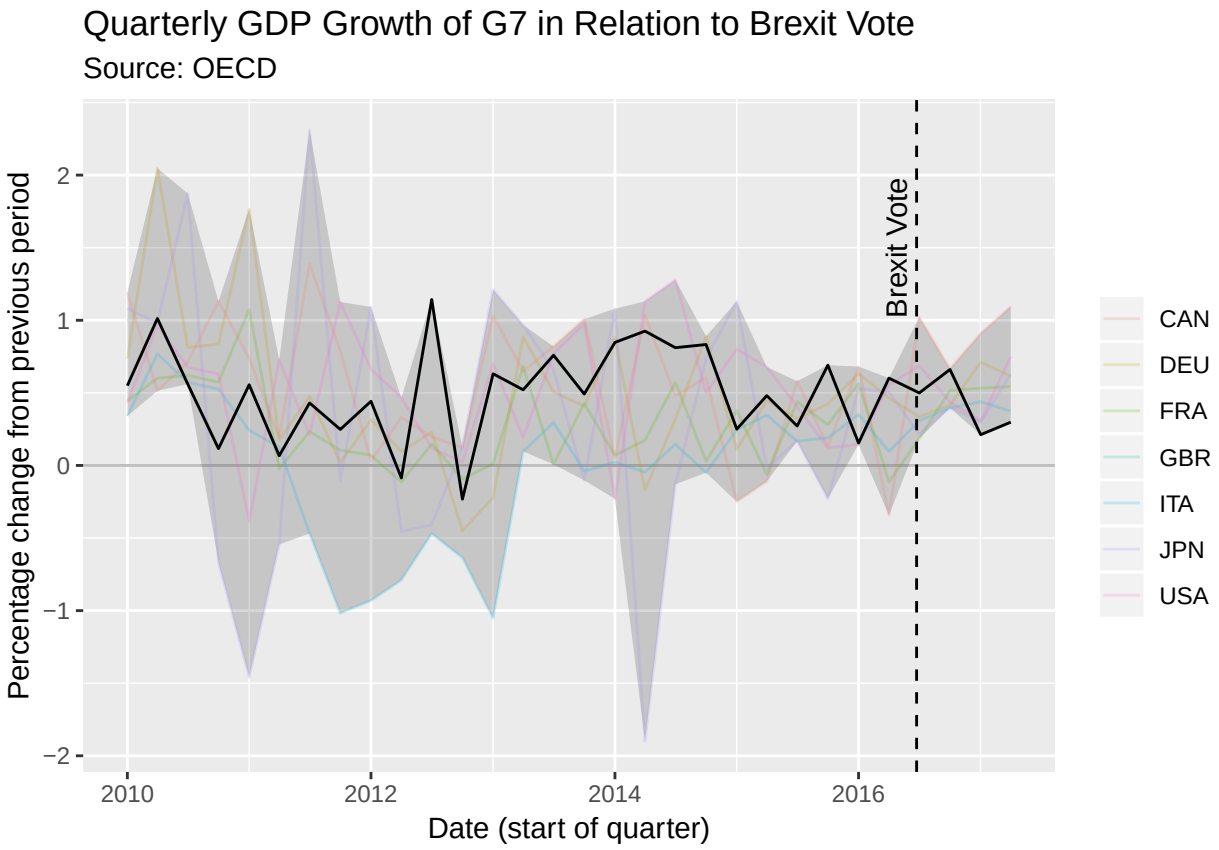
```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_) +
  aes(ymax = max_) +
  geom_hline(yintercept = 0, col = "grey") +
  geom_ribbon(alpha = .2) +
  geom_line(aes(col = Country), alpha = .2) +
  geom_line(data = data_gbr, col = "black") +
  geom_vline(xintercept = as.numeric(as.POSIXct("2016-06-23")), lty = 2)
```



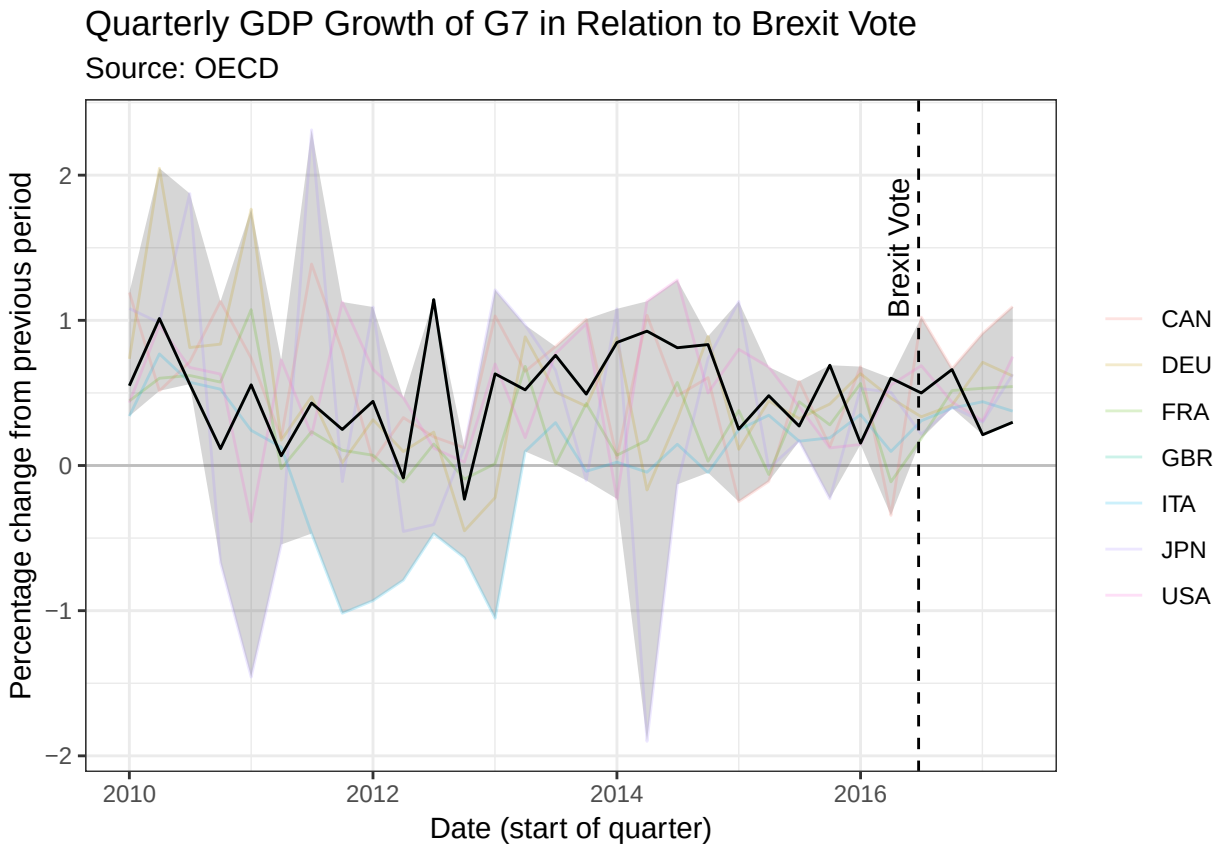
```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_) +
  aes(ymax = max_) +
  geom_hline(yintercept = 0, col = "grey") +
  geom_ribbon(alpha = .2) +
  geom_line(aes(col = Country), alpha = .2) +
  geom_line(data = data_gbr, col = "black") +
  geom_vline(xintercept = as.numeric(as.POSIXct("2016-06-23")), lty = 2) +
  annotate(geom = "text", x = as.POSIXct("2016-04-23"), y = 1.5, label = "Brexit Vote", angle = 90)
```



```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_) +
  aes(ymax = max_) +
  geom_hline(yintercept = 0, col = "grey") +
  geom_ribbon(alpha = .2) +
  geom_line(aes(col = Country), alpha = .2) +
  geom_line(data = data_gbr, col = "black") +
  geom_vline(xintercept = as.numeric(as.POSIXct("2016-06-23")), lty = 2) +
  annotate(geom = "text", x = as.POSIXct("2016-04-23"), y = 1.5, label = "Brexit Vote", angle = 90) +
  labs(title = "Quarterly GDP Growth of G7 in Relation to Brexit Vote", subtitle = "Source: OECD", col =
```



```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_) +
  aes(ymax = max_) +
  geom_hline(yintercept = 0, col = "grey") +
  geom_ribbon(alpha = .2) +
  geom_line(aes(col = Country), alpha = .2) +
  geom_line(data = data_gbr, col = "black") +
  geom_vline(xintercept = as.numeric(as.POSIXct("2016-06-23")), lty = 2) +
  annotate(geom = "text", x = as.POSIXct("2016-04-23"), y = 1.5, label = "Brexit Vote", angle = 90) +
  labs(title = "Quarterly GDP Growth of G7 in Relation to Brexit Vote", subtitle = "Source: OECD", col =
  theme_bw()
```



Chapter 10

Curry in London

This visualization task seemed to get at the question: Does where you eat matter. The data was the cost of identical menu items at different locations of the same restaurant, the Wetherspoon, around the UK.

First, I mapped the cost of a single menu item, the Empire Burger, across the UK. Then, I calculated the distance from Wetherspoon restaurants from the Big Ben, and plotted prices as a function of this distance – plotting only the restaurants in a 10 kilometer radius.

A random sample from the data set:

Name	Location	Latitude	Longitude	Empire State Burger	Chicken Tikka	Gammon afte
The Commercial Rooms	Bristol	51.45457	-2.5945465	9.25	7.99	
The Company Row	Consett	54.85210	-1.8309360	8.75	7.40	
Castle In the Air	Manchester	53.46868	-2.3610950	11.49	8.40	
The Draper's Arms	Peterborough	52.57300	-0.2463635	8.75	8.29	
The Society Room	Glasgow	55.86232	-4.2571650	9.45	7.40	

```
# Mapping data
world_map_df <- map_data("world")
```

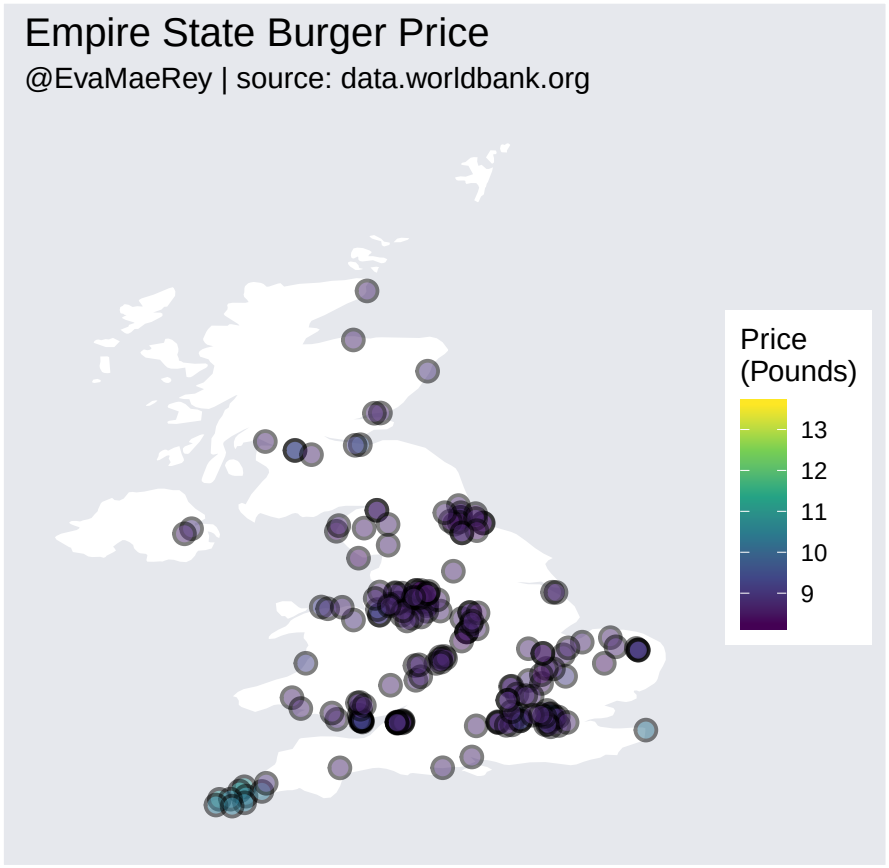
A random sample from the data set:

	long	lat	group	order	region	subregion
48898	135.03740	-3.333105	780	48898	Indonesia	Irian Jaya
78357	73.39941	53.811474	1309	78357	Russia	NA
28121	-72.49414	-51.847561	417	28121	Chile	NA
19669	-82.64204	69.458397	345	19669	Canada	NA
53848	35.40869	31.482910	850	53848	Israel	NA

```
# create a blank ggplot theme
theme_opts <- theme(
  panel.grid.minor = element_blank(),
  panel.grid.major = element_blank(),
  panel.background = element_blank(),
  plot.background = element_rect(fill = "#e6e8ed"),
  panel.border = element_blank(),
  axis.line = element_blank(),
  axis.text.x = element_blank(),
  axis.text.y = element_blank(),
  axis.ticks = element_blank(),
  axis.title.x = element_blank(),
  axis.title.y = element_blank(),
```

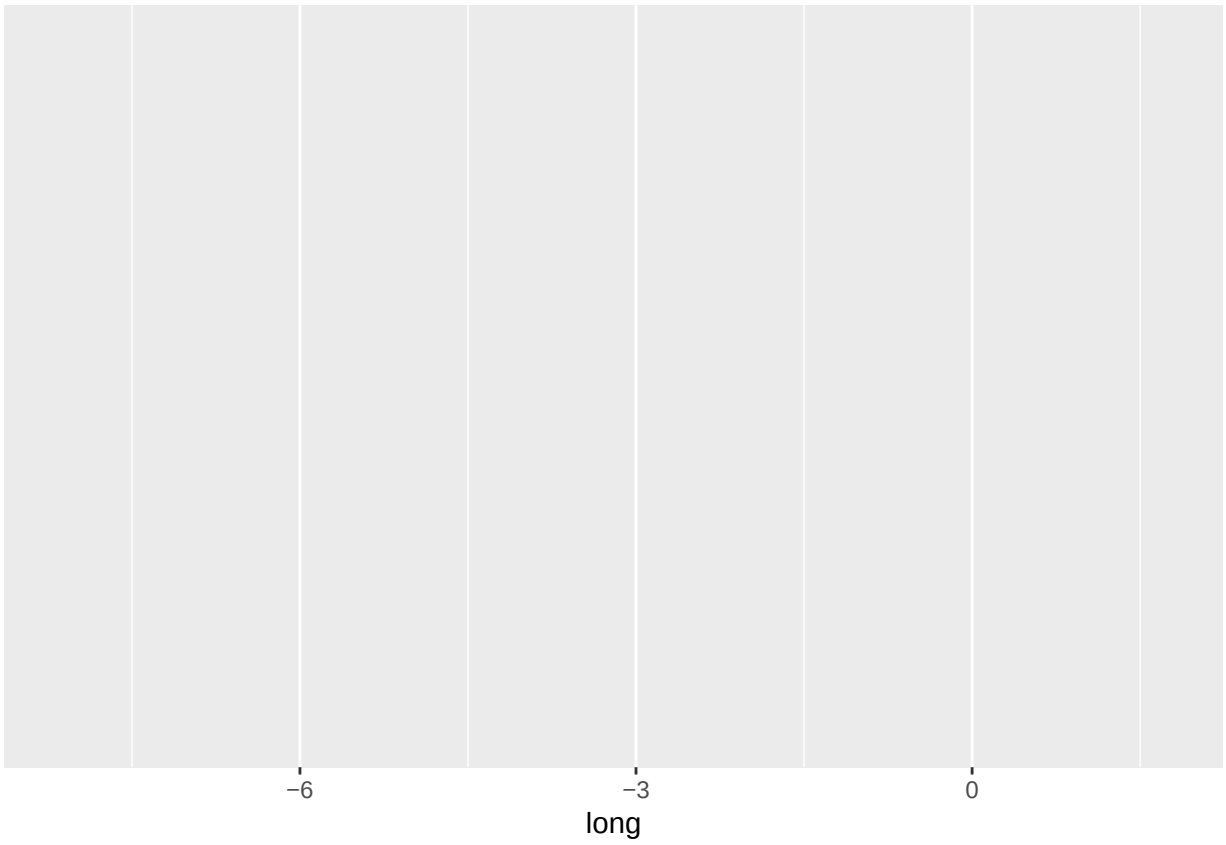
```
  plot.title = element_text(size = 15)
)

data_UK <- world_map_df %>% filter(region == "UK")
```

```
ggplot(data = data_UK) +
  aes(x = long) +
  aes(y = lat) +
  aes(group = group) +
  geom_polygon(fill = "white") +
  coord_equal() +
  scale_fill_viridis_c(option = "viridis") +
  geom_point(data = data0, mapping = aes(x = Longitude, y = Latitude, group = NULL, fill = `Empire State
  labs(fill = "Price\n(Pounds)") +
  labs(title = "Empire State Burger Price") +
  labs(subtitle = "@EvaMaeRey | source: data.worldbank.org") +
  theme_opts
```

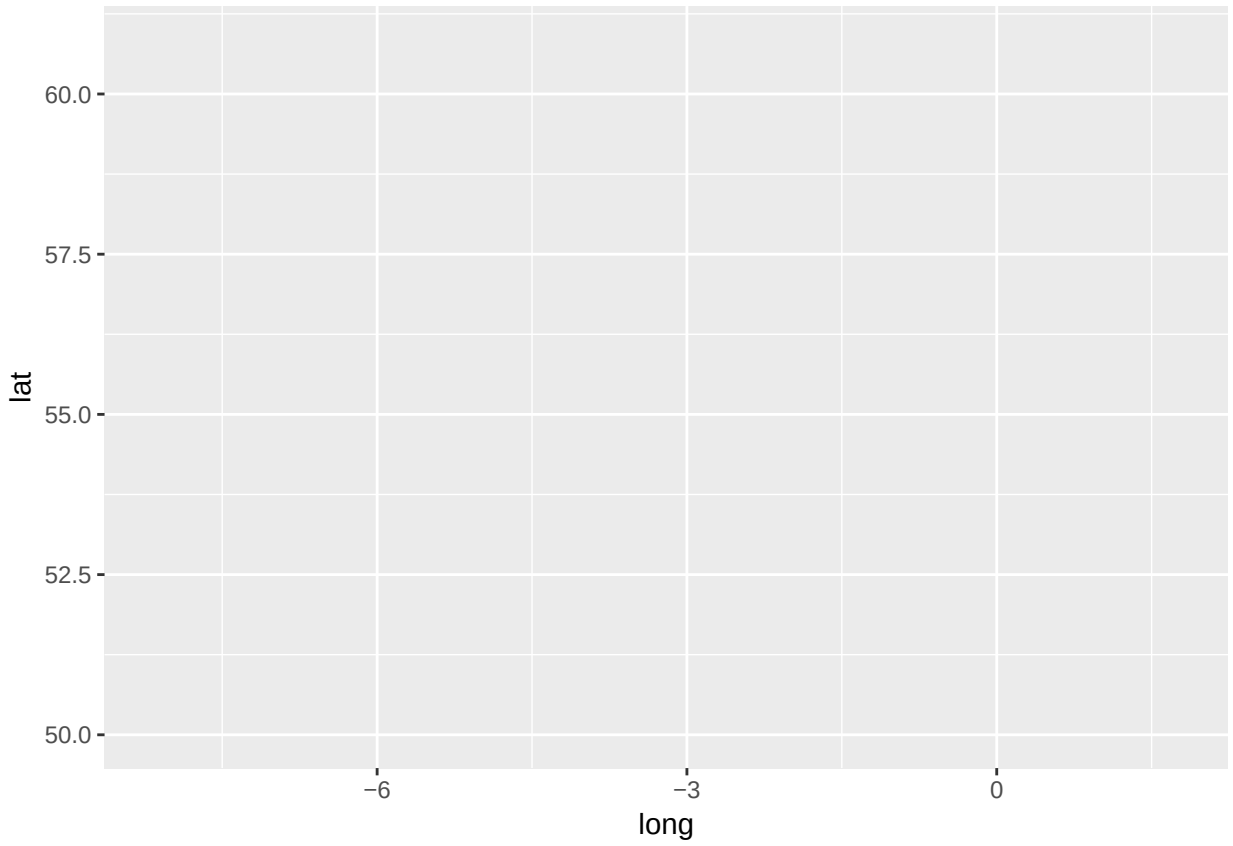


```
ggplot(data = data_UK)
```

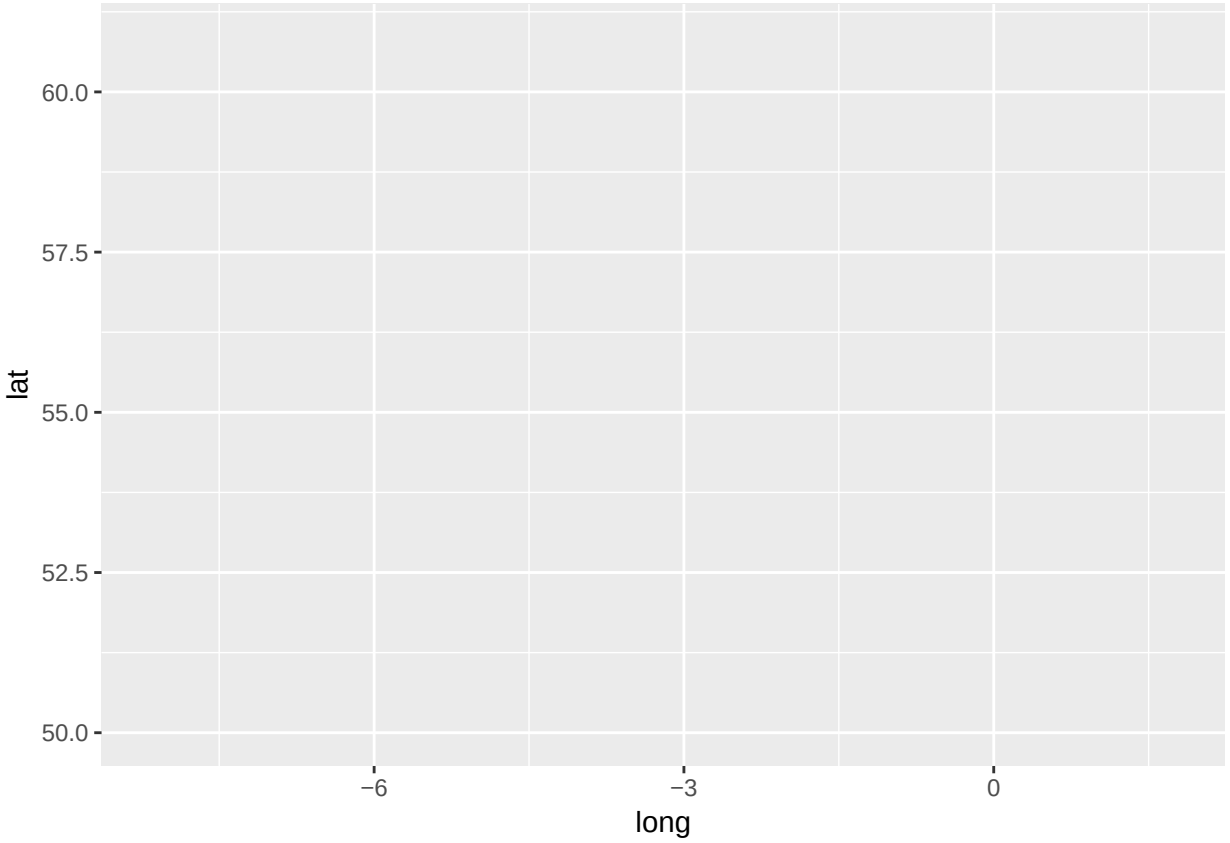
```
ggplot(data = data_UK) +
  aes(x = long)
```



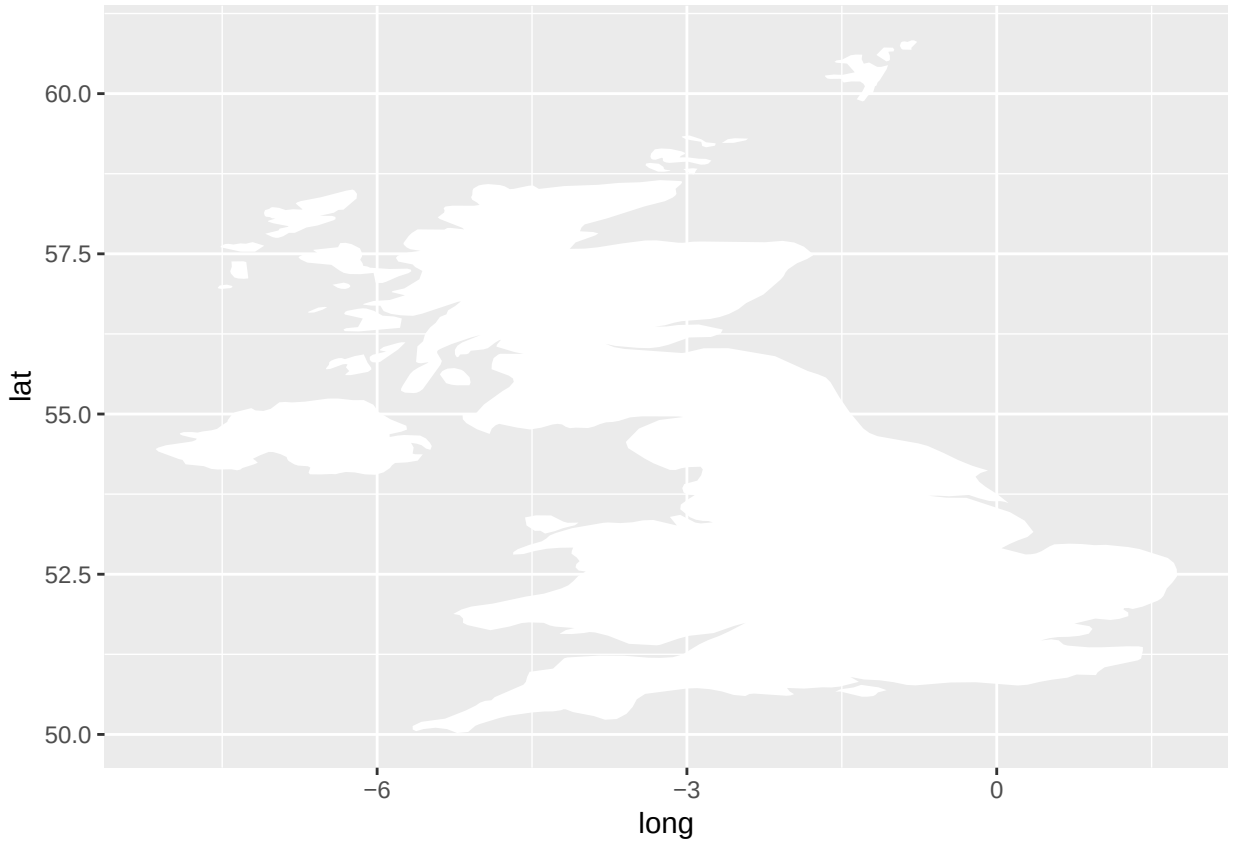
```
ggplot(data = data_UK) +
  aes(x = long) +
  aes(y = lat)
```



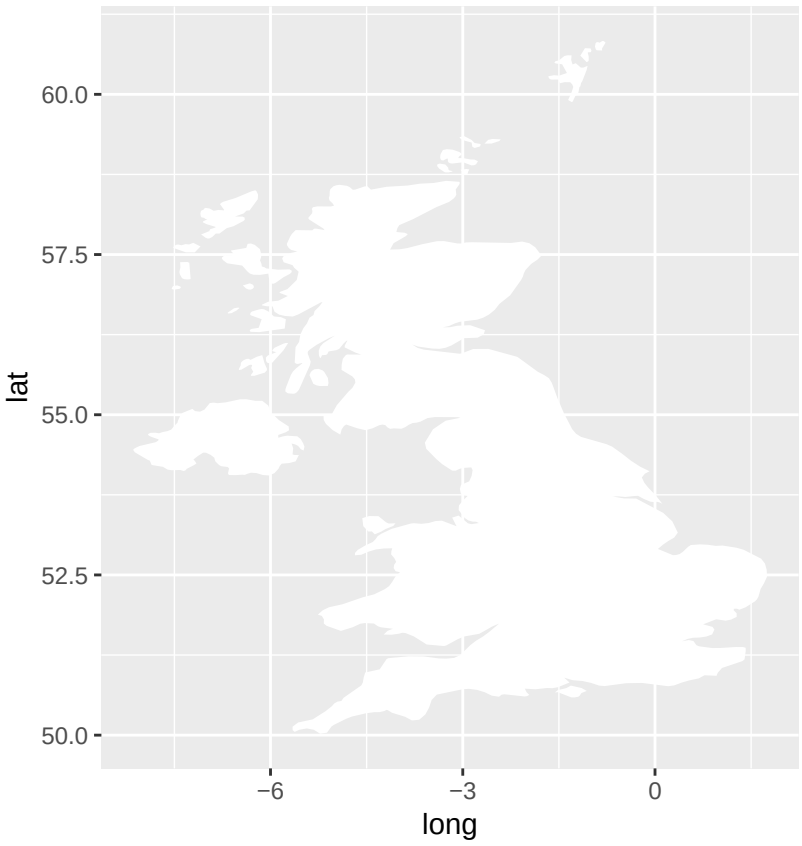
```
ggplot(data = data_UK) +
  aes(x = long) +
  aes(y = lat) +
  aes(group = group)
```



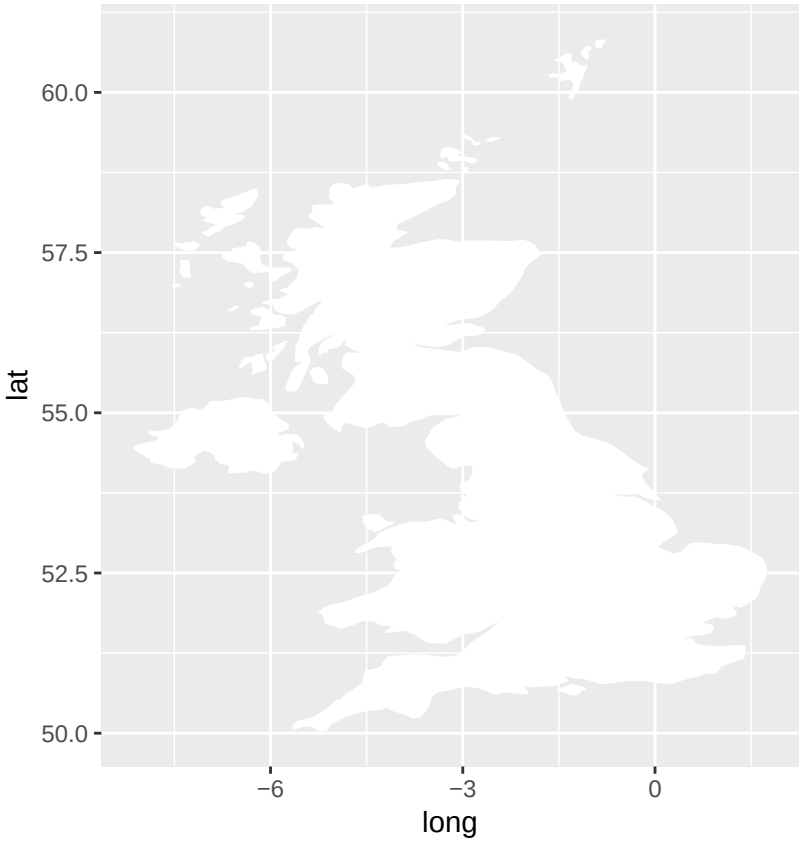
```
ggplot(data = data_UK) +
  aes(x = long) +
  aes(y = lat) +
  aes(group = group) +
  geom_polygon(fill = "white")
```



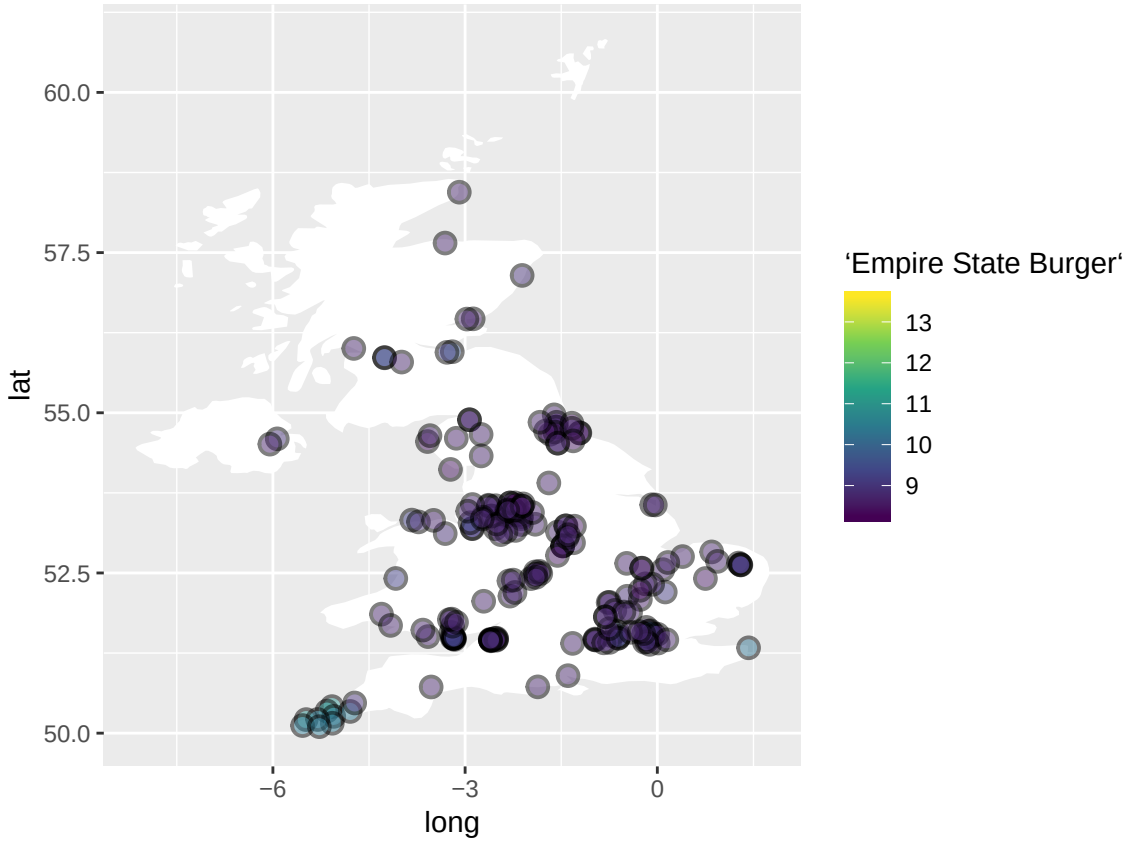
```
ggplot(data = data_UK) +
  aes(x = long) +
  aes(y = lat) +
  aes(group = group) +
  geom_polygon(fill = "white") +
  coord_equal()
```



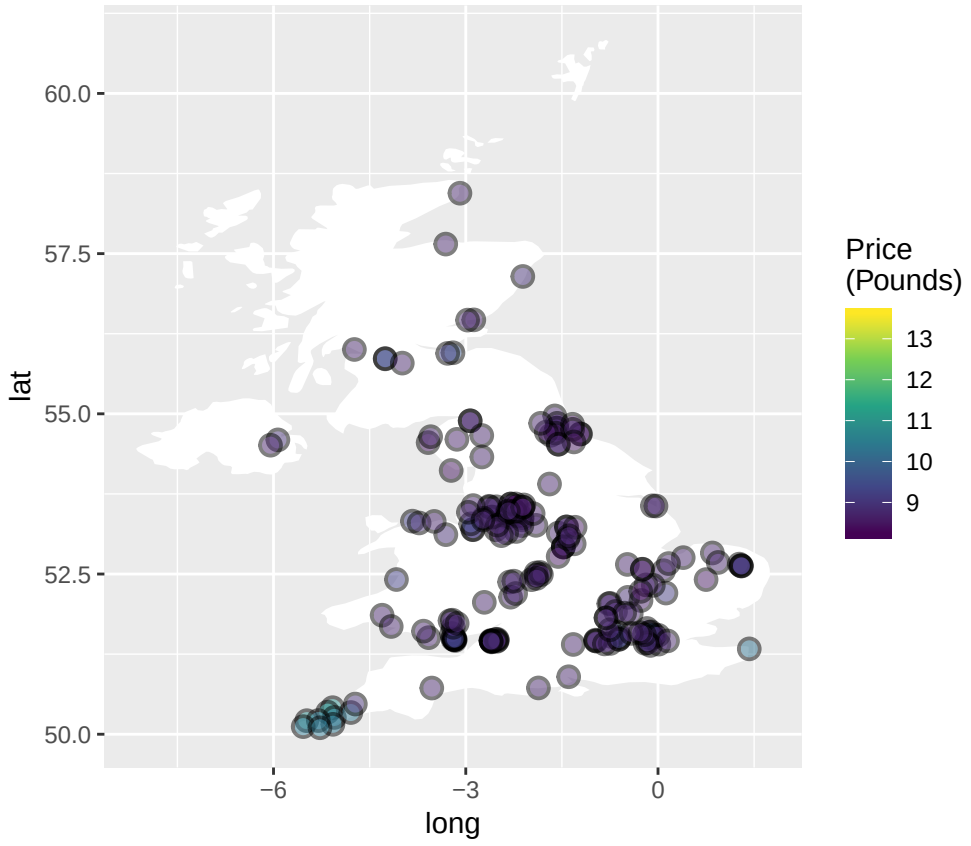
```
ggplot(data = data_UK) +
  aes(x = long) +
  aes(y = lat) +
  aes(group = group) +
  geom_polygon(fill = "white") +
  coord_equal() +
  scale_fill_viridis_c(option = "viridis")
```



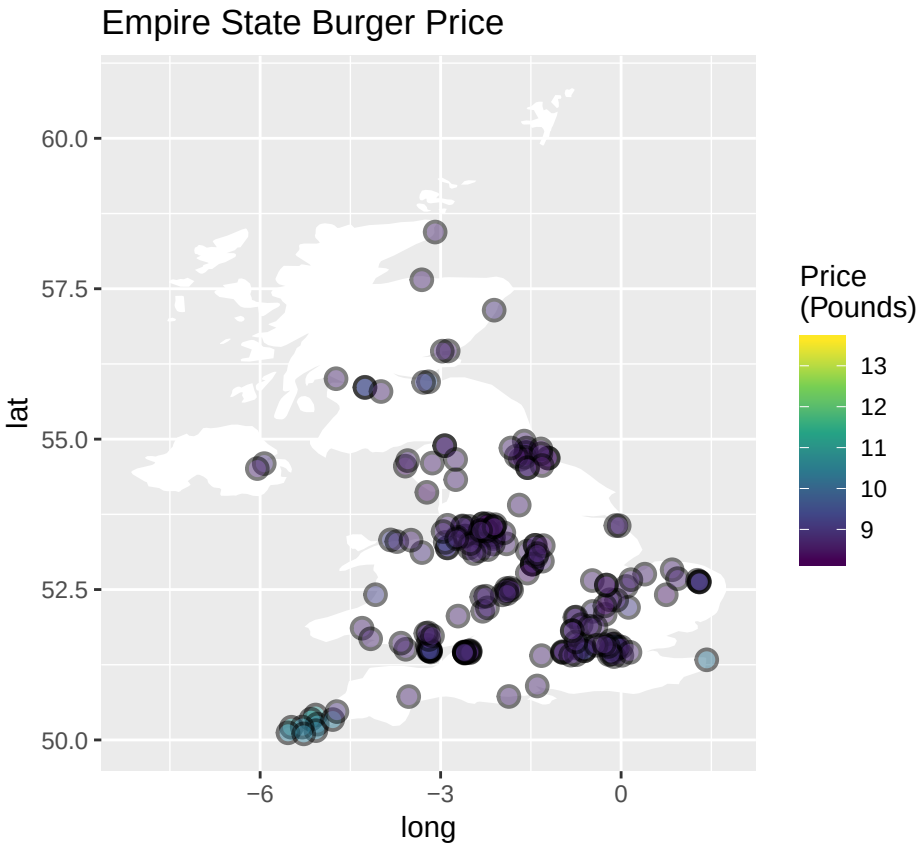
```
ggplot(data = data_UK) +
  aes(x = long) +
  aes(y = lat) +
  aes(group = group) +
  geom_polygon(fill = "white") +
  coord_equal() +
  scale_fill_viridis_c(option = "viridis") +
  geom_point(data = data0, mapping = aes(x = Longitude, y = Latitude, group = NULL, fill = `Empire State
```



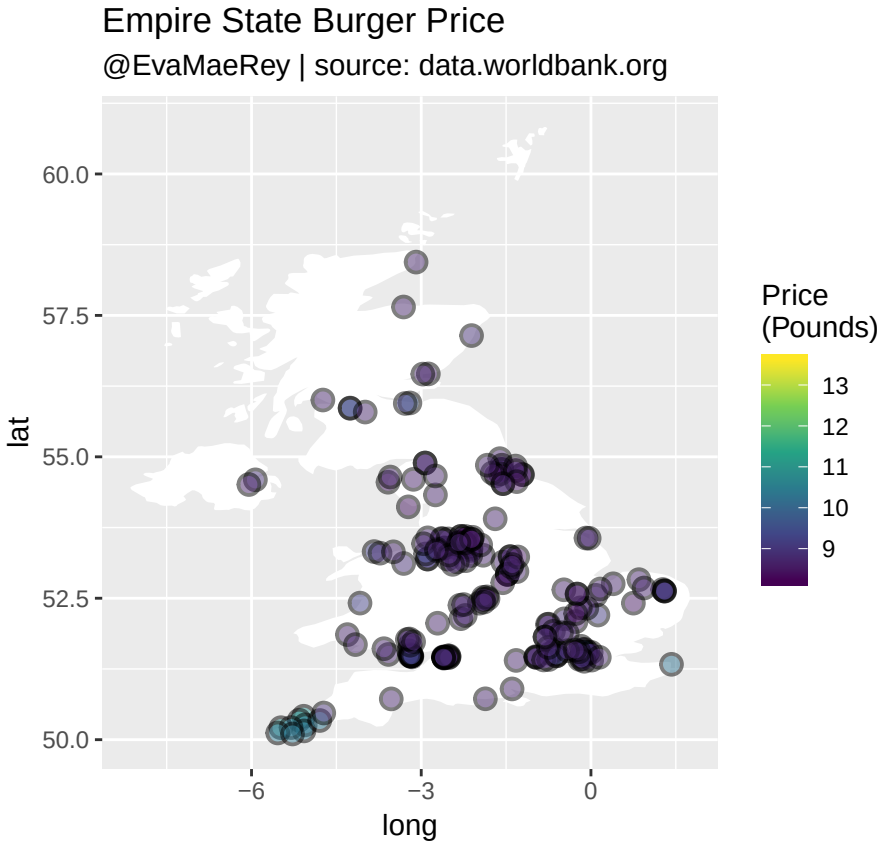
```
ggplot(data = data_UK) +
  aes(x = long) +
  aes(y = lat) +
  aes(group = group) +
  geom_polygon(fill = "white") +
  coord_equal() +
  scale_fill_viridis_c(option = "viridis") +
  geom_point(data = data0, mapping = aes(x = Longitude, y = Latitude, group = NULL, fill = `Empire State
  labs(fill = "Price\n(Pounds)")
```



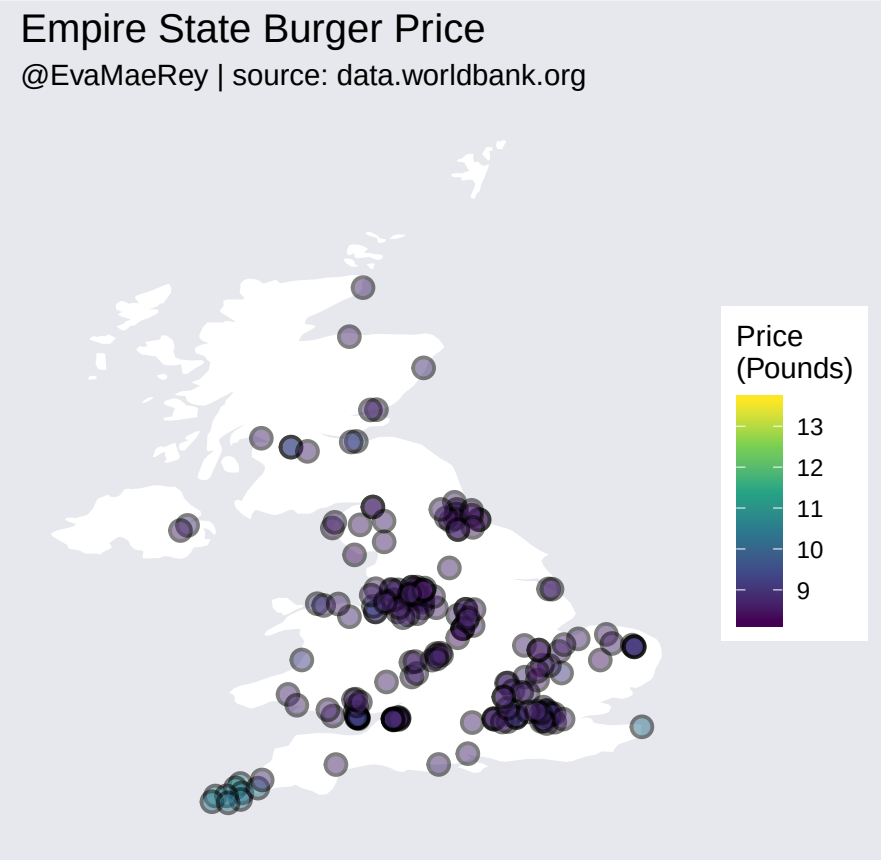
```
ggplot(data = data_UK) +
  aes(x = long) +
  aes(y = lat) +
  aes(group = group) +
  geom_polygon(fill = "white") +
  coord_equal() +
  scale_fill_viridis_c(option = "viridis") +
  geom_point(data = data0, mapping = aes(x = Longitude, y = Latitude, group = NULL, fill = `Empire State
  labs(fill = "Price\n(Pounds)") +
  labs(title = "Empire State Burger Price")
```



```
ggplot(data = data_UK) +
  aes(x = long) +
  aes(y = lat) +
  aes(group = group) +
  geom_polygon(fill = "white") +
  coord_equal() +
  scale_fill_viridis_c(option = "viridis") +
  geom_point(data = data0, mapping = aes(x = Longitude, y = Latitude, group = NULL, fill = `Empire State
  labs(fill = "Price\n(Pounds)") +
  labs(title = "Empire State Burger Price") +
  labs(subtitle = "@EvaMaeRey | source: data.worldbank.org")
```



```
ggplot(data = data_UK) +
  aes(x = long) +
  aes(y = lat) +
  aes(group = group) +
  geom_polygon(fill = "white") +
  coord_equal() +
  scale_fill_viridis_c(option = "viridis") +
  geom_point(data = data0, mapping = aes(x = Longitude, y = Latitude, group = NULL, fill = `Empire State
  labs(fill = "Price\n(Pounds)") +
  labs(title = "Empire State Burger Price") +
  labs(subtitle = "@EvaMaeRey | source: data.worldbank.org") +
  theme_opts
```

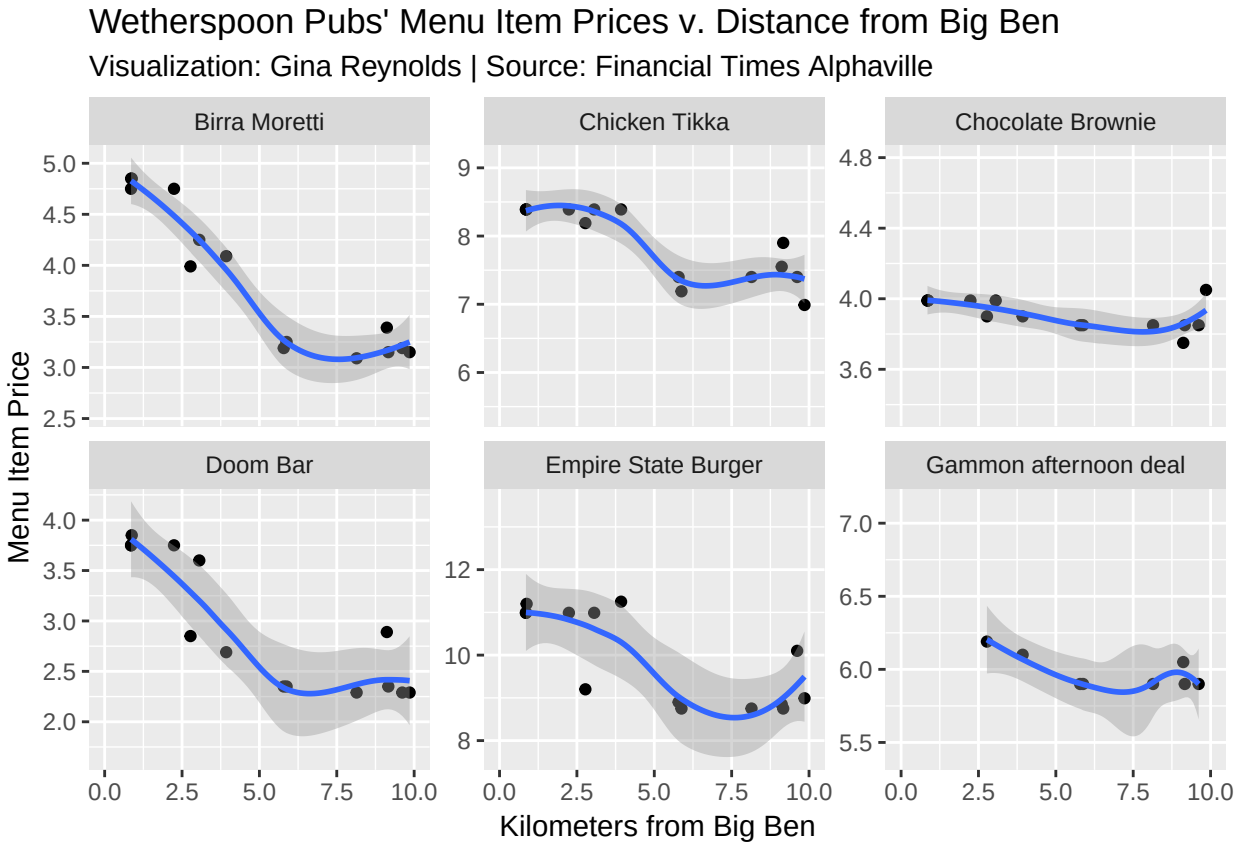


A random sample from the data set:

Name	Location	Latitude	Longitude	Notes	Moretti as a % of a tikka	Moretti as % of burger
Castle In the Air	Manchester	53.46868	-2.361095	NA	0.4988095	0.3646649
The Bishop Blaize	Stretford	53.46126	-2.289141	NA	0.4436718	0.3645714
Castle In the Air	Manchester	53.46868	-2.361095	NA	0.4988095	0.3646649
The Aneurin Bevin	Cardiff	51.50772	-3.200037	NA	0.3754470	0.3405405
The Muckle Cross	Elgin	57.64887	-3.312444	NA	0.3770270	0.3188571

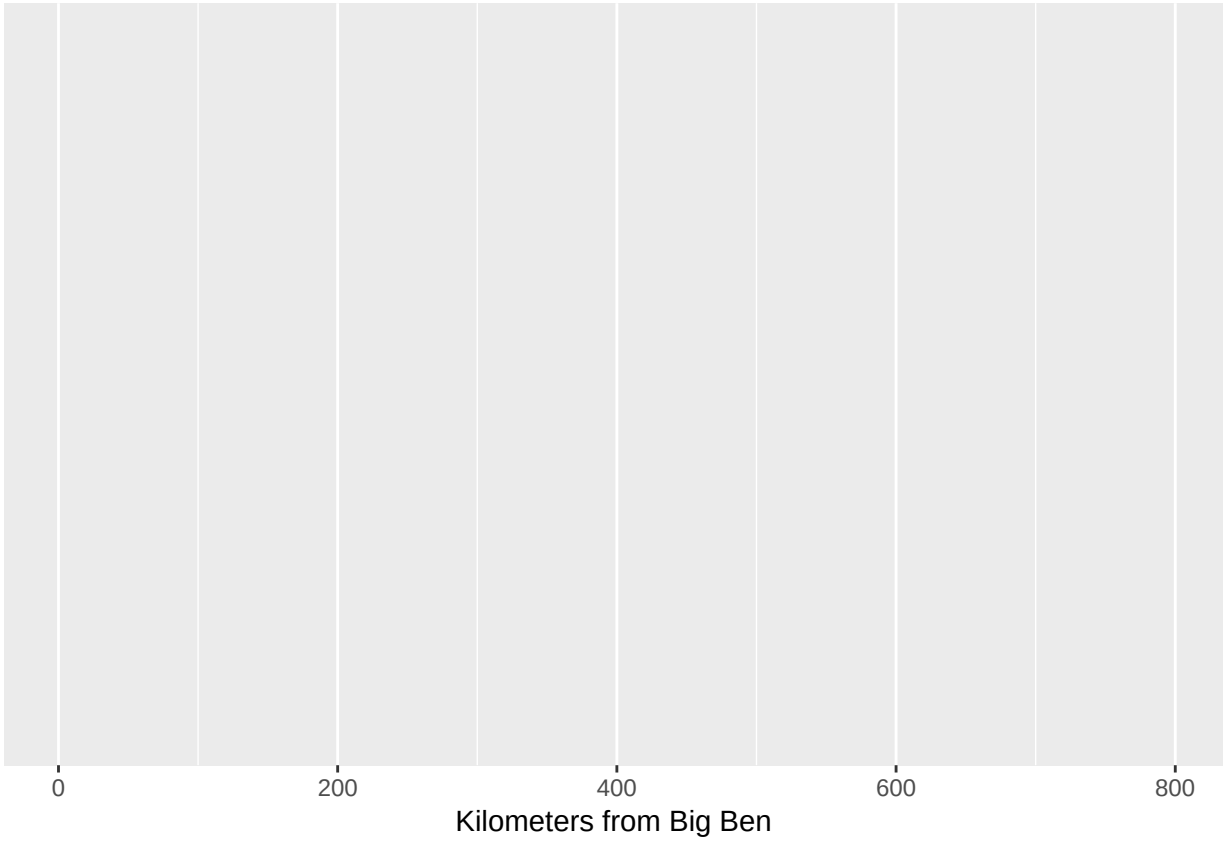
```
ggplot(data = dataLong) +
  aes(x = `Kilometers from Big Ben`) +
  aes(y = `Menu Item Price`) +
  facet_wrap(~ Item, scales = "free_y") +
  geom_point() +
```

```
  geom_smooth() +
  xlim(c(0, 10)) +
  labs(title = "Wetherspoon Pubs' Menu Item Prices v. Distance from Big Ben") +
  labs(subtitle = "Visualization: Gina Reynolds | Source: Financial Times Alphaville")
```

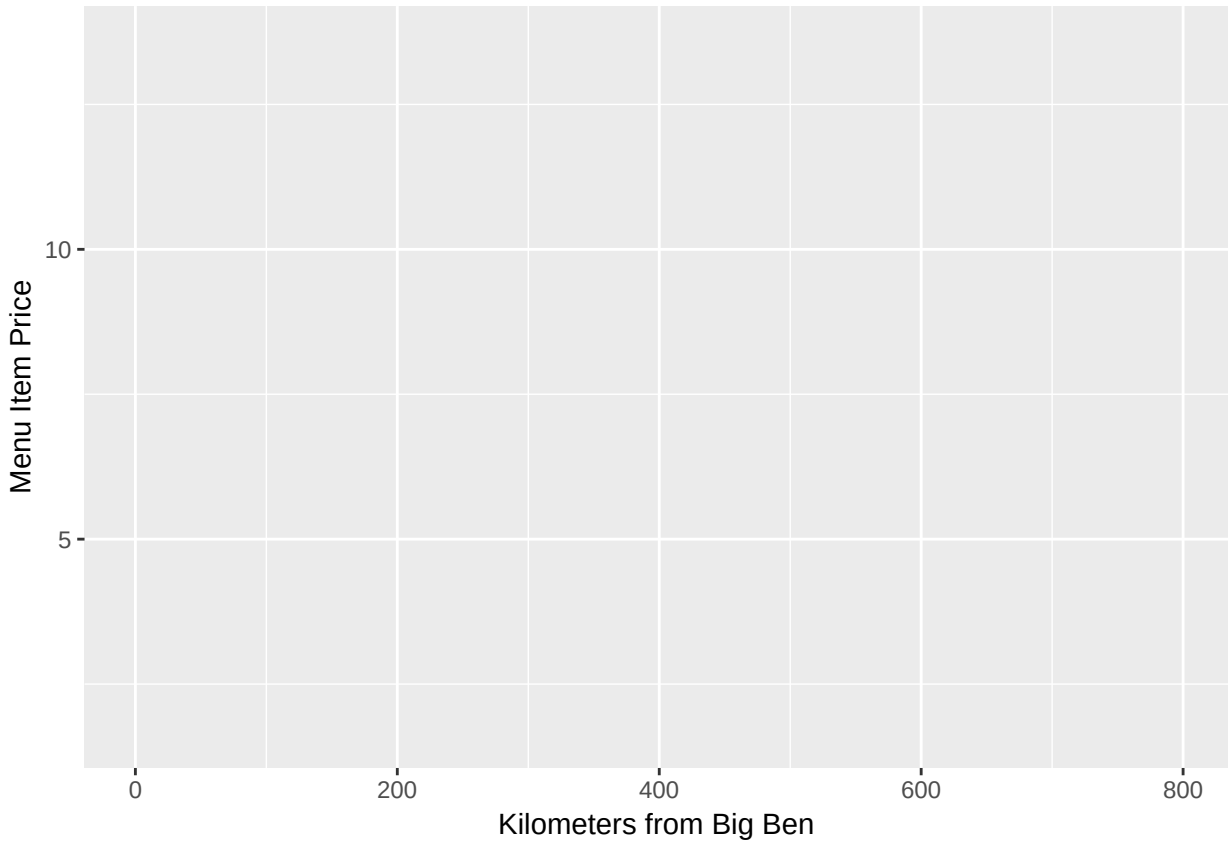


```
ggplot(data = dataLong)
```

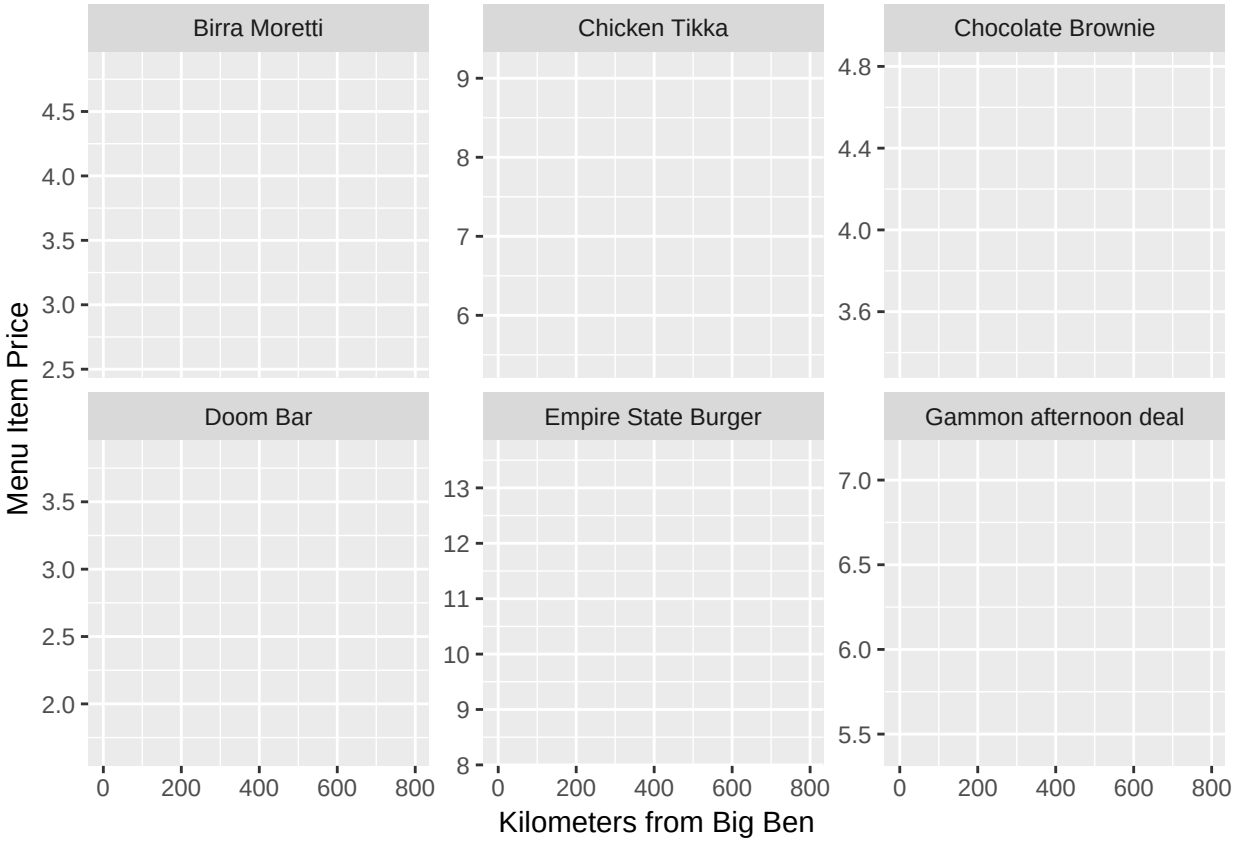
```
ggplot(data = dataLong) +
  aes(x = `Kilometers from Big Ben`)
```



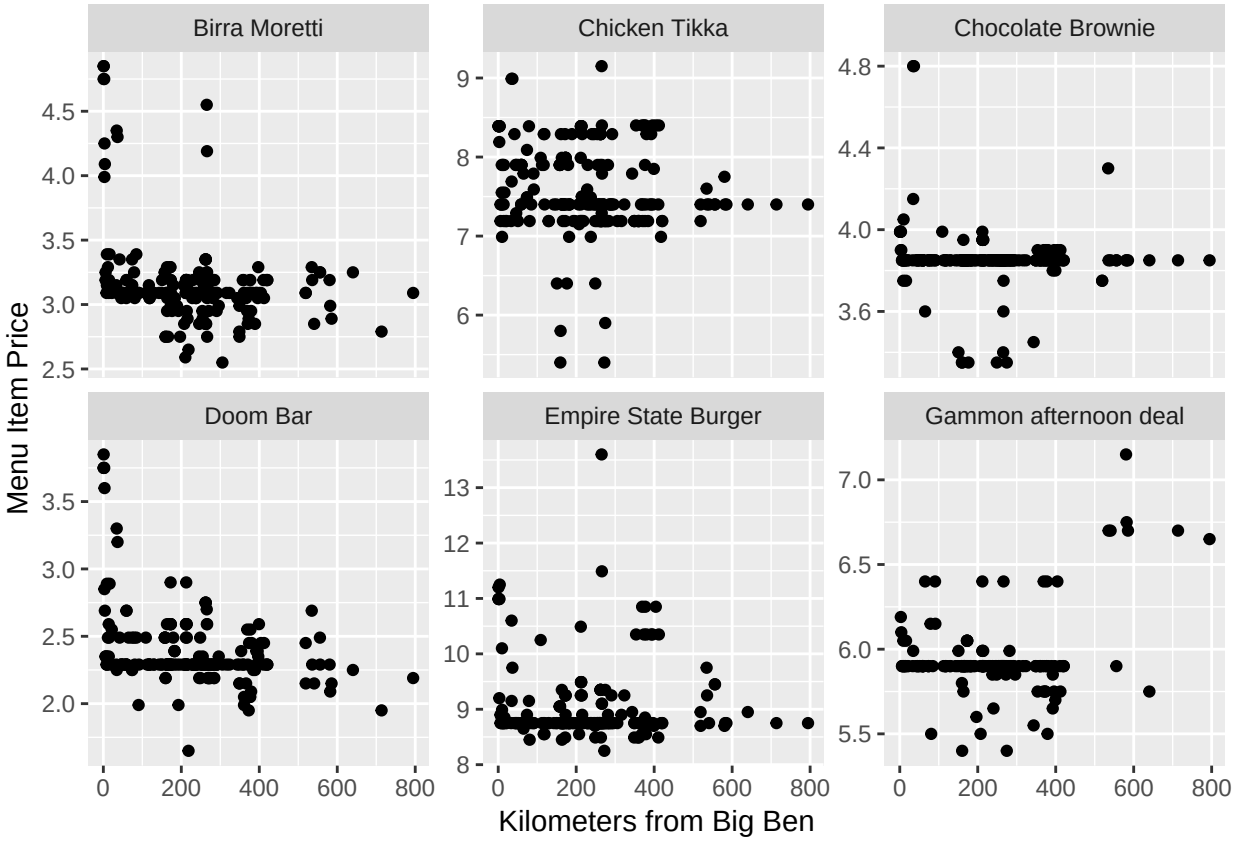
```
ggplot(data = dataLong) +
  aes(x = `Kilometers from Big Ben`) +
  aes(y = `Menu Item Price`)
```



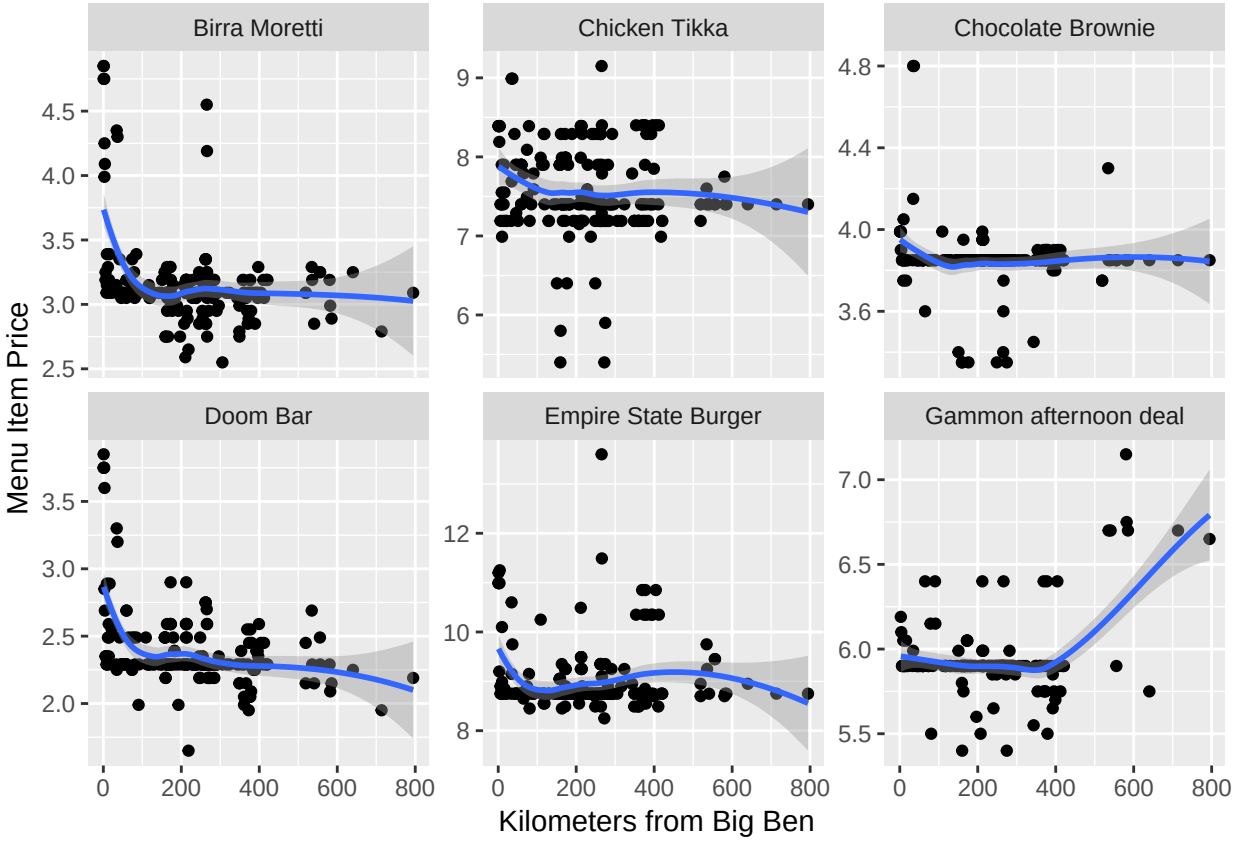
```
ggplot(data = dataLong) +
  aes(x = `Kilometers from Big Ben`) +
  aes(y = `Menu Item Price`) +
  facet_wrap(~ Item, scales = "free_y")
```



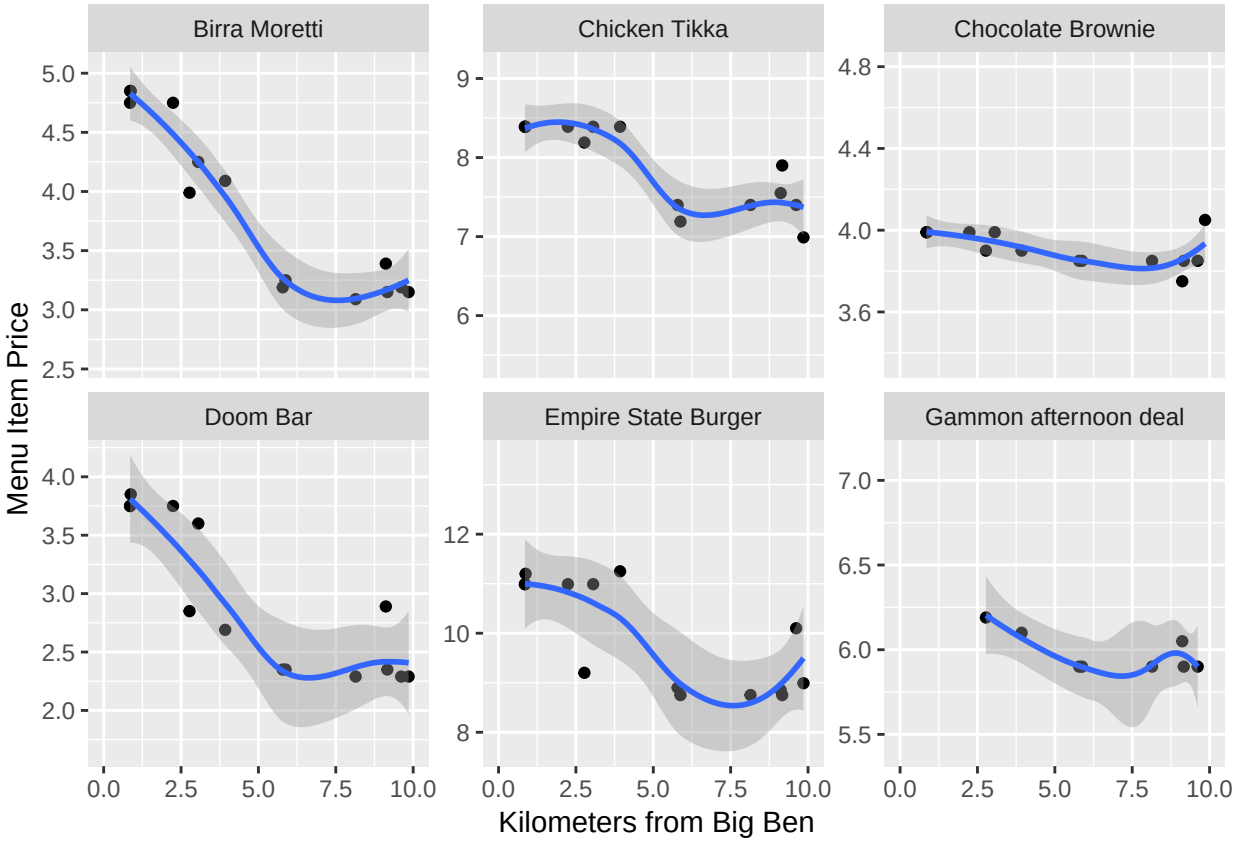
```
ggplot(data = dataLong) +
  aes(x = `Kilometers from Big Ben`) +
  aes(y = `Menu Item Price`) +
  facet_wrap(~ Item, scales = "free_y") +
  geom_point()
```



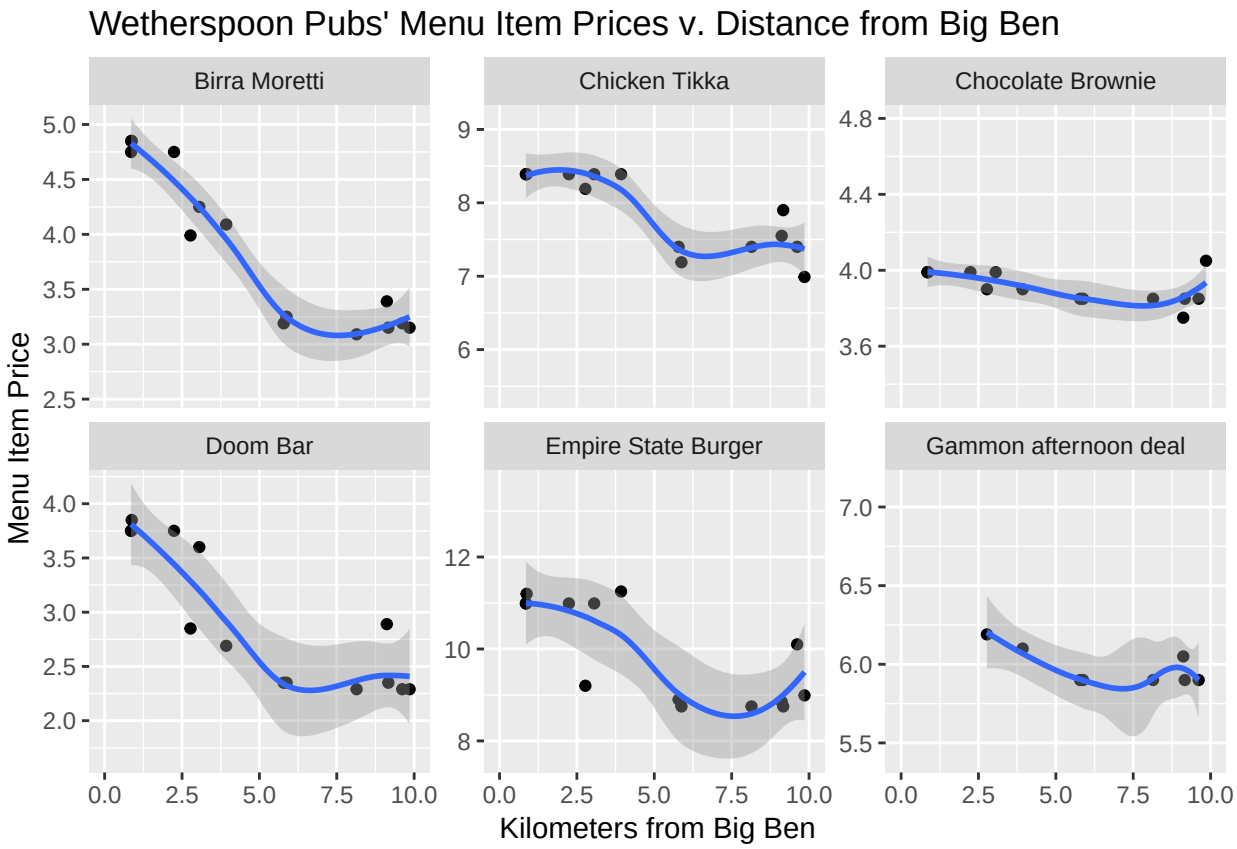
```
ggplot(data = dataLong) +
  aes(x = `Kilometers from Big Ben`) +
  aes(y = `Menu Item Price`) +
  facet_wrap(~ Item, scales = "free_y") +
  geom_point() +
  geom_smooth()
```



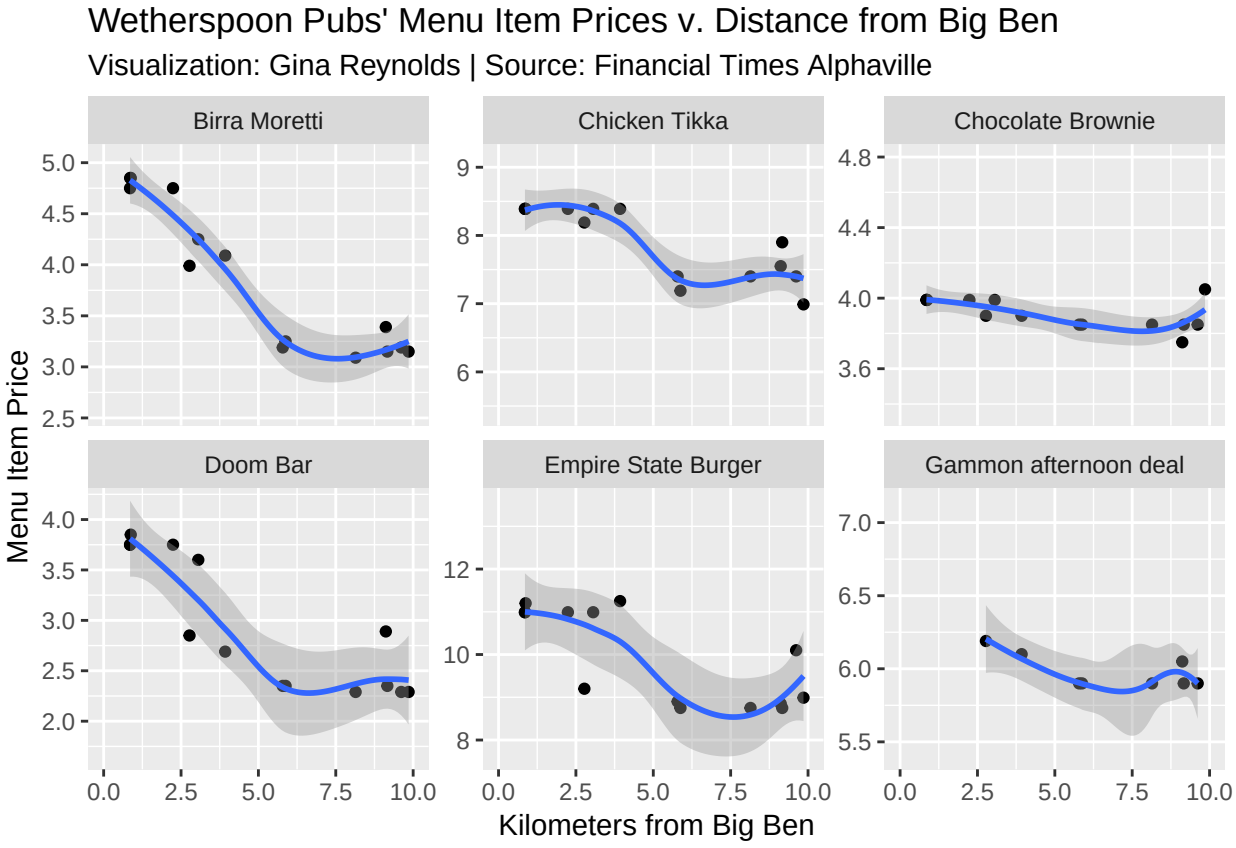
```
ggplot(data = dataLong) +
  aes(x = `Kilometers from Big Ben`) +
  aes(y = `Menu Item Price`) +
  facet_wrap(~ Item, scales = "free_y") +
  geom_point() +
  geom_smooth() +
  xlim(c(0, 10))
```



```
ggplot(data = dataLong) +
  aes(x = `Kilometers from Big Ben`) +
  aes(y = `Menu Item Price`) +
  facet_wrap(~ Item, scales = "free_y") +
  geom_point() +
  geom_smooth() +
  xlim(c(0, 10)) +
  labs(title = "Wetherspoon Pubs' Menu Item Prices v. Distance from Big Ben")
```



```
ggplot(data = dataLong) +
  aes(x = `Kilometers from Big Ben`) +
  aes(y = `Menu Item Price`) +
  facet_wrap(~ Item, scales = "free_y") +
  geom_point() +
  geom_smooth() +
  xlim(c(0, 10)) +
  labs(title = "Wetherspoon Pubs' Menu Item Prices v. Distance from Big Ben") +
  labs(subtitle = "Visualization: Gina Reynolds | Source: Financial Times Alphaville")
```



Chapter 11

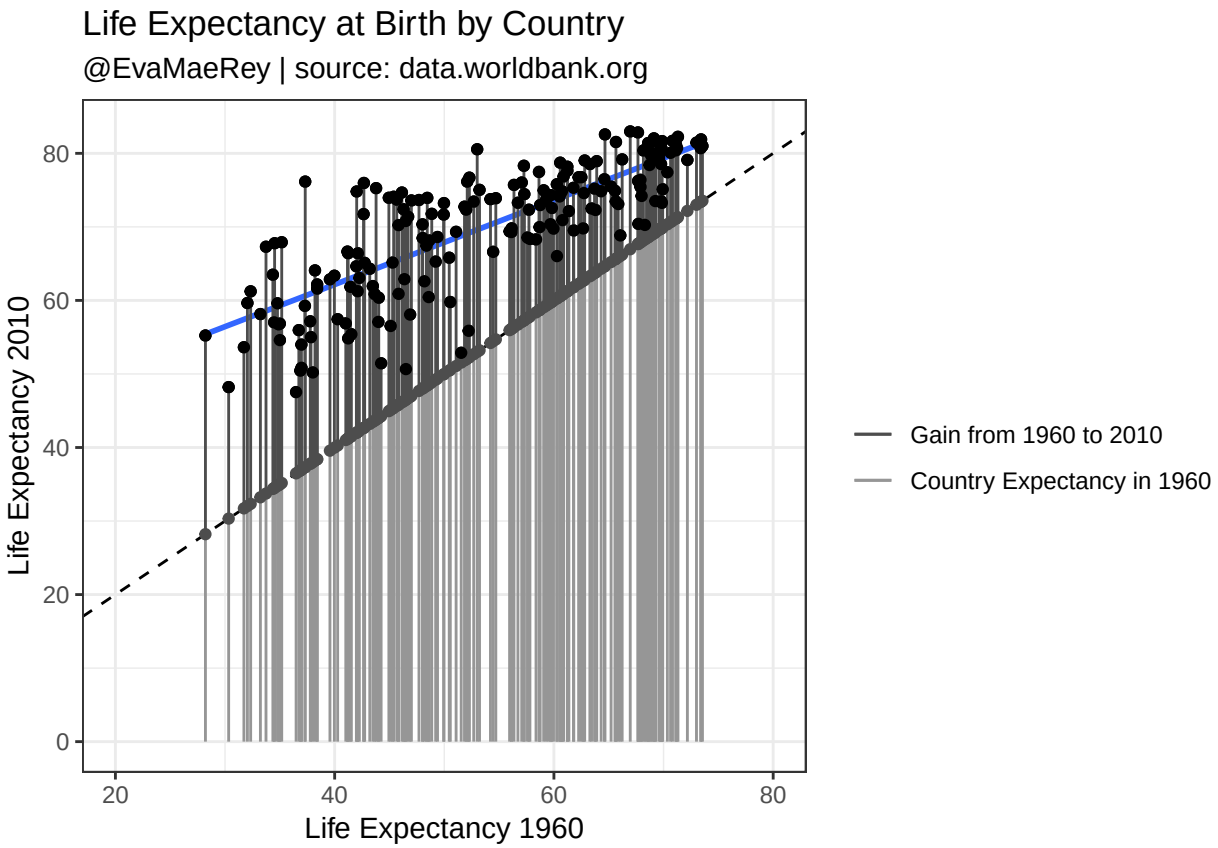
Life Expectancy Increases

To dramatically show the increases in life expectancy by country from 1960 to 2010, I plot the variable in 1960 versus itself in 2010. The line of equivilance (a 45% angle) is used as a reference and shows the result that you would see if there where no growth. The vertical distance from this line is the increase in life expectancy. I also superimpose a linear model on top of the scatter plot. You can see that the gains are greater for countries that started off with lower life expectancies.

A random sample from the data set:

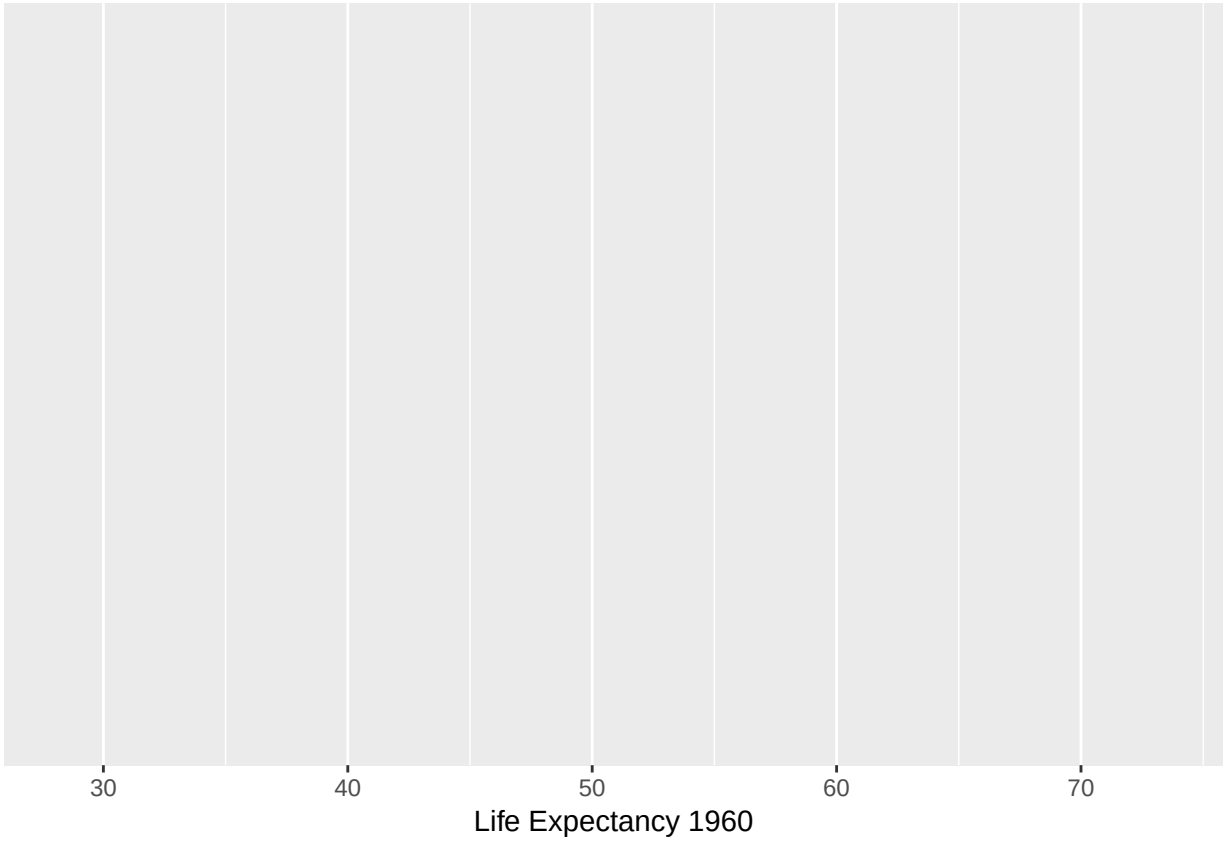
Life Expectancy 1960	Country Code	Country Name	Region	Income Group	Year	Life Ex
67.90290	EST	Estonia	Europe & Central Asia	High income	2010	
52.22624	ZAF	South Africa	Sub-Saharan Africa	Upper middle income	2010	
56.15354	TJK	Tajikistan	Europe & Central Asia	Lower middle income	2010	
37.31376	MDV	Maldives	South Asia	Upper middle income	2010	
63.43185	GEO	Georgia	Europe & Central Asia	Lower middle income	2010	

```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010")) +
  geom_segment(mapping = aes(y = 0, col = "Country Expectancy in 1960")) +
  scale_color_manual( breaks = c("Gain from 1960 to 2010",  "Country Expectancy in 1960"), values = c("g
  geom_point(aes(y = `Life Expectancy 1960`), col = "grey30") +
  geom_point() +
  labs(subtitle = "@EvaMaeRey | source: data.worldbank.org", size = .7) +
  labs(title = "Life Expectancy at Birth by Country") +
  labs(col = "") +
  theme(legend.title = element_blank()) +
  theme_bw() +
  xlim(c(20, 80))
```

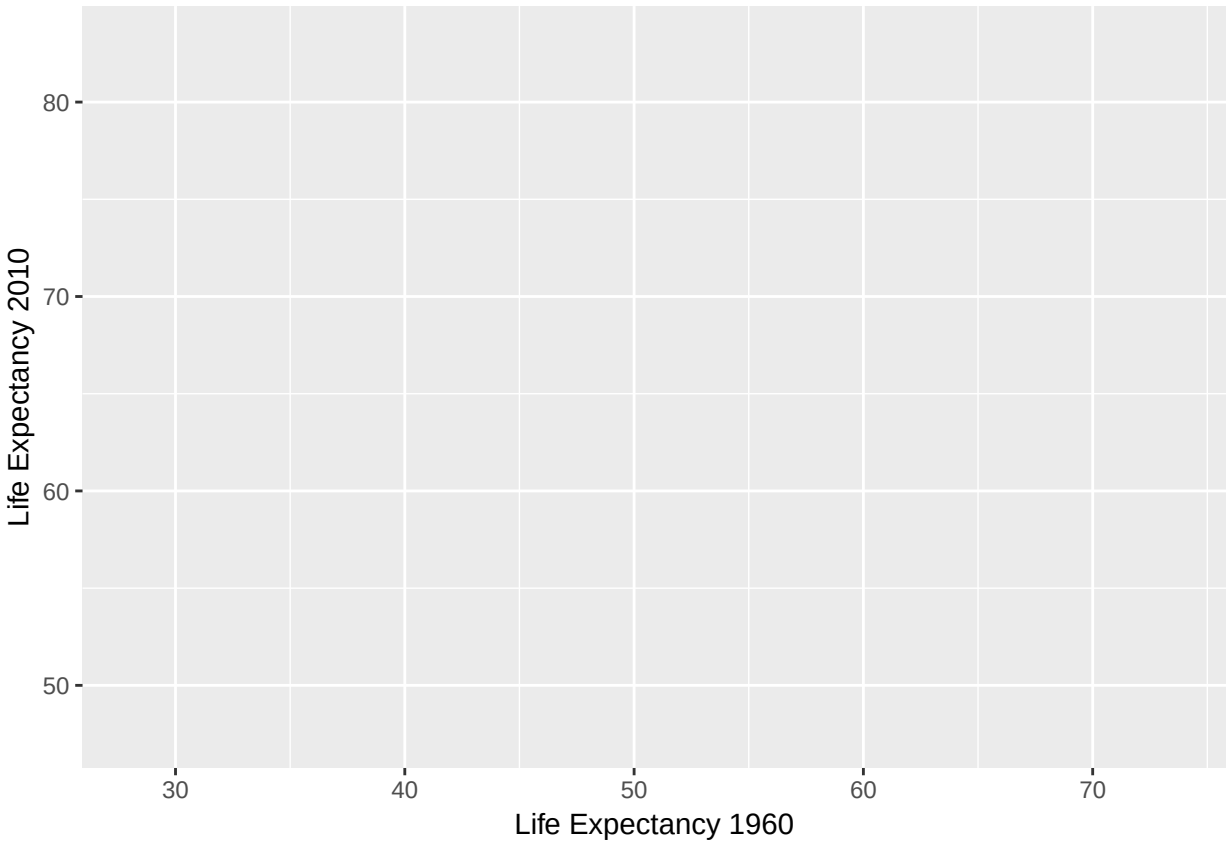

Life Expectancy at Birth by Country
@EvaMaeRey | source: data.worldbank.org
Life Expectancy 2010
Life Expectancy 1960
0
20
40
60
80
20
40
60
80
Gain from 1960 to 2010
Country Expectancy in 1960

```
ggplot(compare)
```

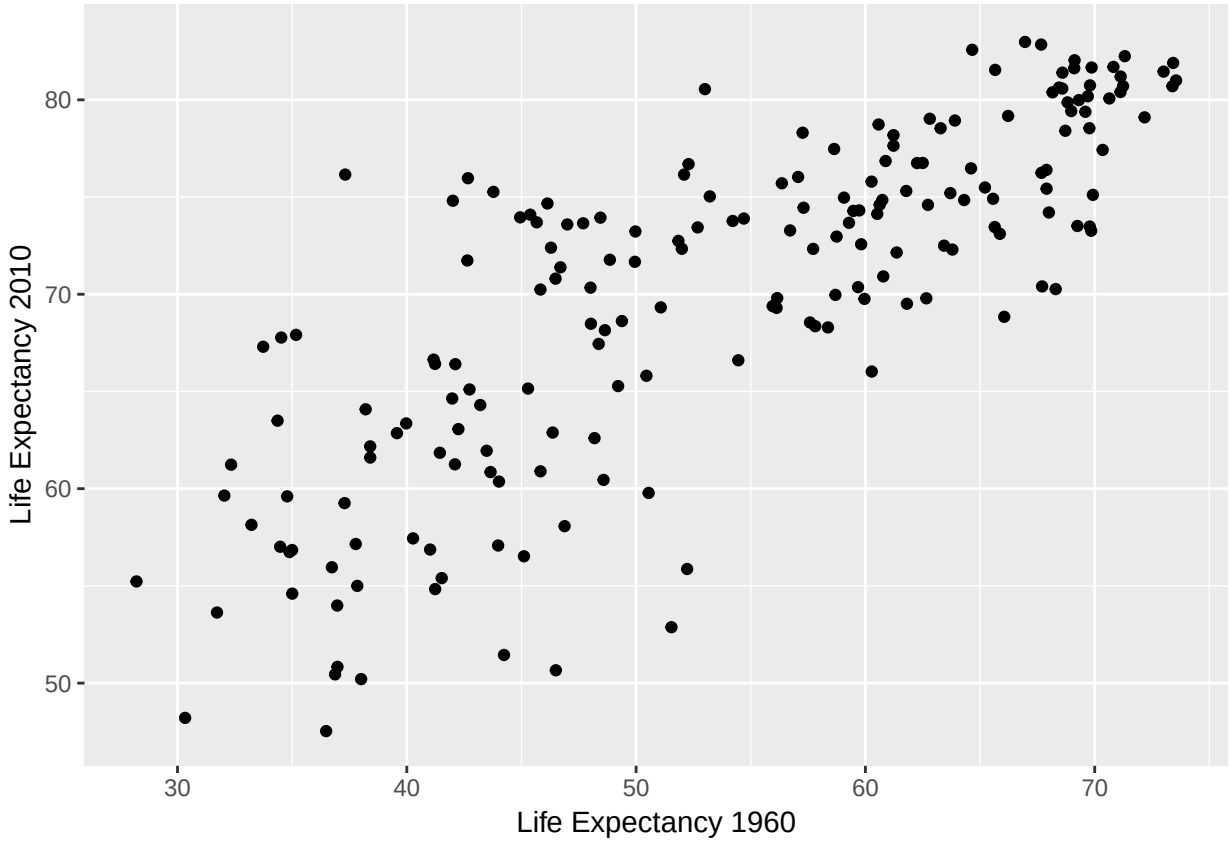
```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`)
```



```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`)
```

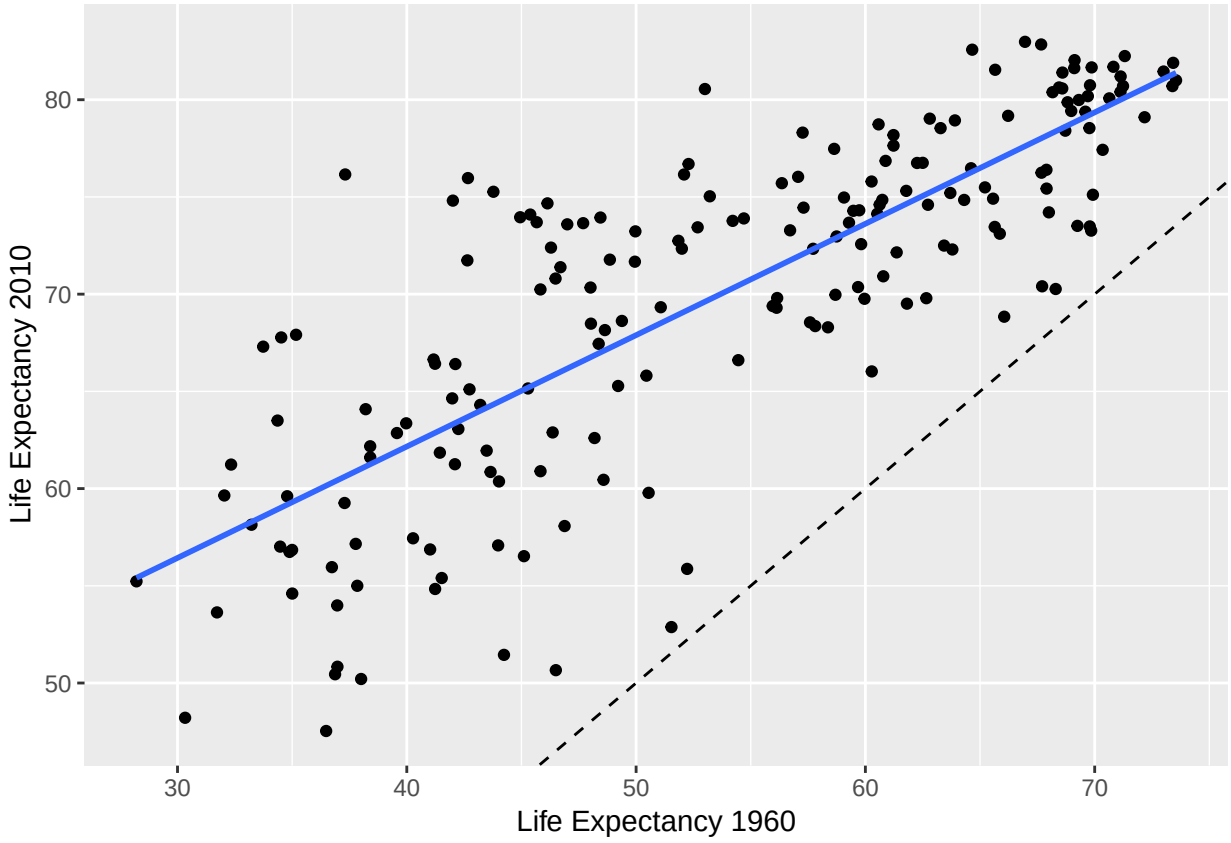


```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point()
```

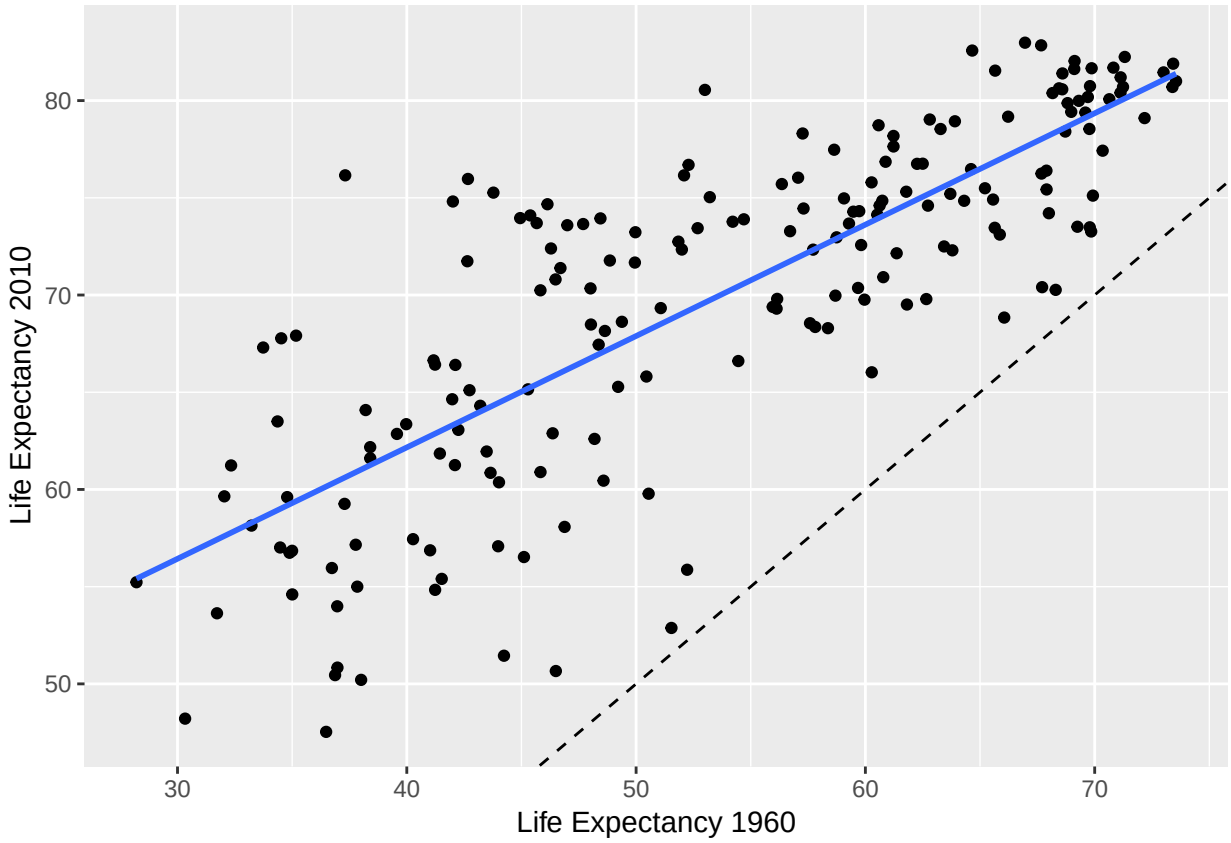


```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm")
```

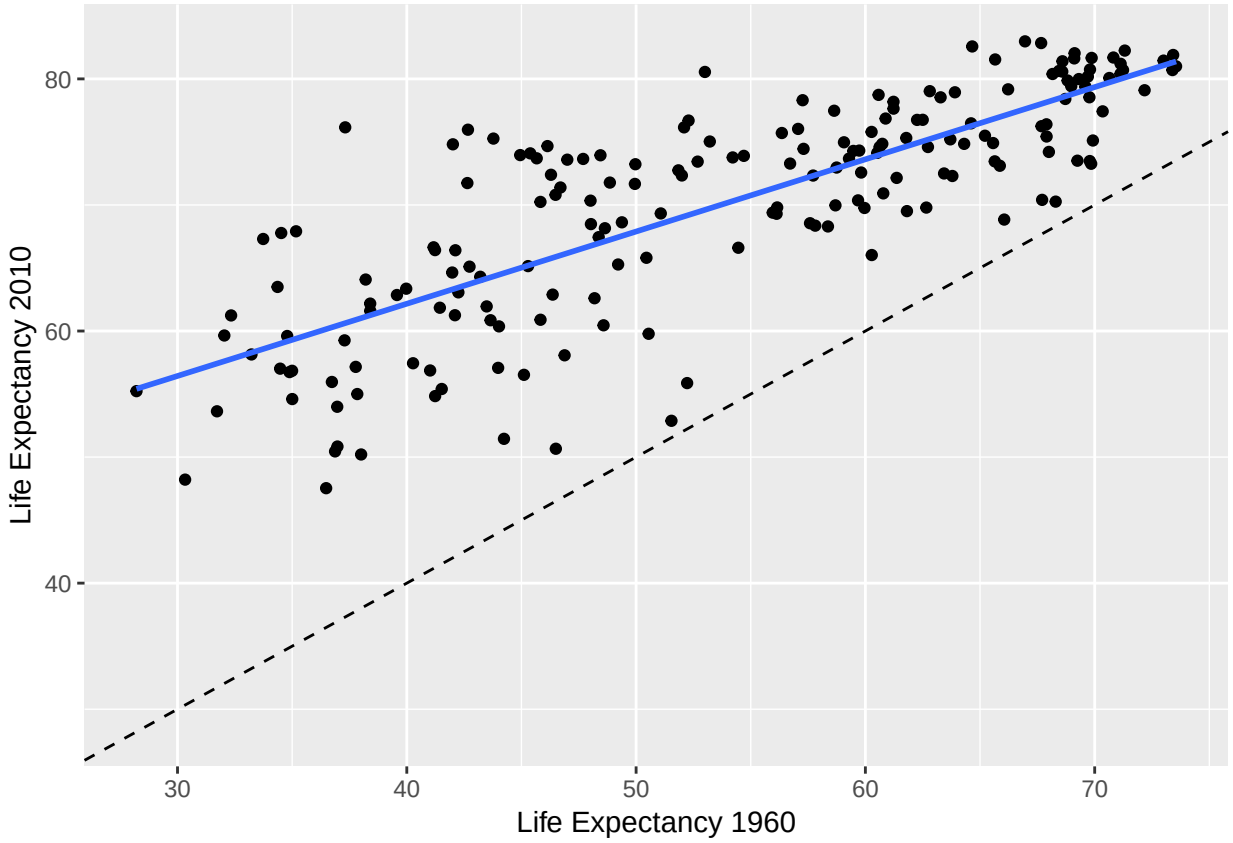
```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2)
```



```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`)
```

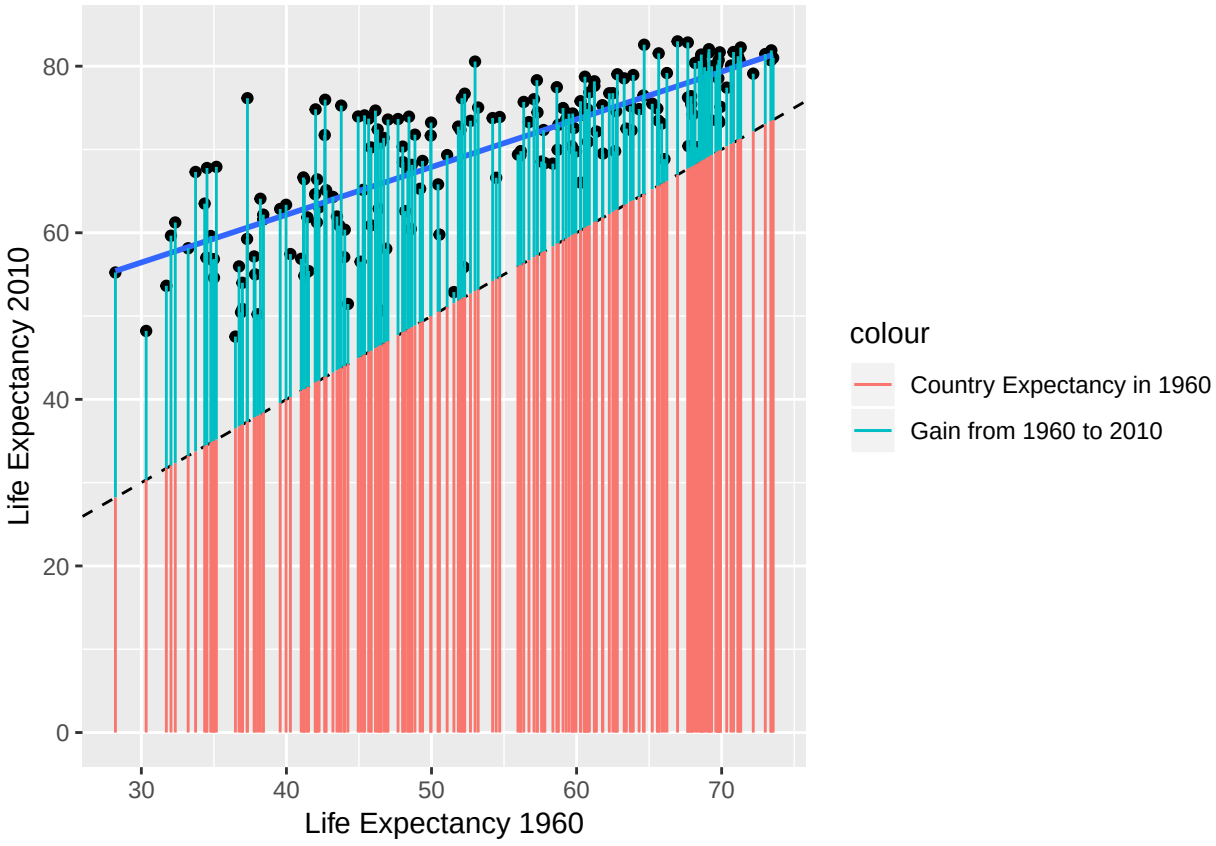


```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`)
```

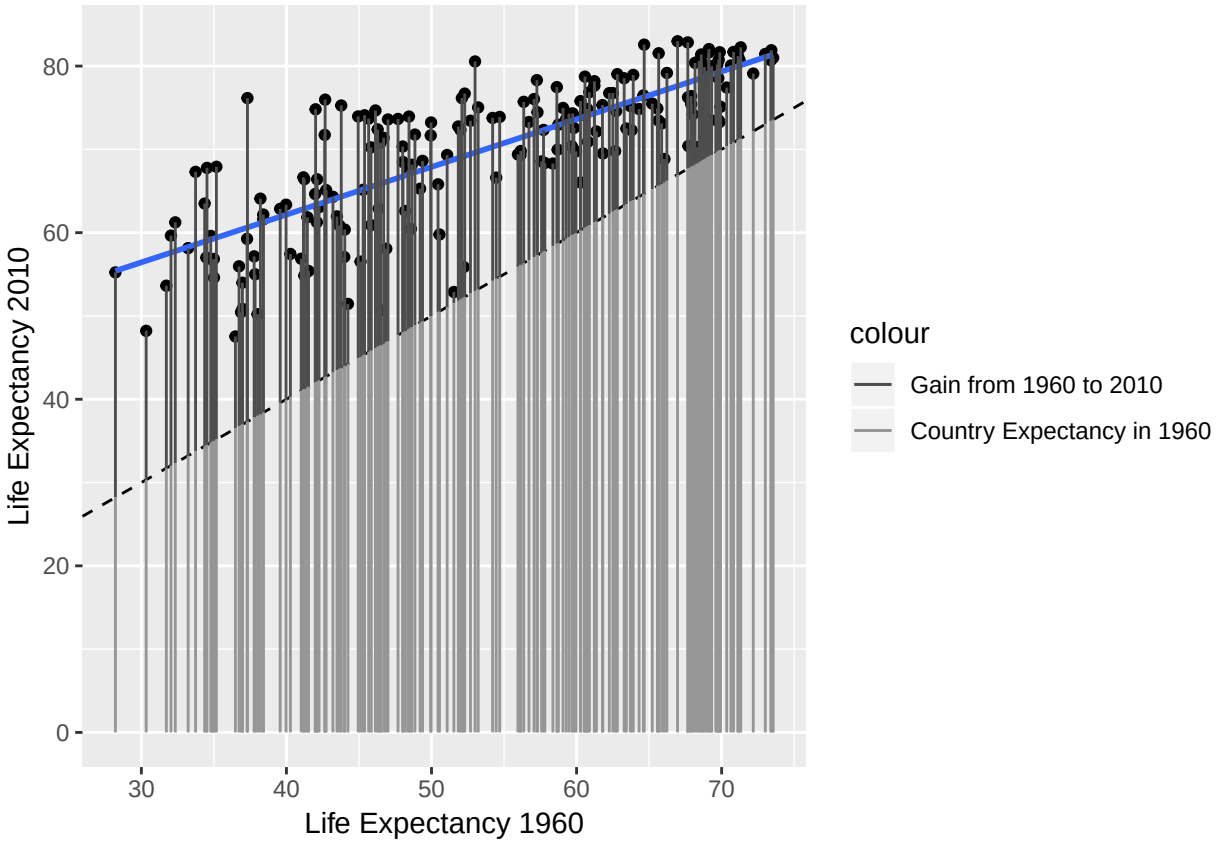


```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010"))
```

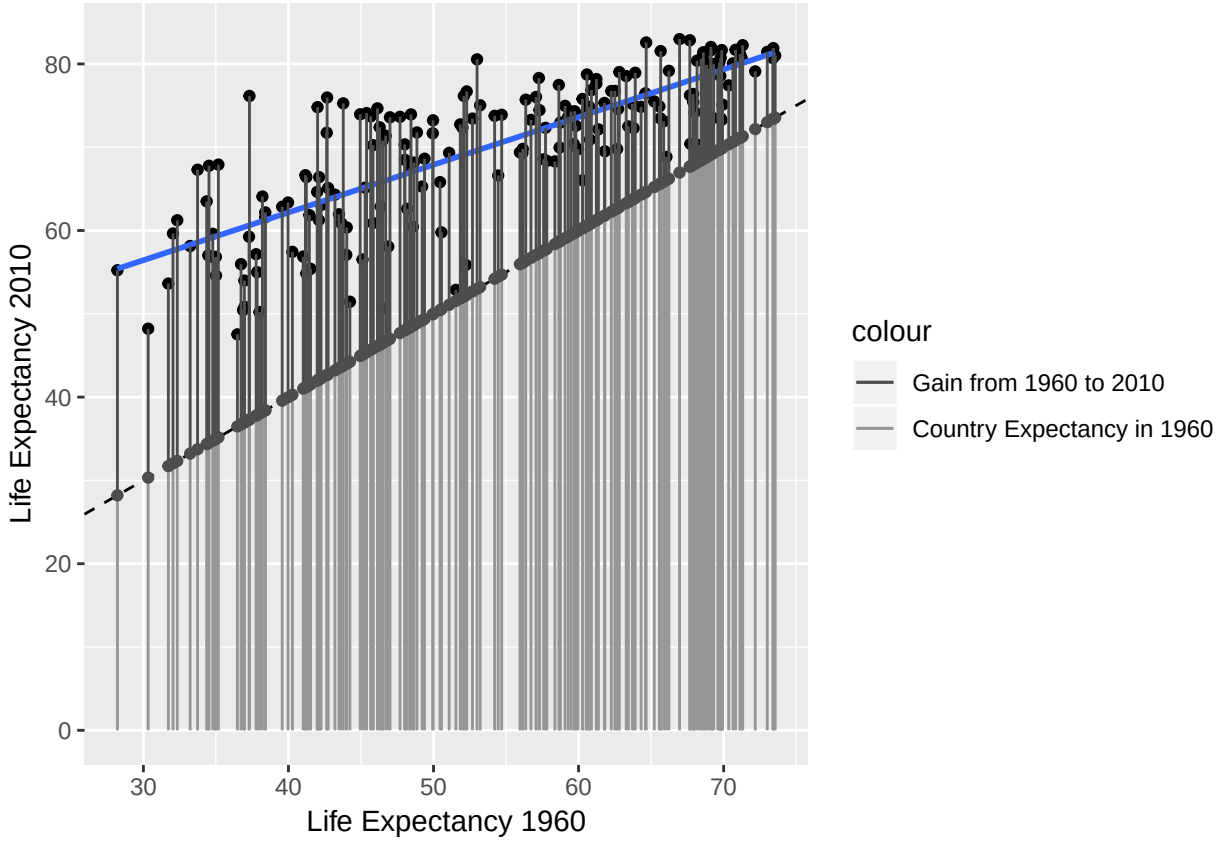
```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010")) +
  geom_segment(mapping = aes(y = 0, col = "Country Expectancy in 1960"))
```



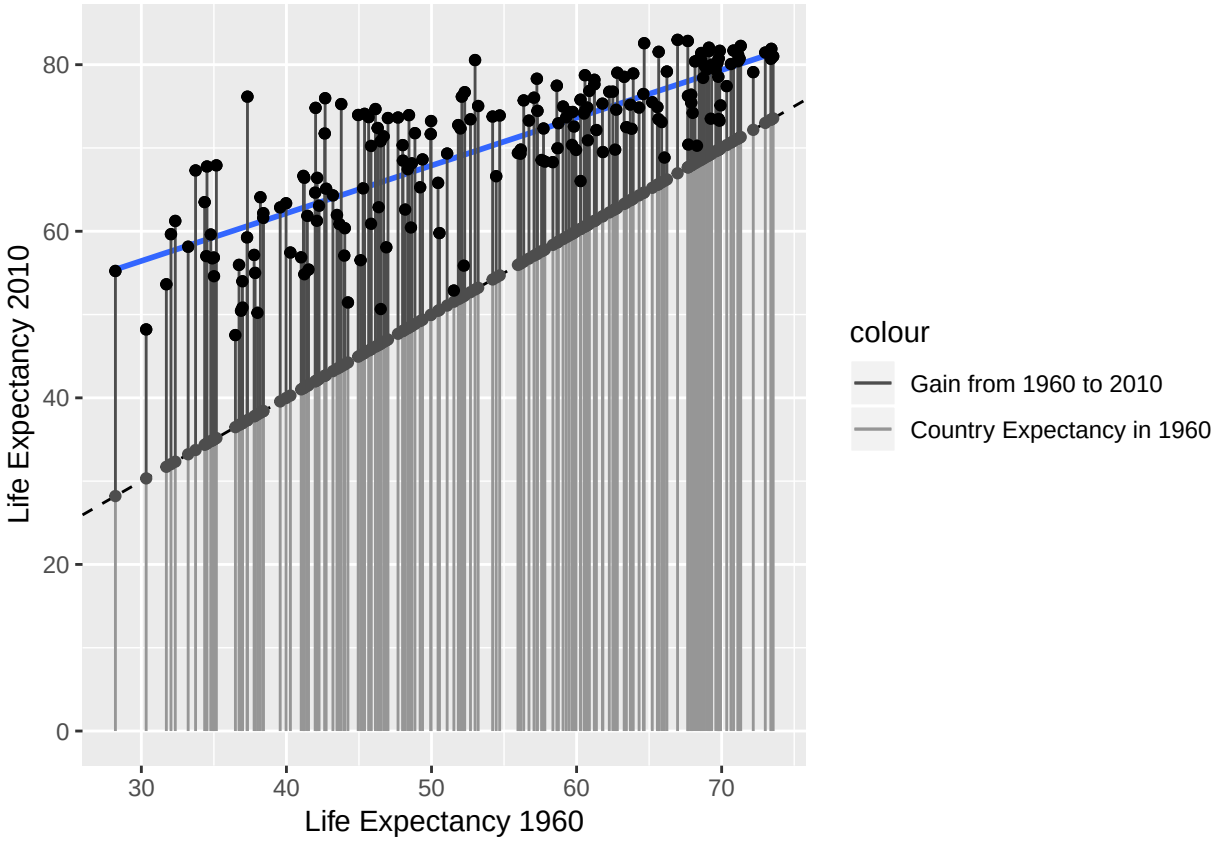
```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010")) +
  geom_segment(mapping = aes(y = 0, col = "Country Expectancy in 1960")) +
  scale_color_manual( breaks = c("Gain from 1960 to 2010",  "Country Expectancy in 1960"), values = c("
```



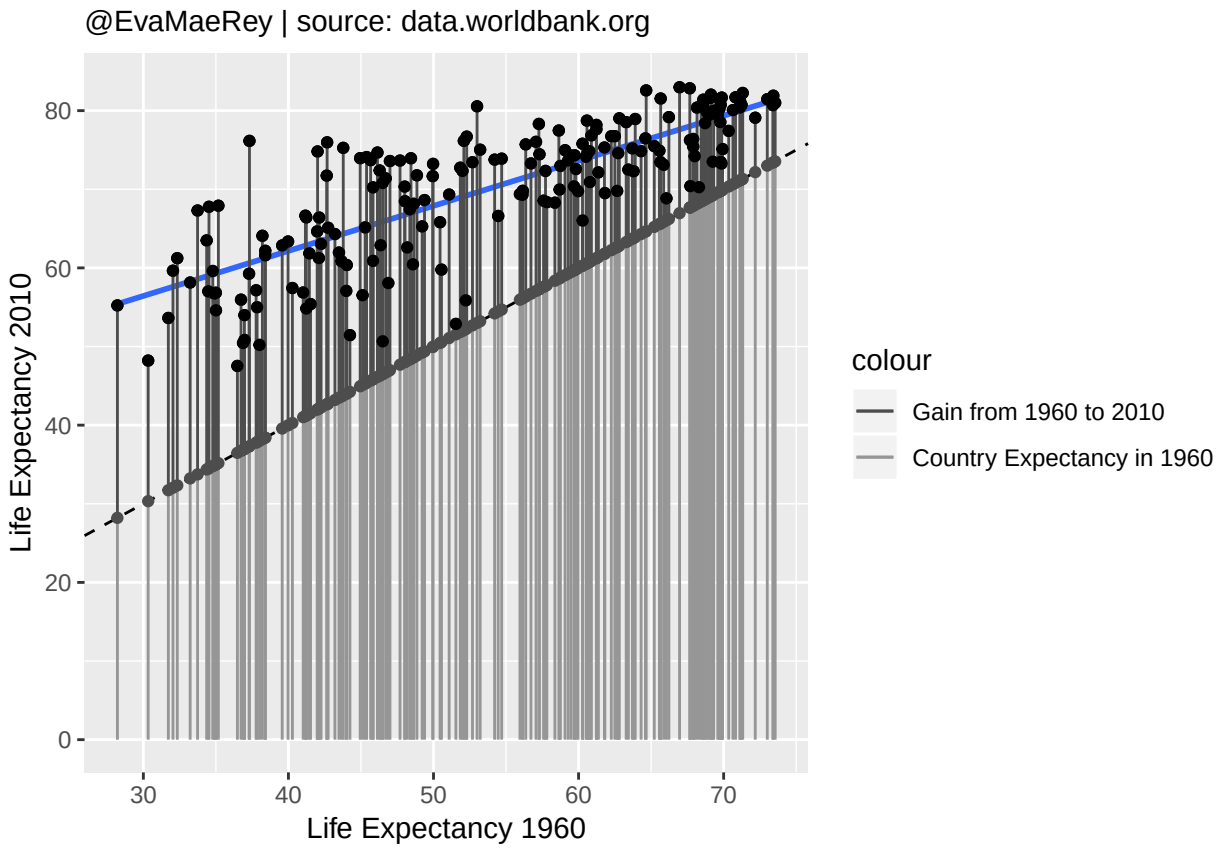
```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010")) +
  geom_segment(mapping = aes(y = 0, col = "Country Expectancy in 1960")) +
  scale_color_manual( breaks = c("Gain from 1960 to 2010",  "Country Expectancy in 1960"), values = c("
  geom_point(aes(y = `Life Expectancy 1960`), col = "grey30")
```



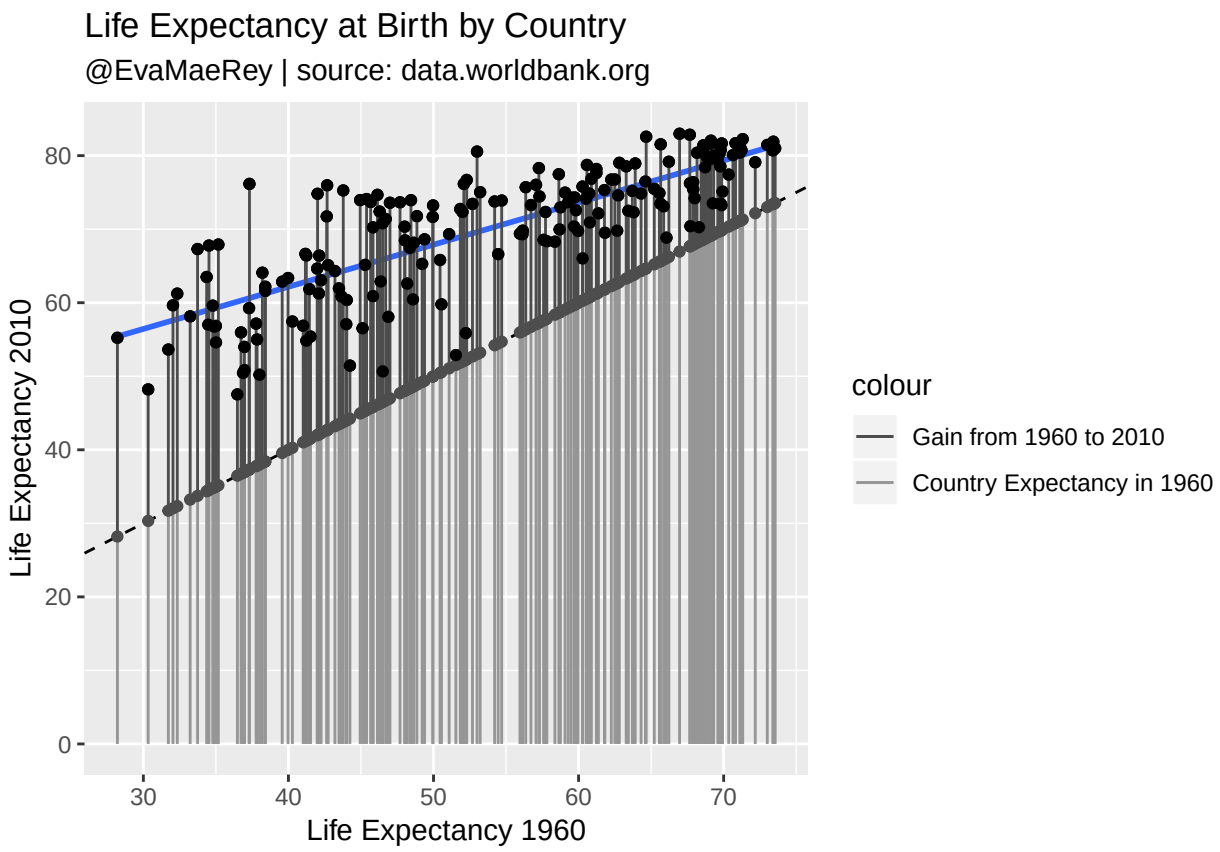
```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010")) +
  geom_segment(mapping = aes(y = 0, col = "Country Expectancy in 1960")) +
  scale_color_manual( breaks = c("Gain from 1960 to 2010",  "Country Expectancy in 1960"), values = c("
  geom_point(aes(y = `Life Expectancy 1960`), col = "grey30") +
  geom_point()
```



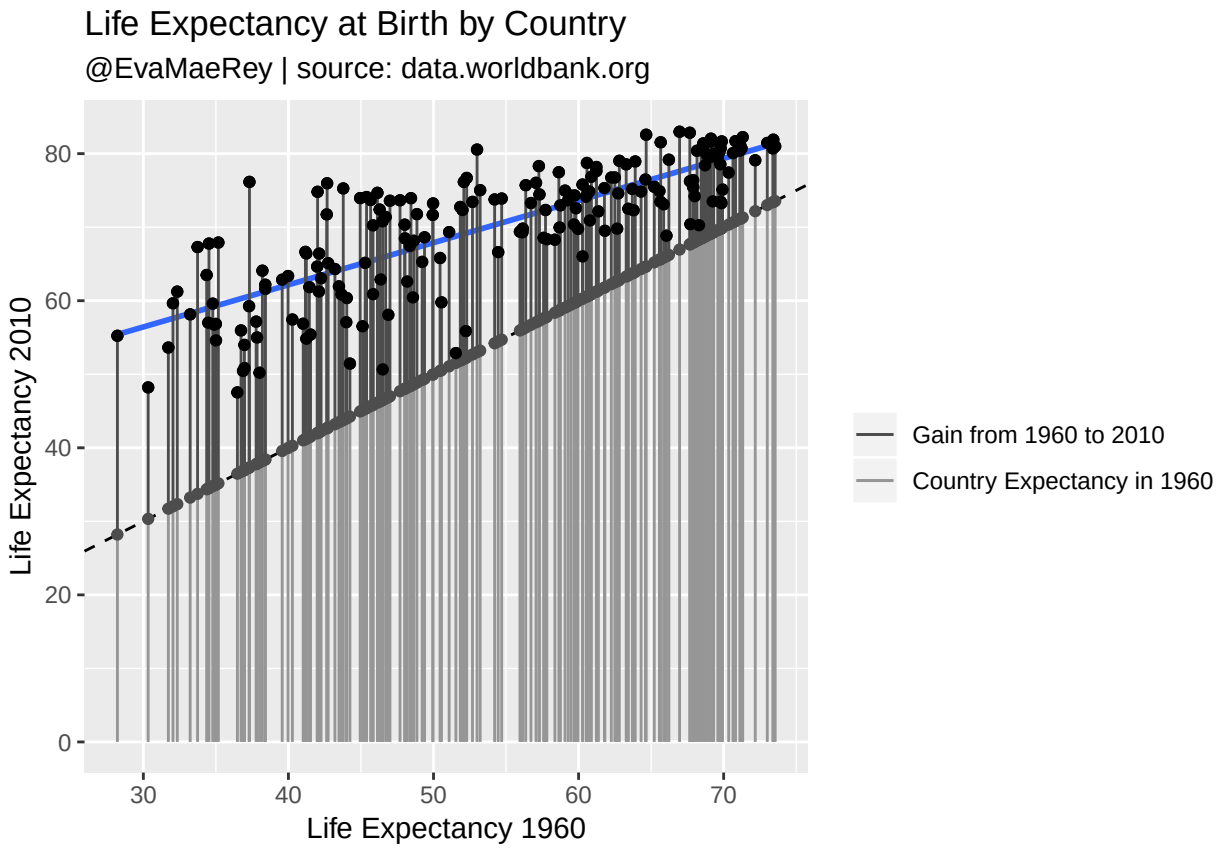
```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010")) +
  geom_segment(mapping = aes(y = 0, col = "Country Expectancy in 1960")) +
  scale_color_manual( breaks = c("Gain from 1960 to 2010",  "Country Expectancy in 1960"), values = c("
  geom_point(aes(y = `Life Expectancy 1960`), col = "grey30") +
  geom_point() +
  labs(subtitle = "@EvaMaeRey | source: data.worldbank.org", size = .7)
```



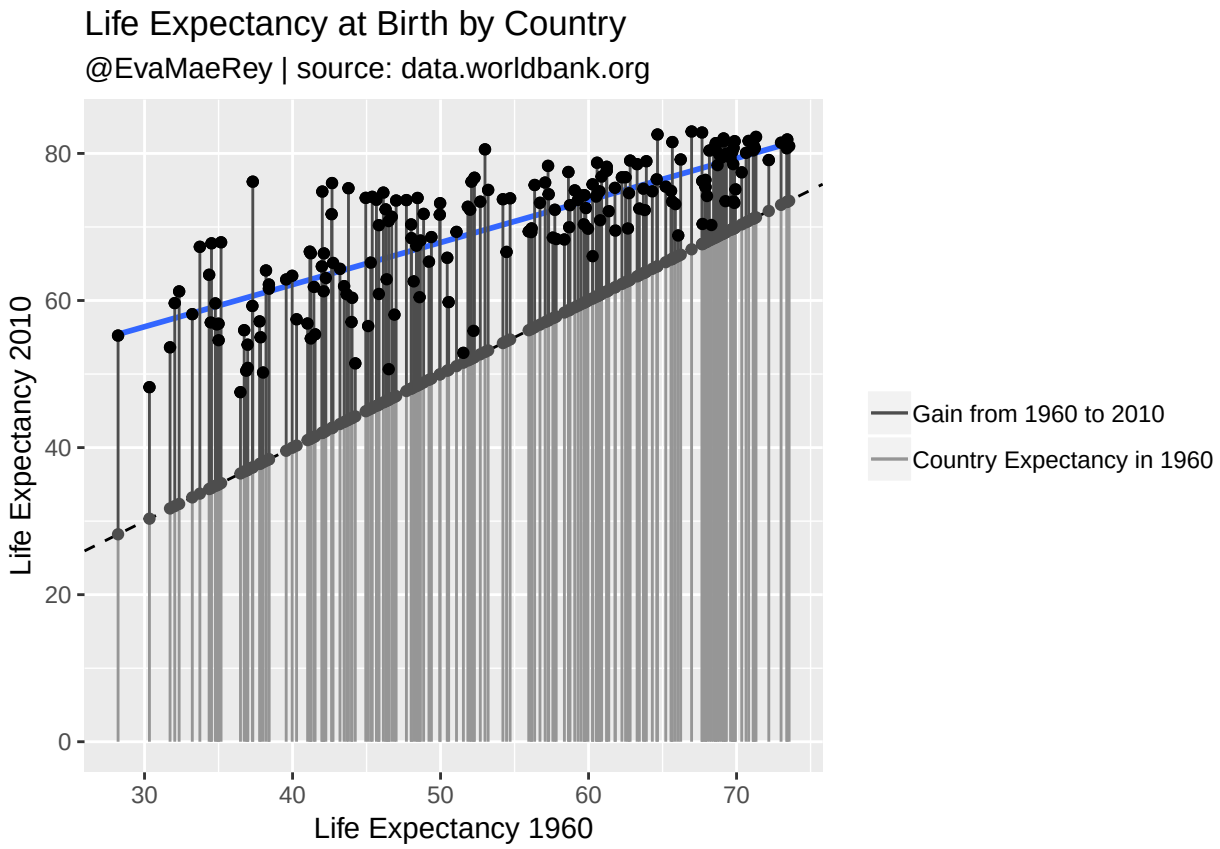
```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010")) +
  geom_segment(mapping = aes(y = 0, col = "Country Expectancy in 1960")) +
  scale_color_manual( breaks = c("Gain from 1960 to 2010",  "Country Expectancy in 1960"), values = c("
  geom_point(aes(y = `Life Expectancy 1960`), col = "grey30") +
  geom_point() +
  labs(subtitle = "@EvaMaeRey | source: data.worldbank.org", size = .7) +
  labs(title = "Life Expectancy at Birth by Country")
```



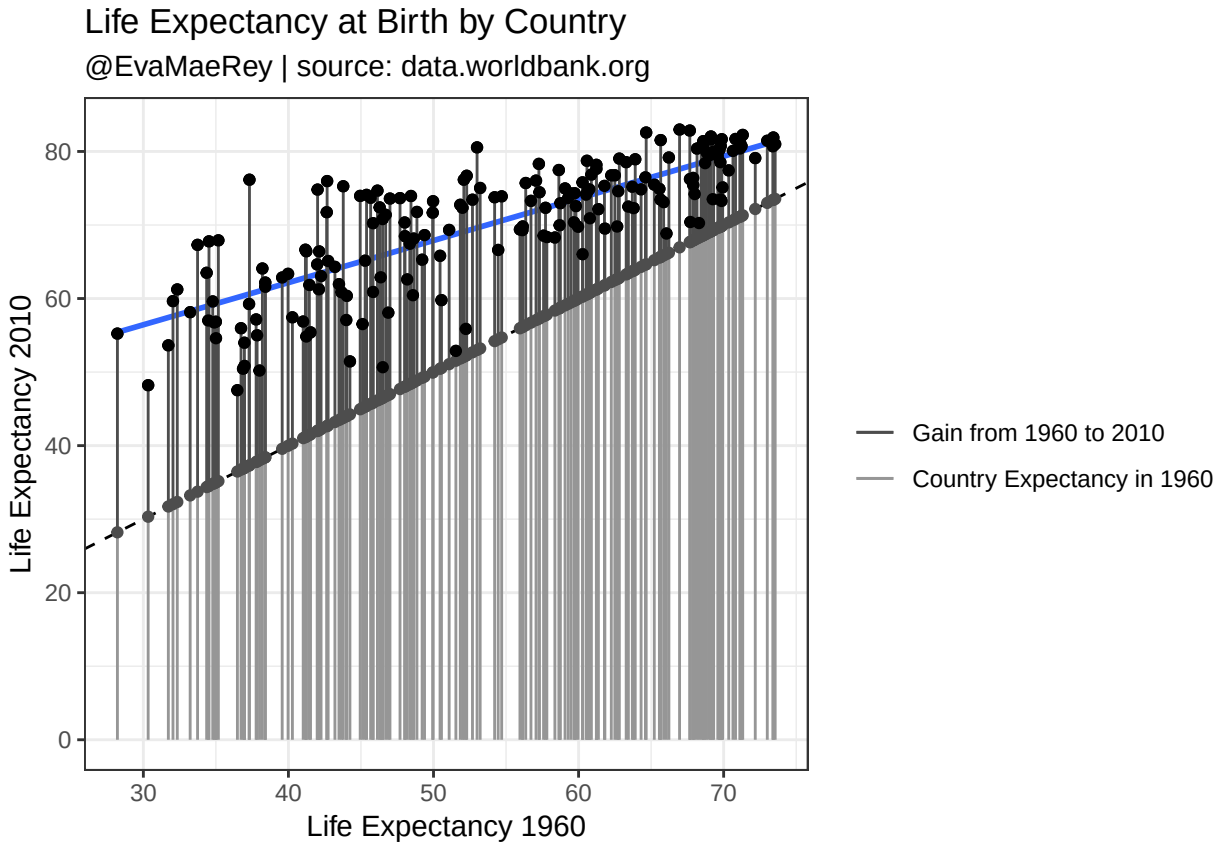
```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010")) +
  geom_segment(mapping = aes(y = 0, col = "Country Expectancy in 1960")) +
  scale_color_manual( breaks = c("Gain from 1960 to 2010",  "Country Expectancy in 1960"), values = c("
  geom_point(aes(y = `Life Expectancy 1960`), col = "grey30") +
  geom_point() +
  labs(subtitle = "@EvaMaeRey | source: data.worldbank.org", size = .7) +
  labs(title = "Life Expectancy at Birth by Country") +
  labs(col = "")
```



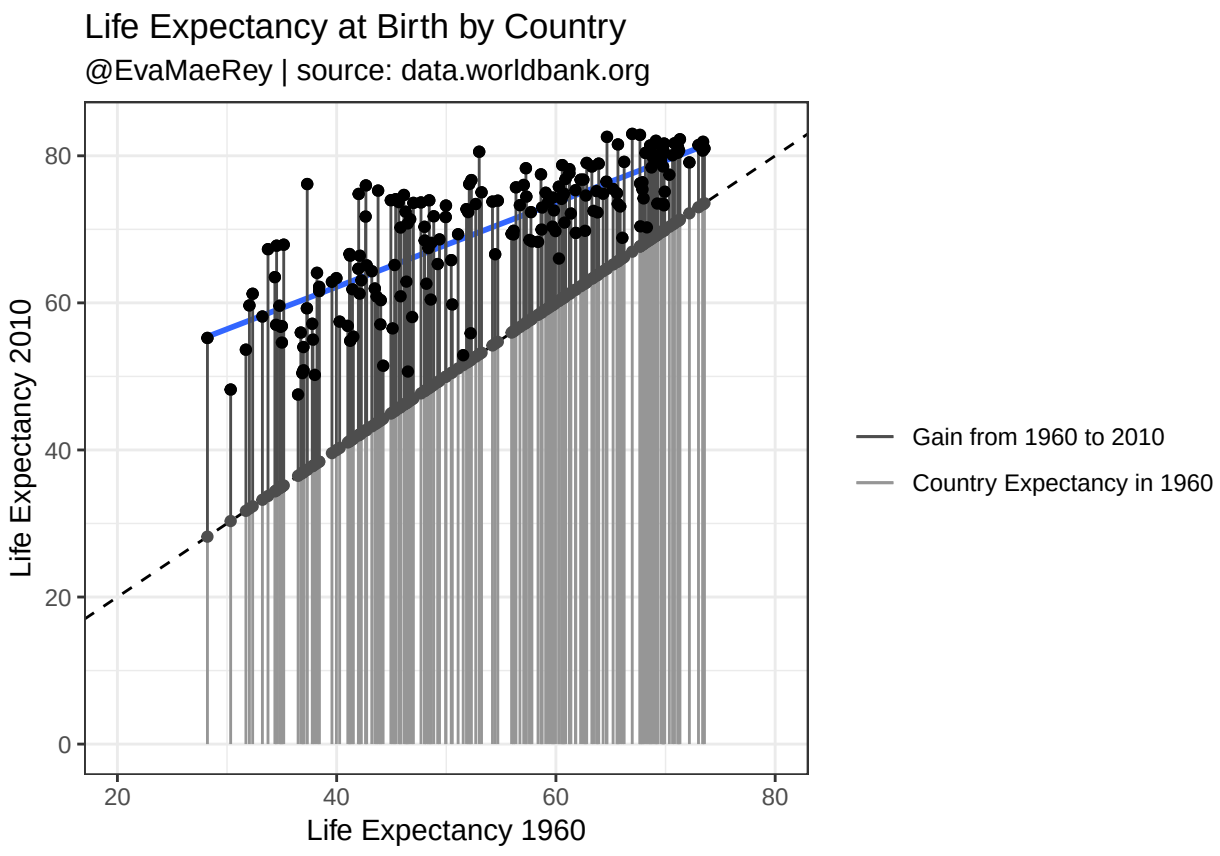
```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010")) +
  geom_segment(mapping = aes(y = 0, col = "Country Expectancy in 1960")) +
  scale_color_manual( breaks = c("Gain from 1960 to 2010",  "Country Expectancy in 1960"), values = c("
  geom_point(aes(y = `Life Expectancy 1960`), col = "grey30") +
  geom_point() +
  labs(subtitle = "@EvaMaeRey | source: data.worldbank.org", size = .7) +
  labs(title = "Life Expectancy at Birth by Country") +
  labs(col = "") +
  theme(legend.title = element_blank())
```



```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010")) +
  geom_segment(mapping = aes(y = 0, col = "Country Expectancy in 1960")) +
  scale_color_manual( breaks = c("Gain from 1960 to 2010",  "Country Expectancy in 1960"), values = c("
  geom_point(aes(y = `Life Expectancy 1960`), col = "grey30") +
  geom_point() +
  labs(subtitle = "@EvaMaeRey | source: data.worldbank.org", size = .7) +
  labs(title = "Life Expectancy at Birth by Country") +
  labs(col = "") +
  theme(legend.title = element_blank()) +
  theme_bw()
```



```
ggplot(compare) +
  aes(x = `Life Expectancy 1960`) +
  aes(y = `Life Expectancy 2010`) +
  geom_point() +
  geom_smooth(se = F, method = "lm") +
  geom_abline(slope = 1, intercept = 0, lty = 2) +
  aes(xend = `Life Expectancy 1960`) +
  aes(yend = `Life Expectancy 1960`) +
  geom_segment(mapping = aes(col = "Gain from 1960 to 2010")) +
  geom_segment(mapping = aes(y = 0, col = "Country Expectancy in 1960")) +
  scale_color_manual( breaks = c("Gain from 1960 to 2010",  "Country Expectancy in 1960"), values = c("
  geom_point(aes(y = `Life Expectancy 1960`), col = "grey30") +
  geom_point() +
  labs(subtitle = "@EvaMaeRey | source: data.worldbank.org", size = .7) +
  labs(title = "Life Expectancy at Birth by Country") +
  labs(col = "") +
  theme(legend.title = element_blank()) +
  theme_bw() +
  xlim(c(20, 80))
```



Chapter 12

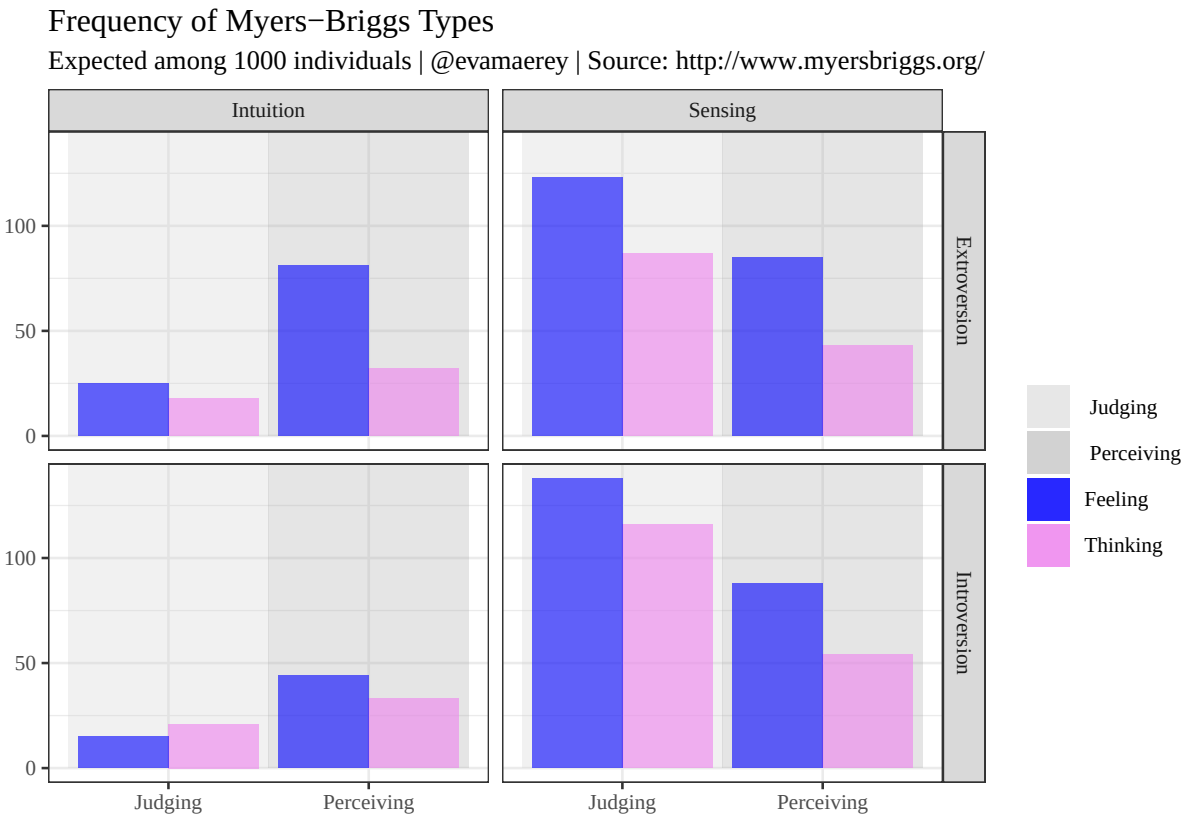
Myers Briggs

This data looks at the relationship betweeen four binary variables. The challenge is how to display that in one visualization. My first idea was to use a mosaic plot. However, I came across advice from "The Perceptual Edge", that generally advised against the use of the mosaic plot, instead favoring a kind of nested bar plot. I tried to implement that. While I do think that it is pretty, I think that it still requires a lot of the reader to interpret the graph. Perhaps more annotation could alleviate this burden.

A random sample from the data set:

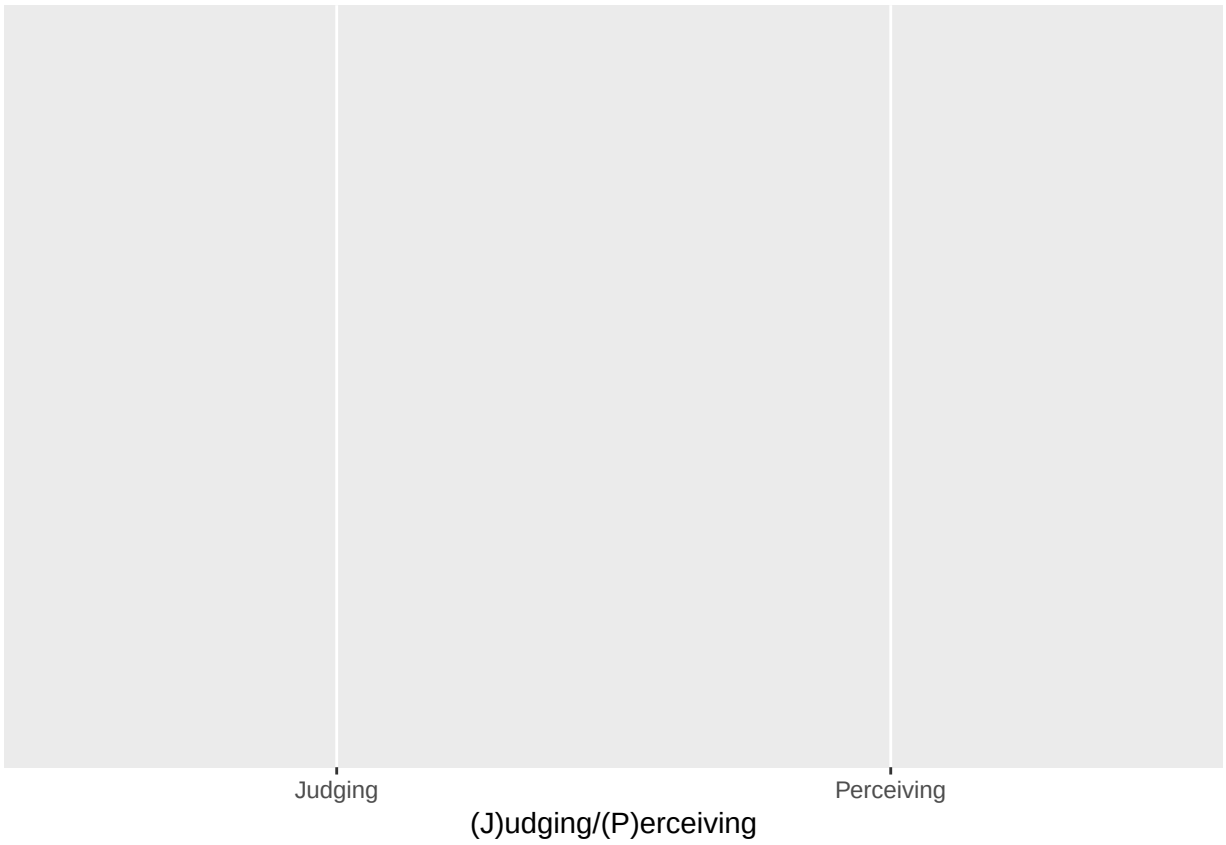
(S)ensing/I(N)tuition	(T)hinking/(F)eeling	(J)udging/(P)erceiving	(E)xtroversion/(I)ntroversion	count
Sensing	Feeling	Judging	Introversion	1
Intuition	Thinking	Judging	Introversion	1
Sensing	Thinking	Judging	Extroversion	1
Intuition	Thinking	Judging	Extroversion	1
Intuition	Feeling	Perceiving	Introversion	1

```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
  geom_bar(position = "dodge") +
  scale_fill_manual(values = alpha(c("lightgrey", "darkgrey", "blue", "violet"), c(.3, .3, .6, .6))) +
  labs(x = "") +
  labs(y = "") +
  labs(fill = "") +
  labs(title = "Frequency of Myers-Briggs Types") +
  labs(subtitle = "Expected among 1000 individuals | @evamaerey | Source: http://www.myersbriggs.org/")
  theme_bw(base_size = 10, base_family = "Times")
```

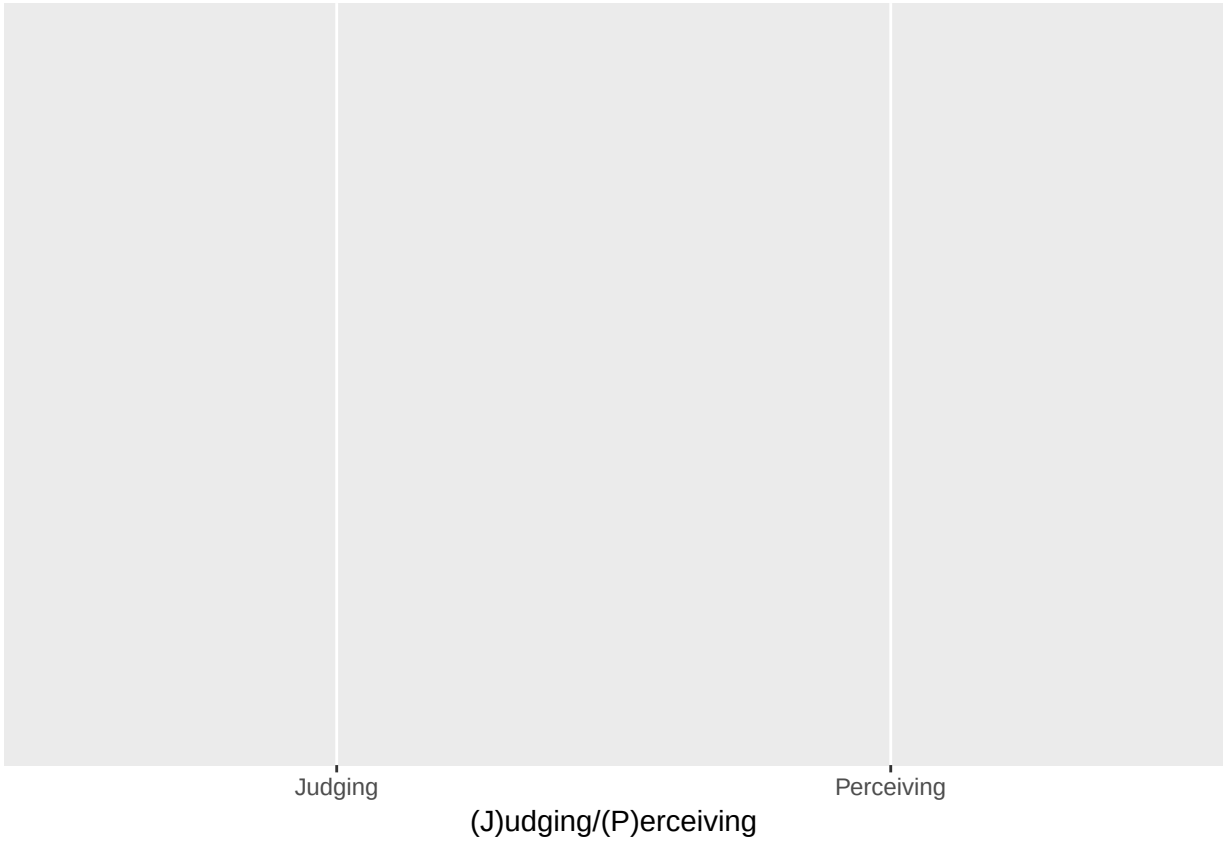

Frequency of Myers−Briggs Types
Expected among 1000 individuals | @evamaerey | Source: http://www.myersbriggs.org/
Intuition
Sensing
Extroversion
Introversion
100
50
0
Judging
Perceiving
Feeling
Thinking

```
ggplot(d)
```

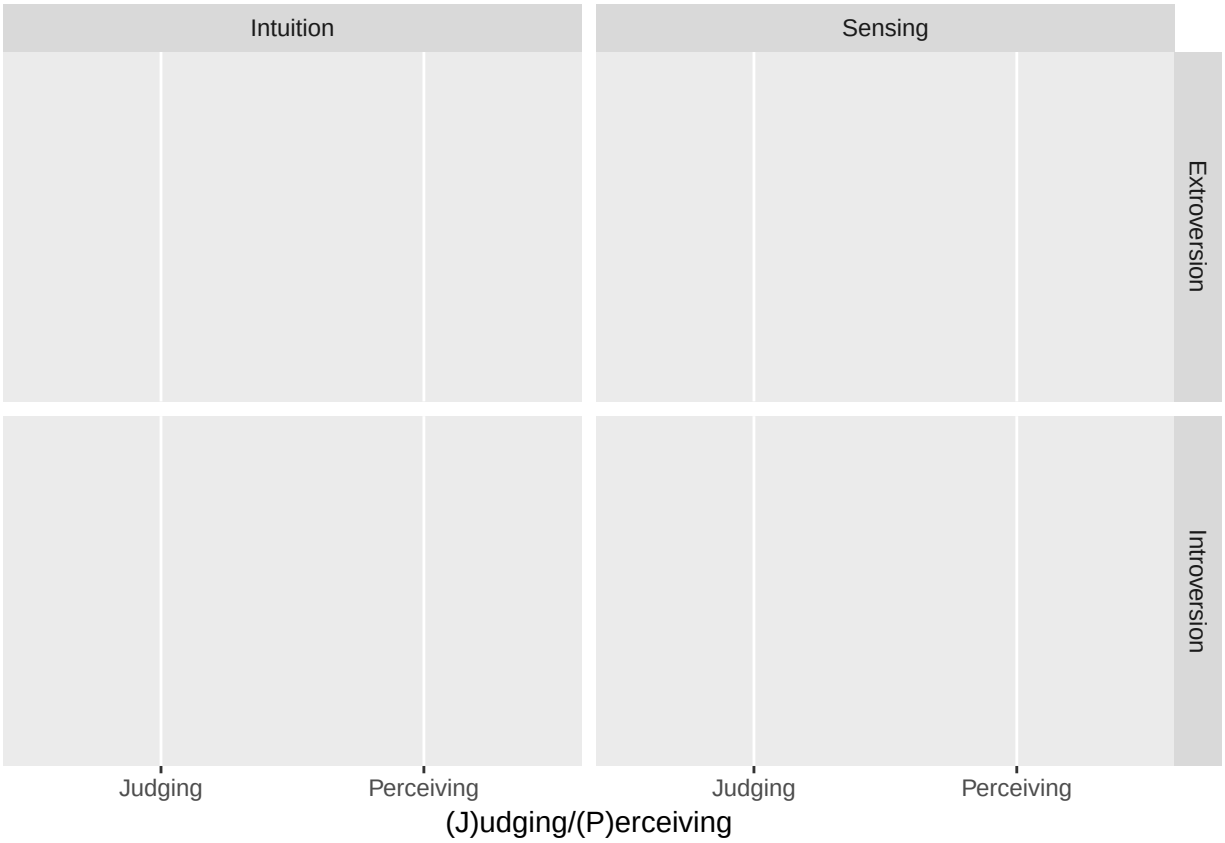
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`)
```



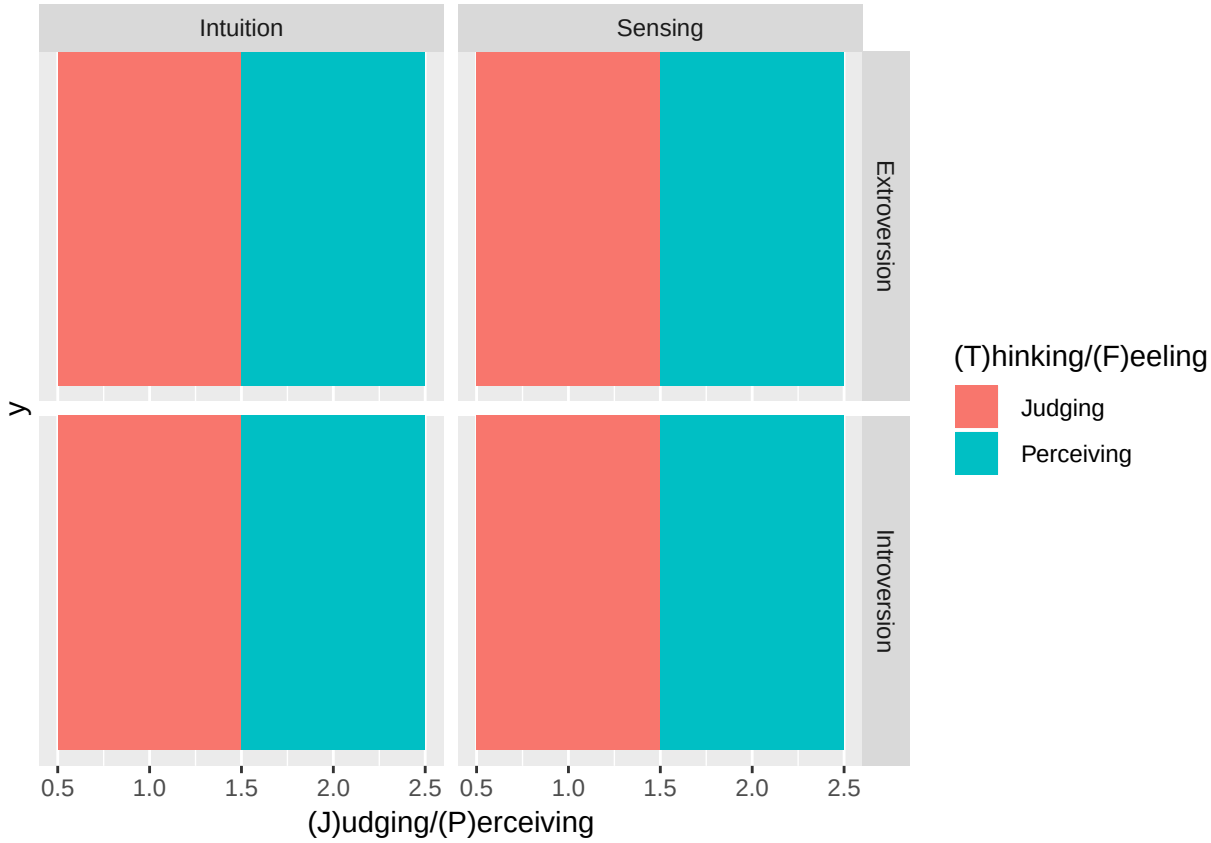
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`)
```



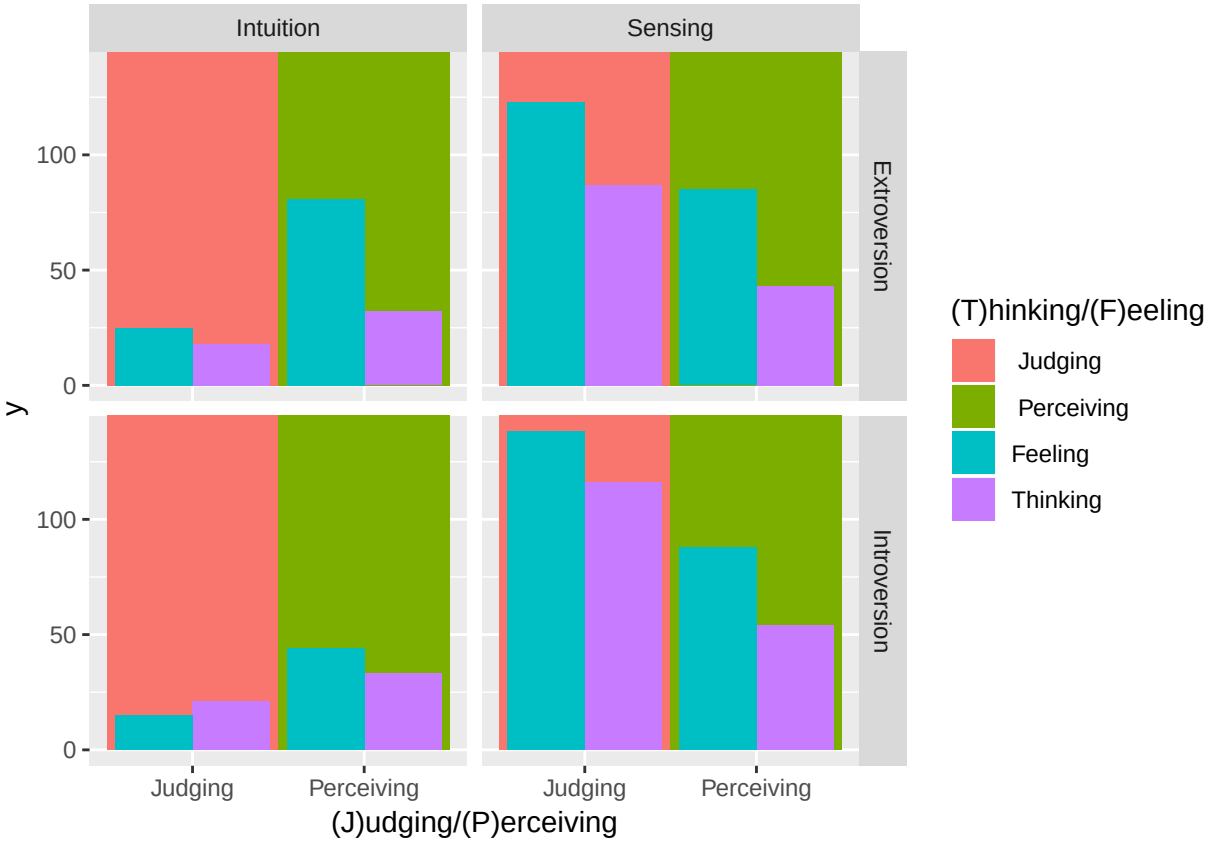
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`)
```



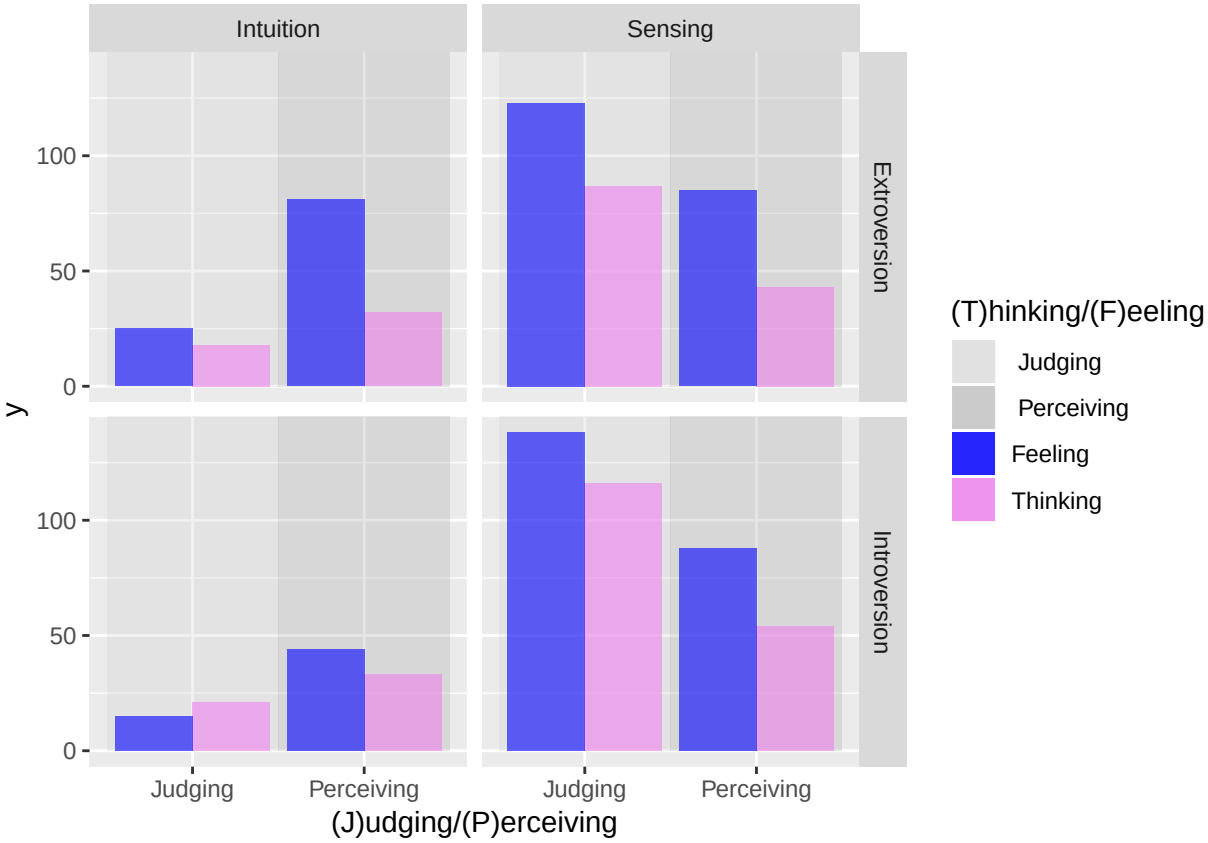
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
```



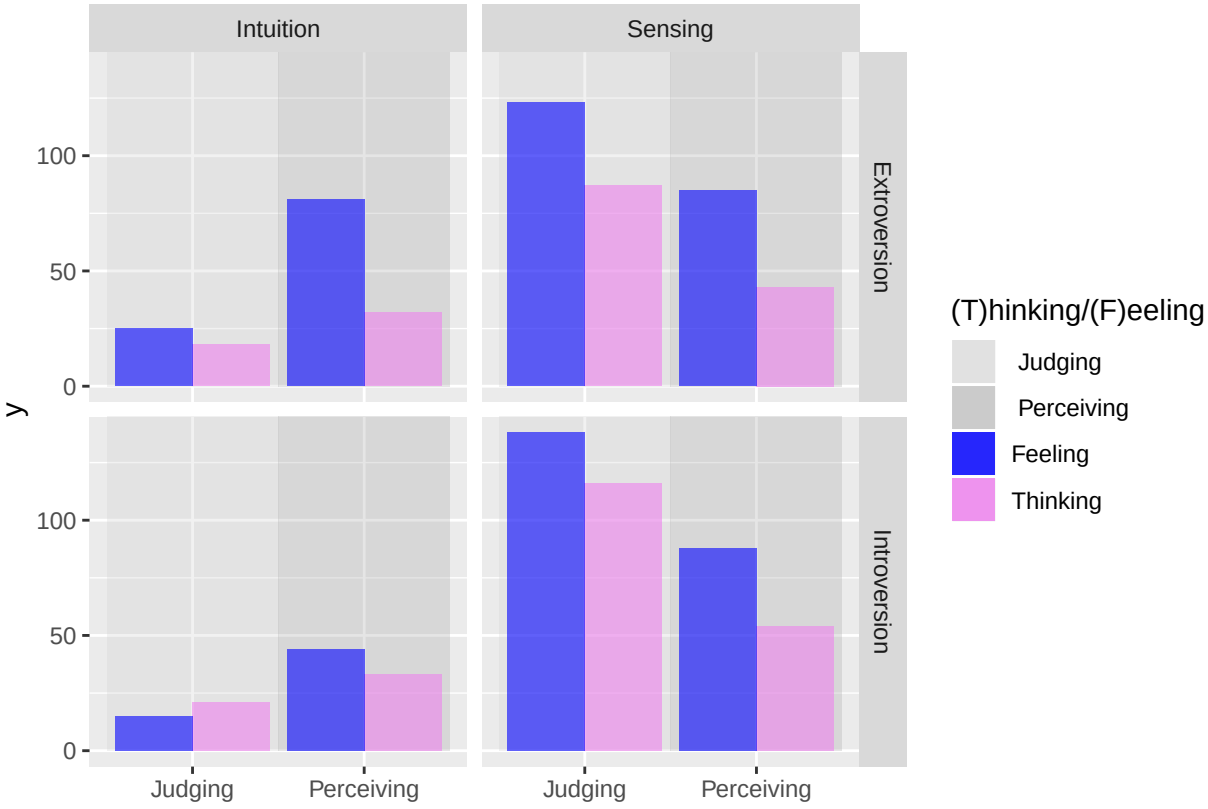
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
  geom_bar(position = "dodge")
```



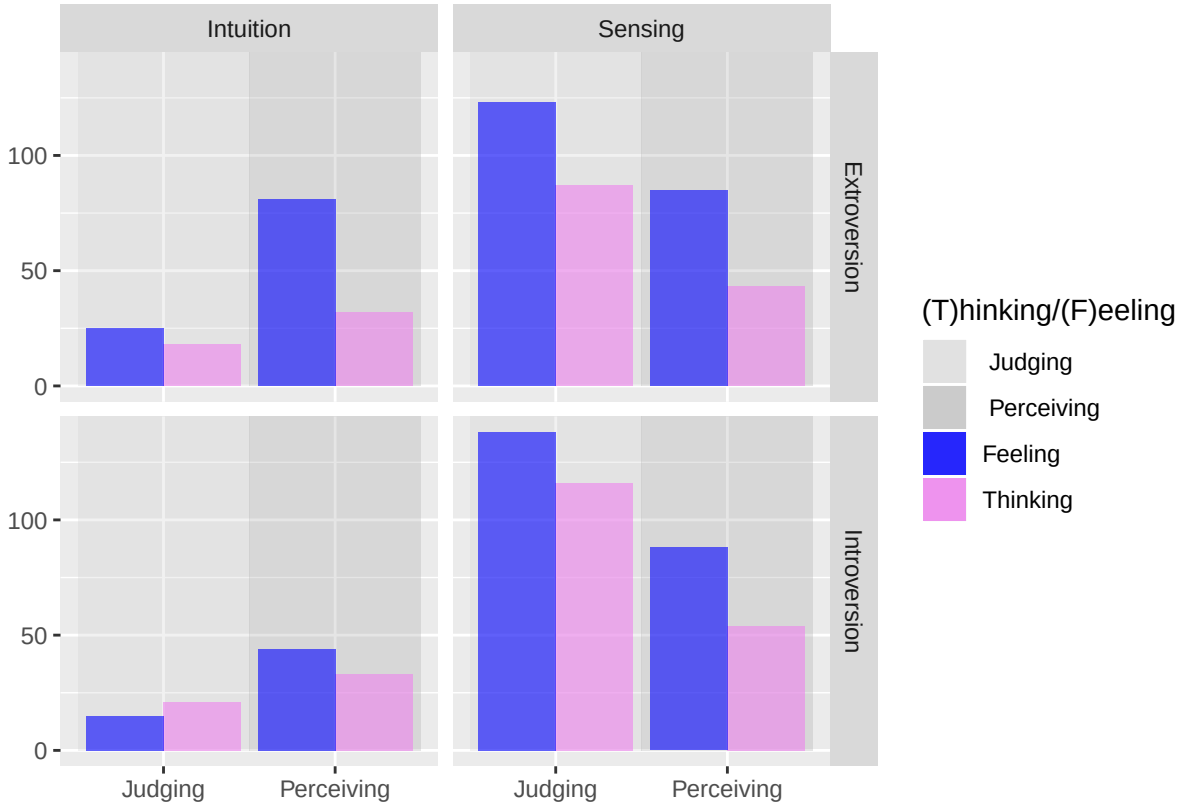
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
  geom_bar(position = "dodge") +
  scale_fill_manual(values = alpha(c("lightgrey", "darkgrey", "blue", "violet"), c(.3, .3, .6, .6)))
```



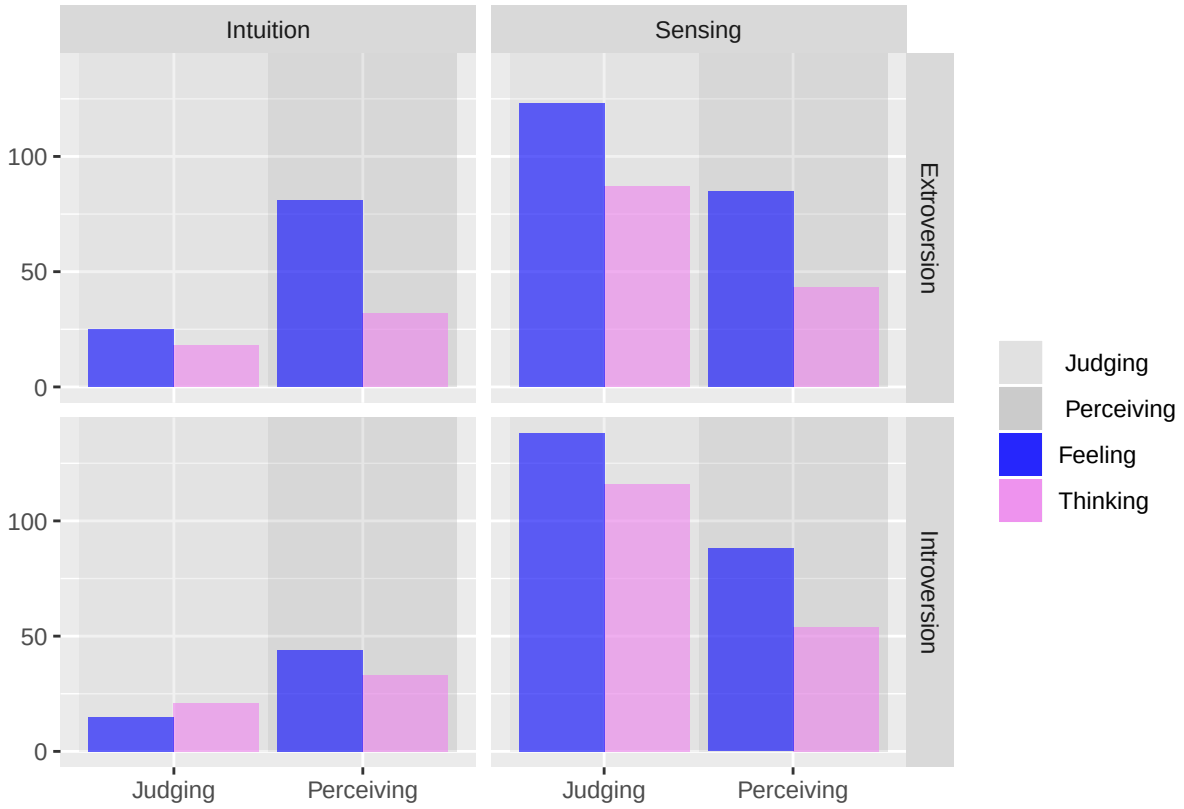
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
  geom_bar(position = "dodge") +
  scale_fill_manual(values = alpha(c("lightgrey", "darkgrey", "blue", "violet"), c(.3, .3, .6, .6))) +
  labs(x = "")
```



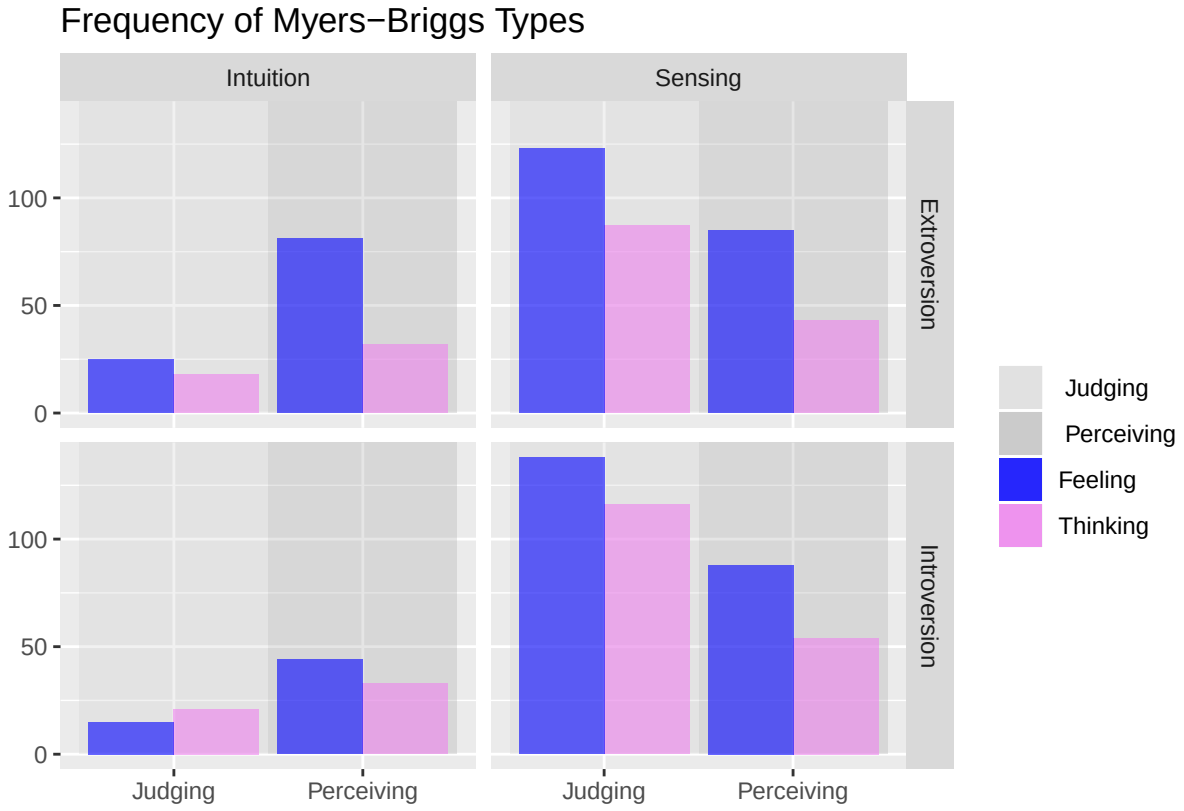
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
  geom_bar(position = "dodge") +
  scale_fill_manual(values = alpha(c("lightgrey", "darkgrey", "blue", "violet"), c(.3, .3, .6, .6))) +
  labs(x = "") +
  labs(y = "")
```



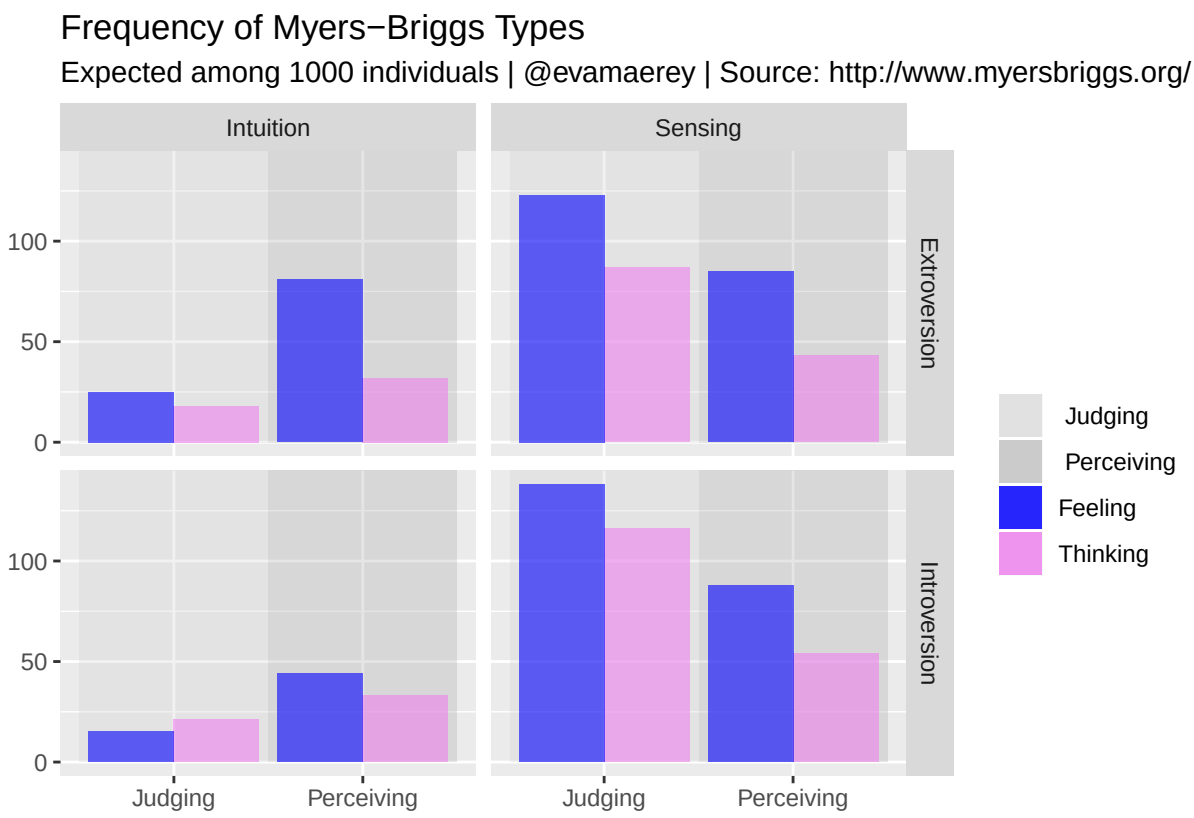
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
  geom_bar(position = "dodge") +
  scale_fill_manual(values = alpha(c("lightgrey", "darkgrey", "blue", "violet"), c(.3, .3, .6, .6))) +
  labs(x = "") +
  labs(y = "") +
  labs(fill = "")
```



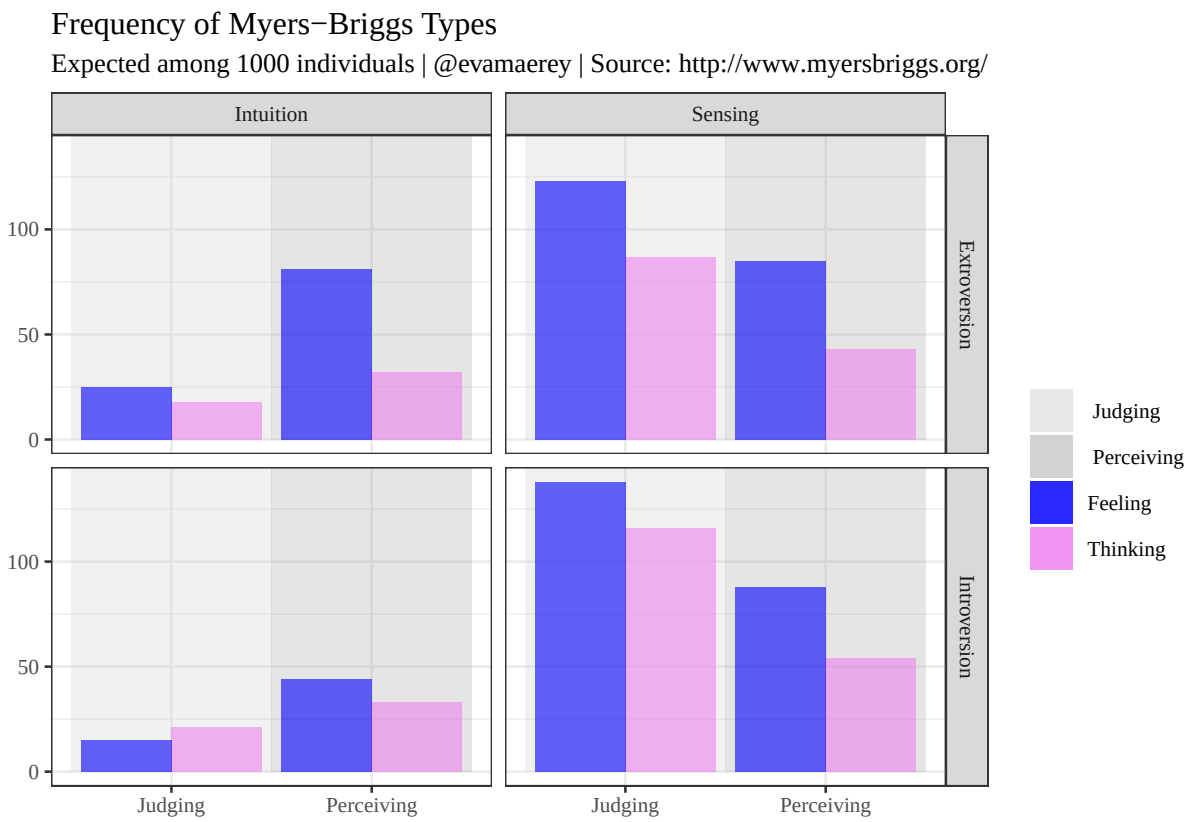
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
  geom_bar(position = "dodge") +
  scale_fill_manual(values = alpha(c("lightgrey", "darkgrey", "blue", "violet"), c(.3, .3, .6, .6))) +
  labs(x = "") +
  labs(y = "") +
  labs(fill = "") +
  labs(title = "Frequency of Myers-Briggs Types")
```



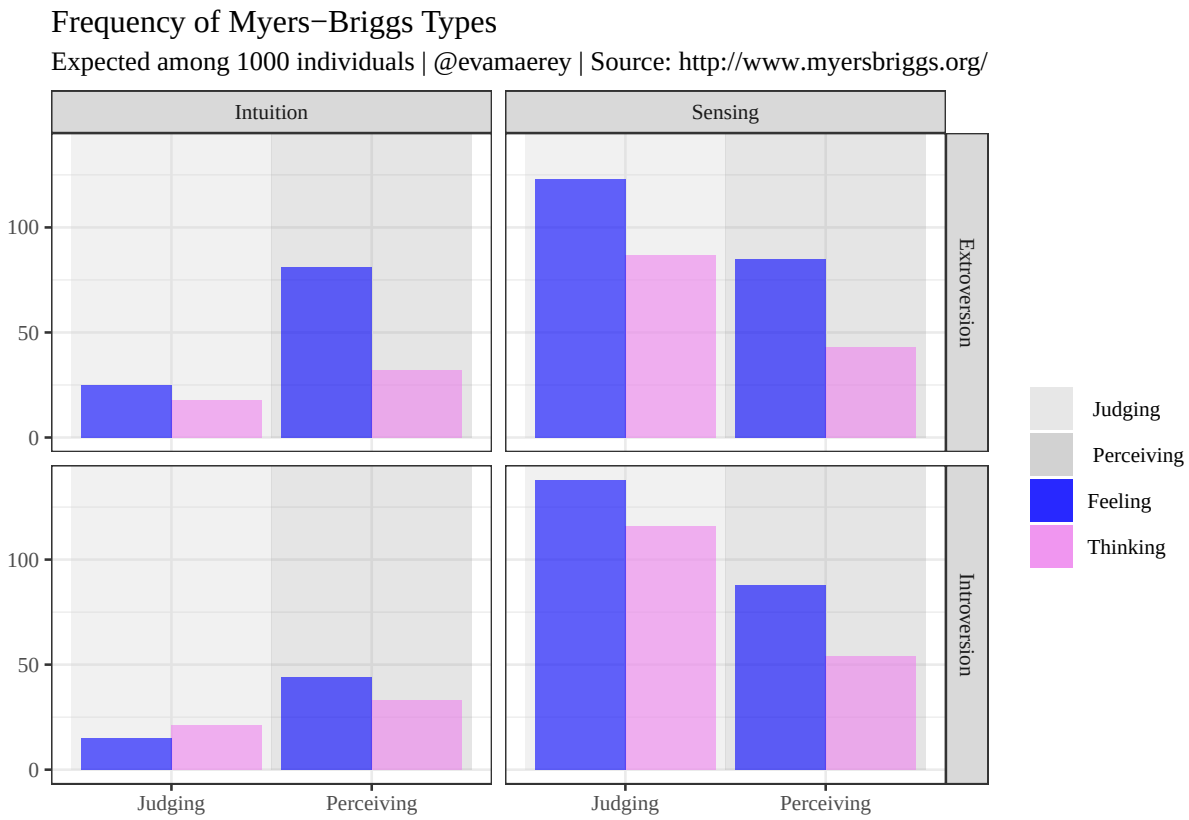
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
  geom_bar(position = "dodge") +
  scale_fill_manual(values = alpha(c("lightgrey", "darkgrey", "blue", "violet"), c(.3, .3, .6, .6))) +
  labs(x = "") +
  labs(y = "") +
  labs(fill = "") +
  labs(title = "Frequency of Myers-Briggs Types") +
  labs(subtitle = "Expected among 1000 individuals | @evamaerey | Source: http://www.myersbriggs.org/")
```



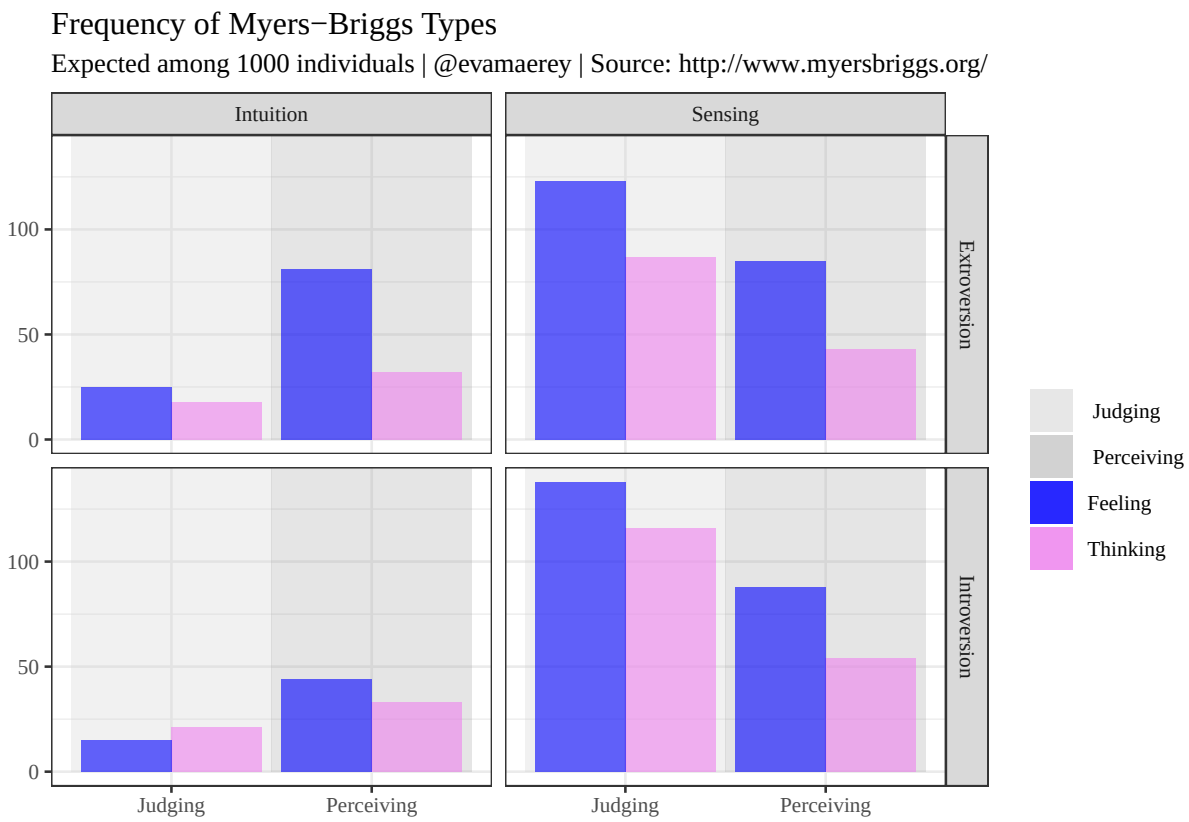
```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
  geom_bar(position = "dodge") +
  scale_fill_manual(values = alpha(c("lightgrey", "darkgrey", "blue", "violet"), c(.3, .3, .6, .6))) +
  labs(x = "") +
  labs(y = "") +
  labs(fill = "") +
  labs(title = "Frequency of Myers-Briggs Types") +
  labs(subtitle = "Expected among 1000 individuals | @evamaerey | Source: http://www.myersbriggs.org/")
  theme_bw(base_size = 10, base_family = "Times")
```



```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
  geom_bar(position = "dodge") +
  scale_fill_manual(values = alpha(c("lightgrey", "darkgrey", "blue", "violet"), c(.3, .3, .6, .6))) +
  labs(x = "") +
  labs(y = "") +
  labs(fill = "") +
  labs(title = "Frequency of Myers-Briggs Types") +
  labs(subtitle = "Expected among 1000 individuals | @evamaerey | Source: http://www.myersbriggs.org/")
  theme_bw(base_size = 10, base_family = "Times")
```



```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~ `(S)ensing/I(N)tuition`) +
  geom_rect(aes(x = NULL, y = NULL, xmin = mins, xmax = max, fill = `judging perceiving`), ymin = 0, yma
  geom_bar(position = "dodge") +
  scale_fill_manual(values = alpha(c("lightgrey", "darkgrey", "blue", "violet"), c(.3, .3, .6, .6))) +
  labs(x = "") +
  labs(y = "") +
  labs(fill = "") +
  labs(title = "Frequency of Myers-Briggs Types") +
  labs(subtitle = "Expected among 1000 individuals | @evamaerey | Source: http://www.myersbriggs.org/")
  theme_bw(base_size = 10, base_family = "Times")
```



Chapter 13

Wine

Wine production in Europe may have been volitile during the years plotted because of production control policies implemented by the EU. The big three, Italy, France and Spain, particularly saw a lot of volitility early in this period.

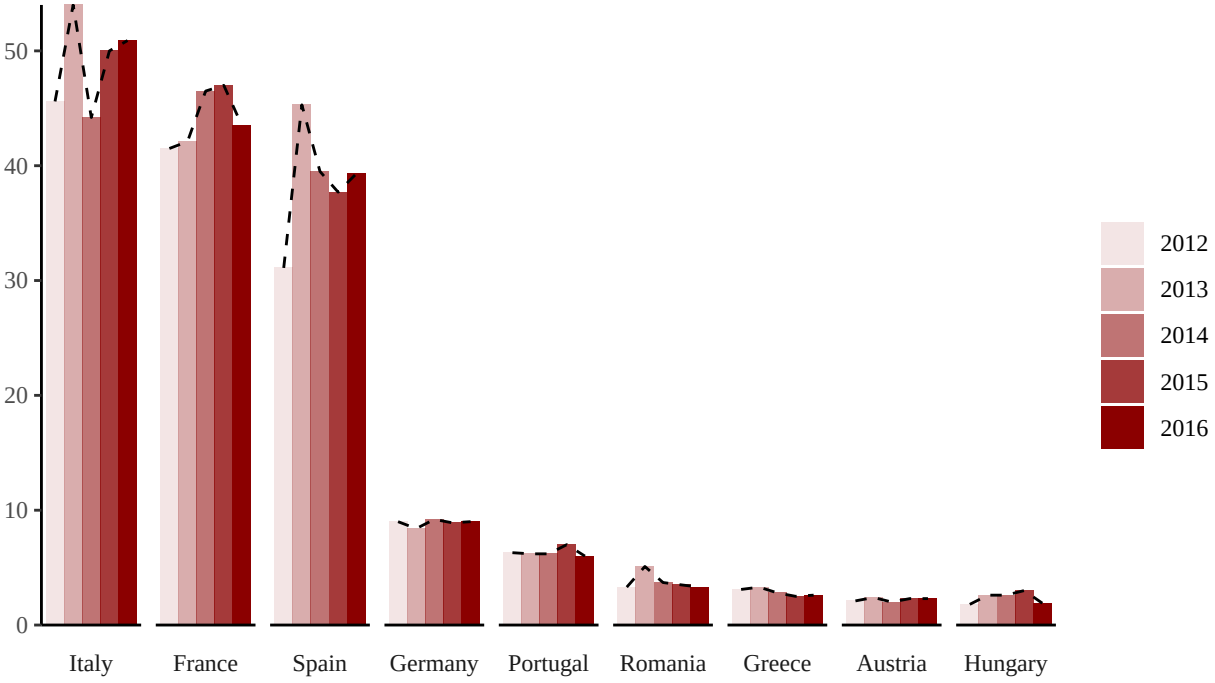
```
df <- readxl::read_xlsx("raw_data/Wine_Production_by_country.xlsx") %>%
  filter(Country != "World total")

Europe <- c(
  "Italy", "France", "Spain",
  "Germany", "Portugal", "Romania",
  "Austria", "Greece", "Hungary"
)
```

```
theme_opts <-
  theme(
    axis.title = element_blank(),
    strip.placement = "outside",
    axis.text.x = element_blank(),
    axis.ticks.x = element_blank(),
    strip.background = element_blank(),
    plot.caption = element_text(size = 10)
  )
```

```
  ggplot(df %>% filter(Country %in% Europe)) +
  aes(x = Year) +
  aes(y = `Wine production in mhl`) +
  facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1) +
  geom_col(aes(alpha = Year), position = "dodge", fill = "darkred", width = 1) +
  geom_line(col = "black", lty = 2) +
  scale_y_continuous(expand = c(0, 0)) +
  labs(fill = "") +
  labs(alpha = "")+
  labs(title = "Wine production (mhl) in principle European markets, 2012-2016") +
  labs(subtitle = "The EU program to regulate viticultural production ended upon the 2011/2012 harvest.
  labs(caption = "Design: Gina Reynolds @EvaMaeRey  \nData Source: International Organisation of Vine a
  theme_classic(base_family = "Times") +
  theme_opts
```

Wine production (mhl) in principle European markets, 2012−2016

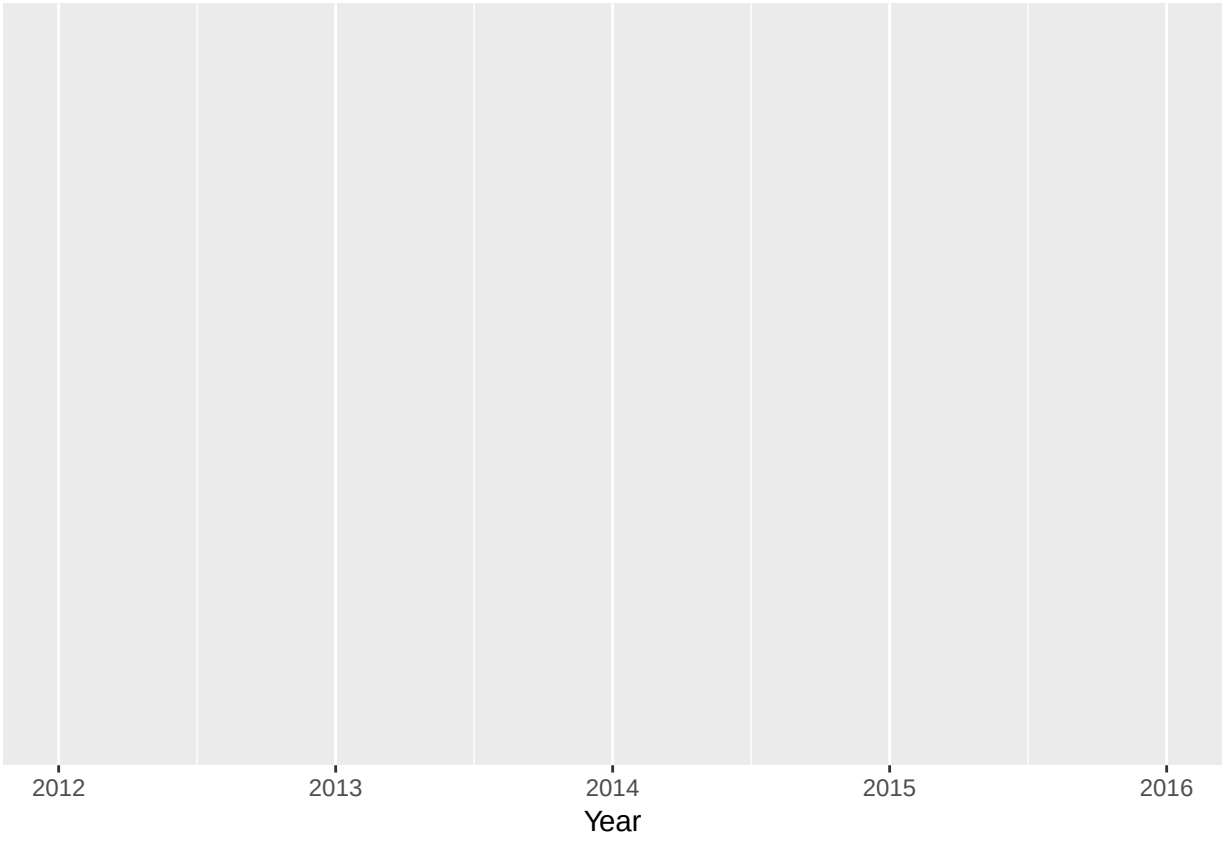
The EU program to regulate viticultural production ended upon the 2011/2012 harvest.



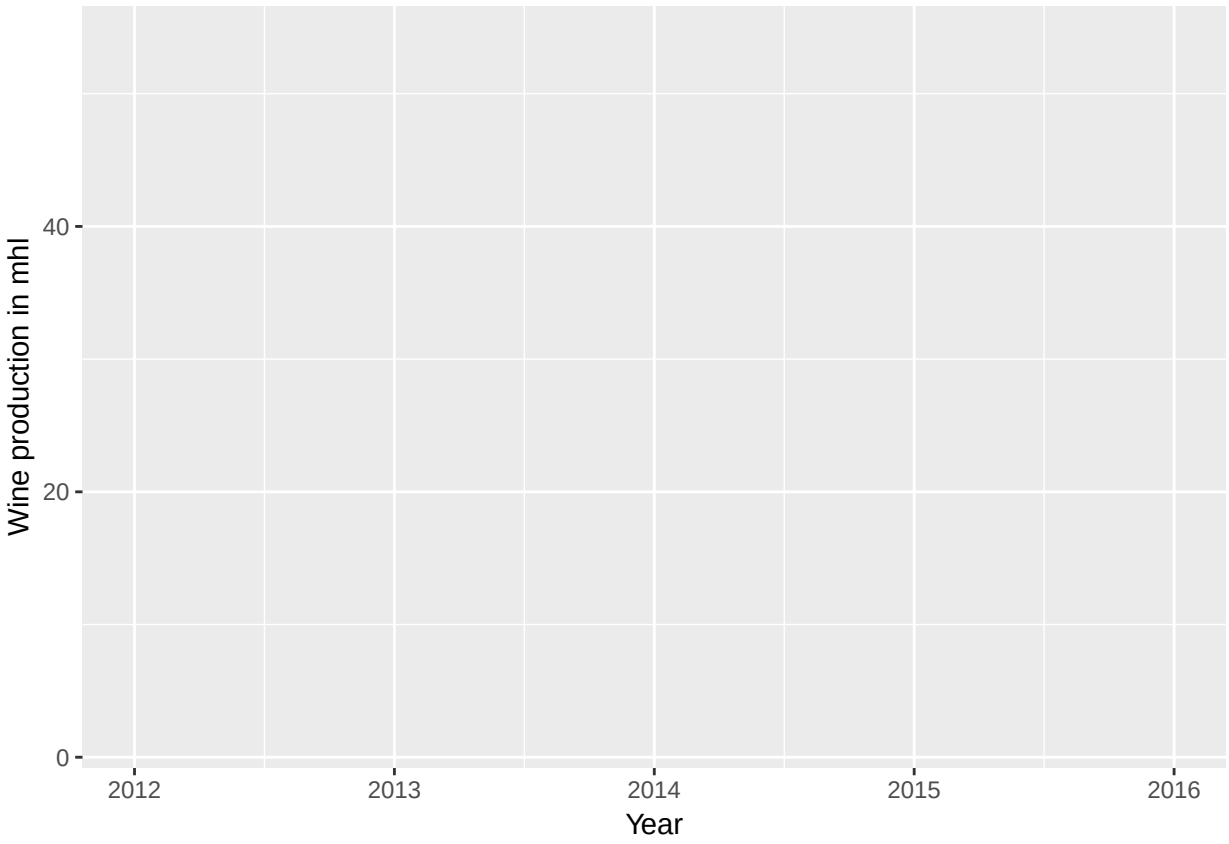
Design: Gina Reynolds @EvaMaeRey
Data Source: International Organisation of Vine and Wine

```
ggplot(df %>% filter(Country %in% Europe))
```

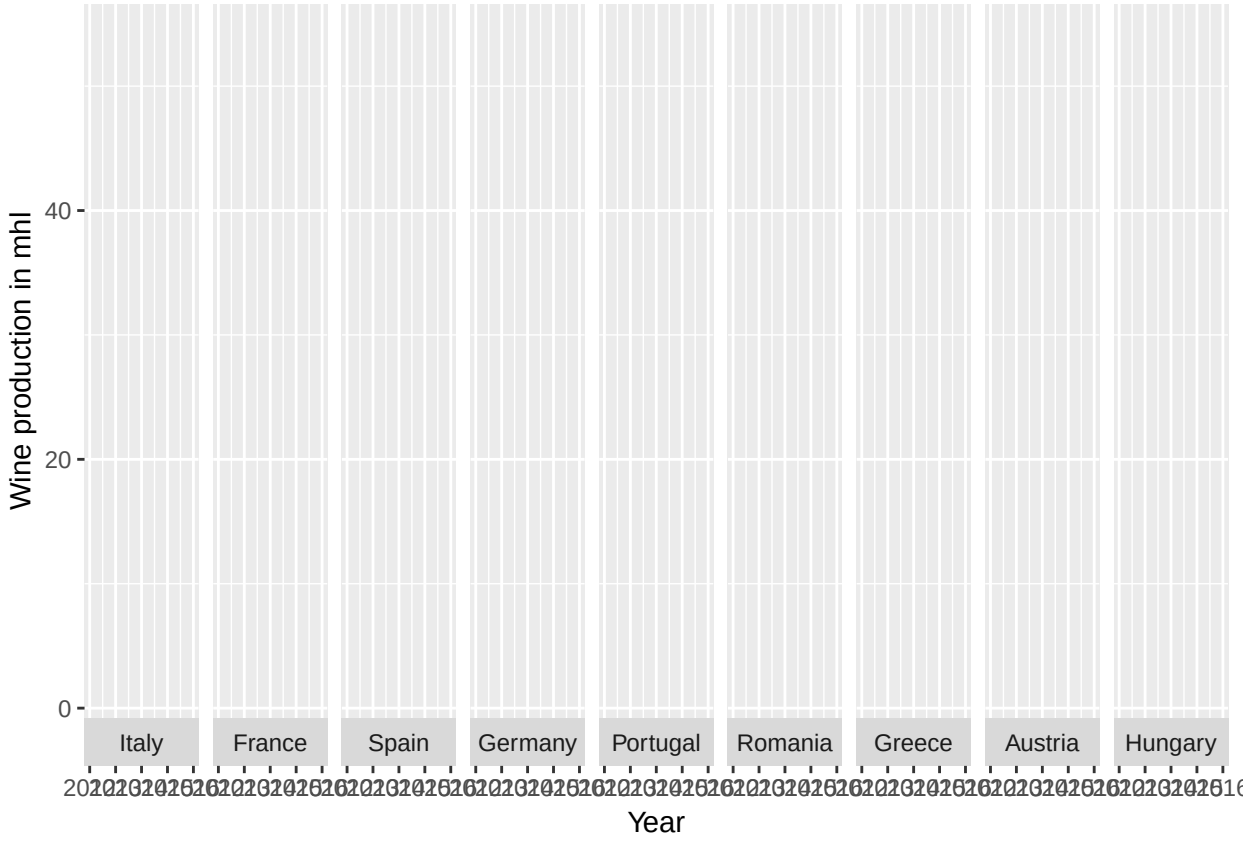
```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year)
```



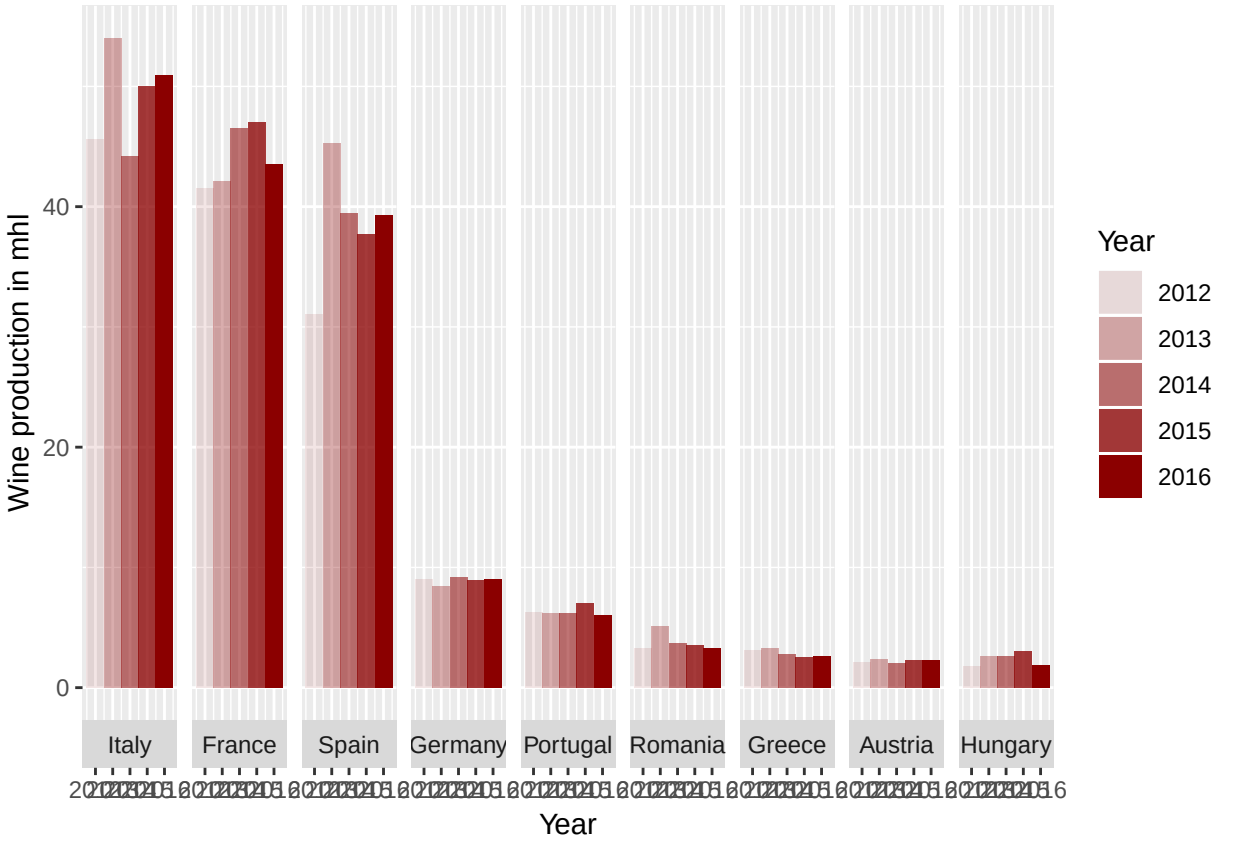
```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`)
```



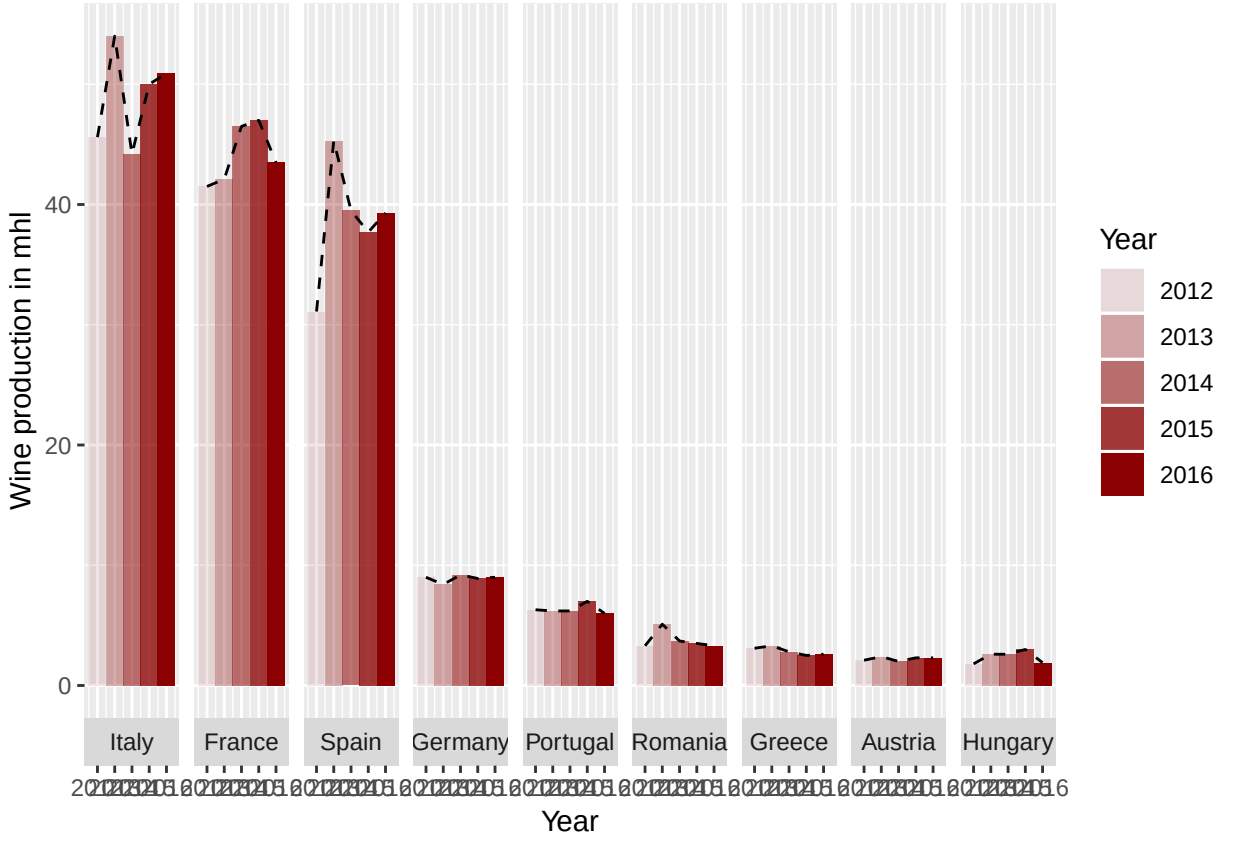
```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`) +
facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1)
```



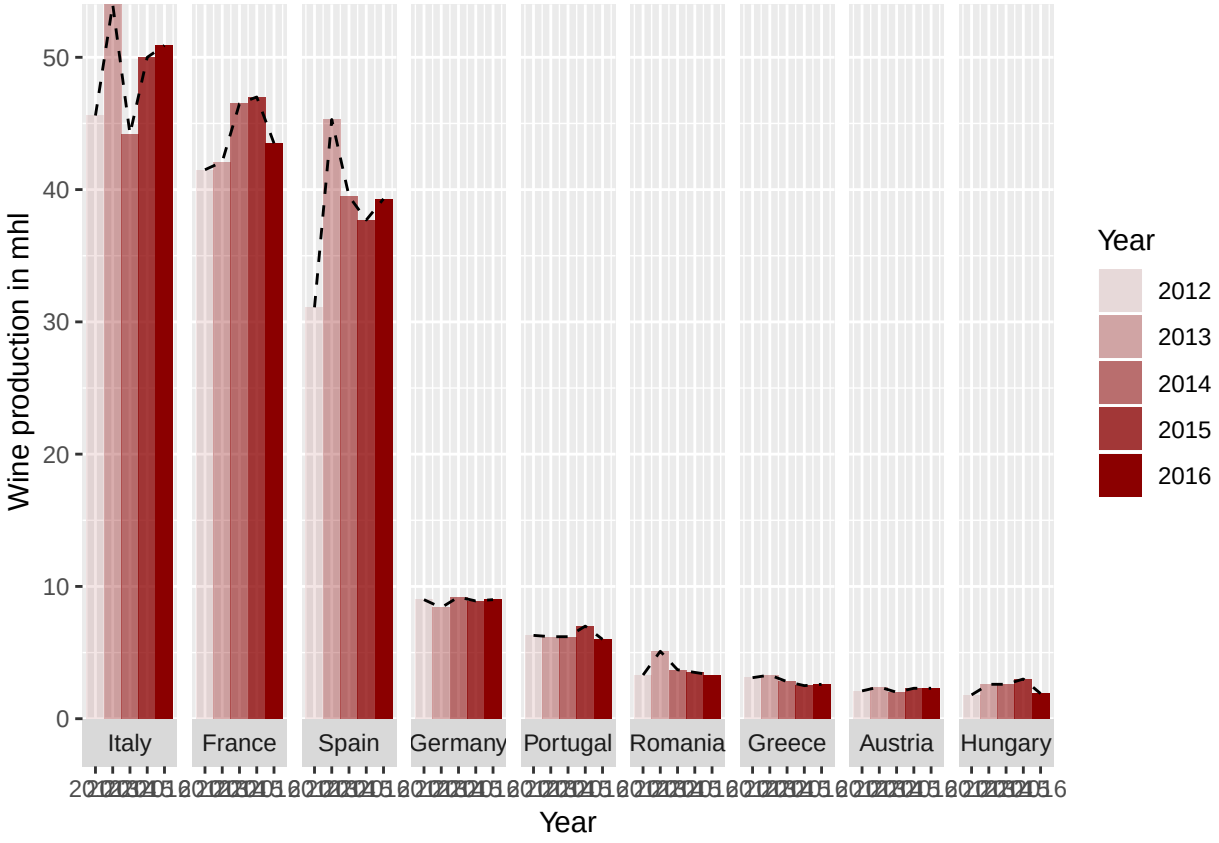
```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`) +
facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1) +
geom_col(aes(alpha = Year), position = "dodge", fill = "darkred", width = 1)
```



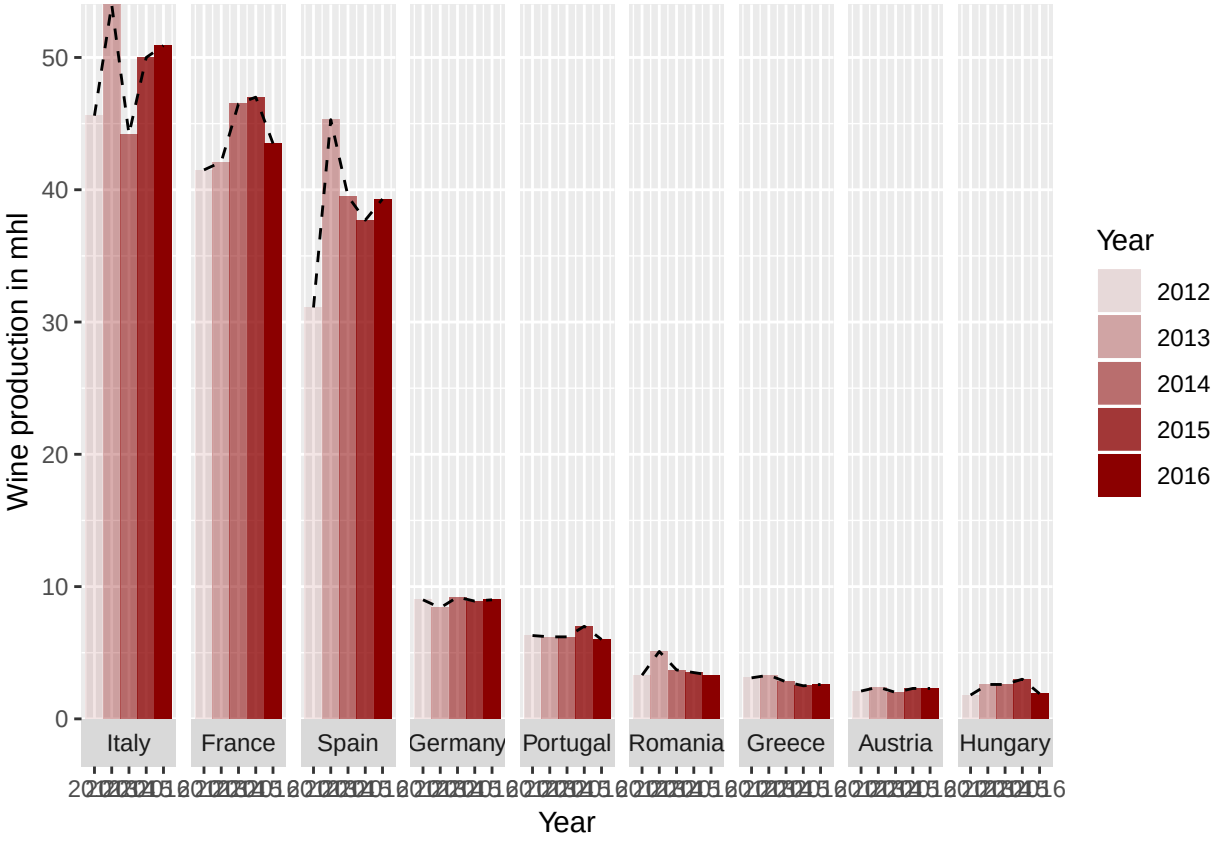
```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`) +
facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1) +
geom_col(aes(alpha = Year), position = "dodge", fill = "darkred", width = 1) +
geom_line(col = "black", lty = 2)
```



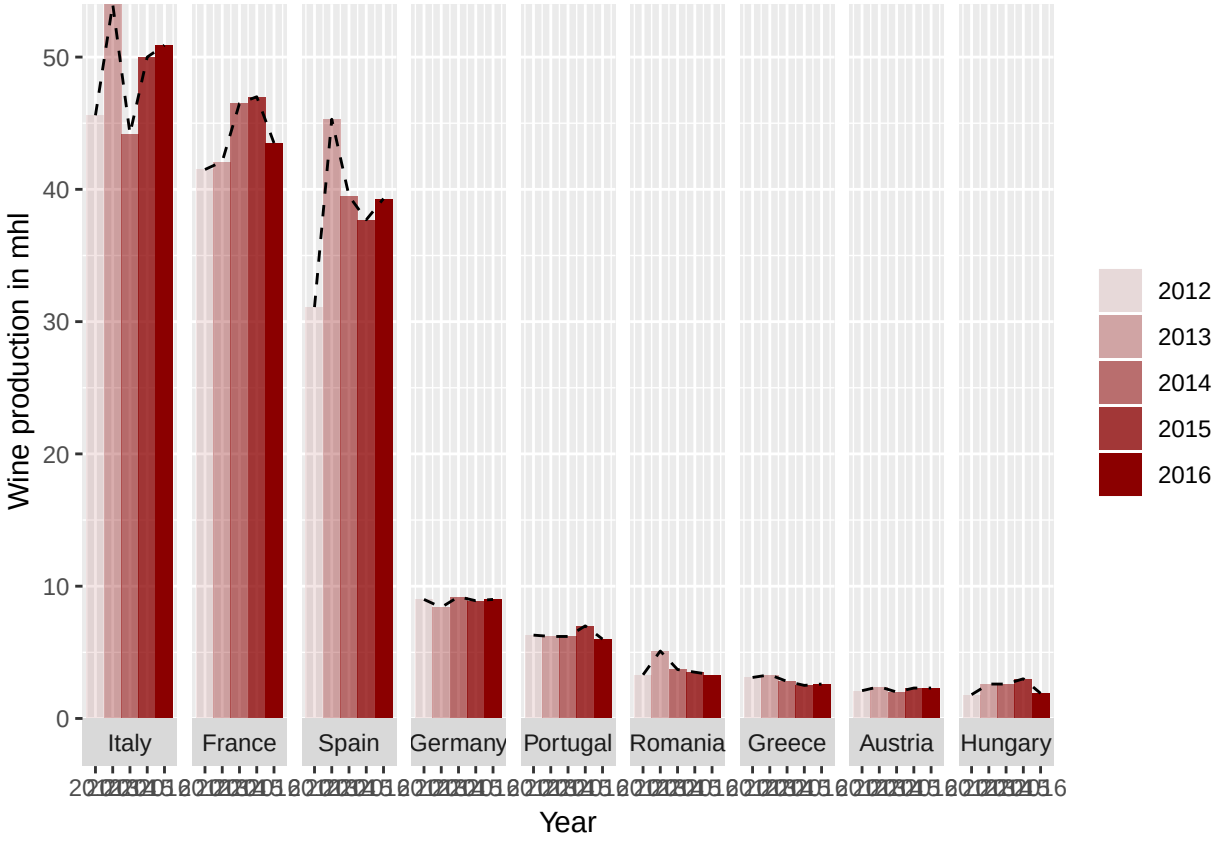
```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`) +
facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1) +
geom_col(aes(alpha = Year), position = "dodge", fill = "darkred", width = 1) +
geom_line(col = "black", lty = 2) +
scale_y_continuous(expand = c(0, 0))
```



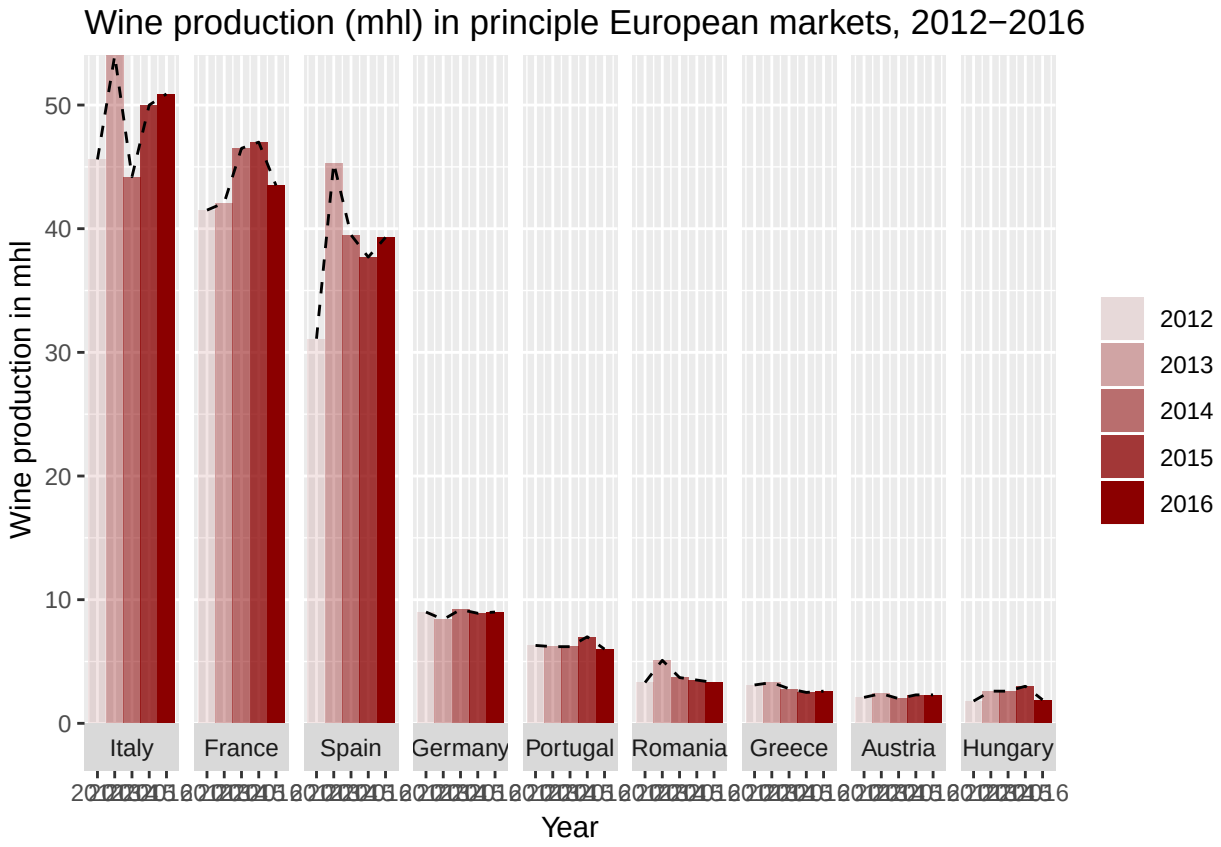
```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`) +
facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1) +
geom_col(aes(alpha = Year), position = "dodge", fill = "darkred", width = 1) +
geom_line(col = "black", lty = 2) +
scale_y_continuous(expand = c(0, 0)) +
labs(fill = "")
```



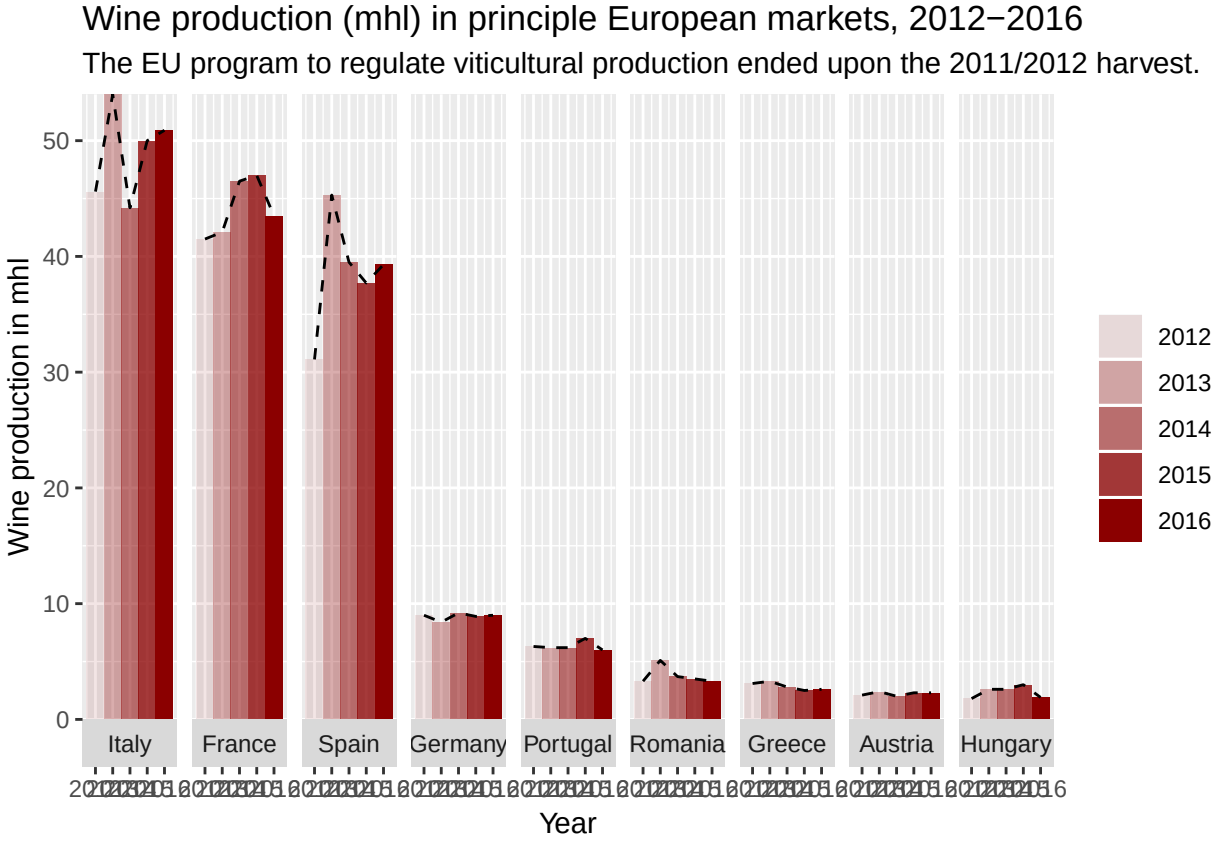
```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`) +
facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1) +
geom_col(aes(alpha = Year), position = "dodge", fill = "darkred", width = 1) +
geom_line(col = "black", lty = 2) +
scale_y_continuous(expand = c(0, 0)) +
labs(fill = "") +
labs(alpha = "")
```



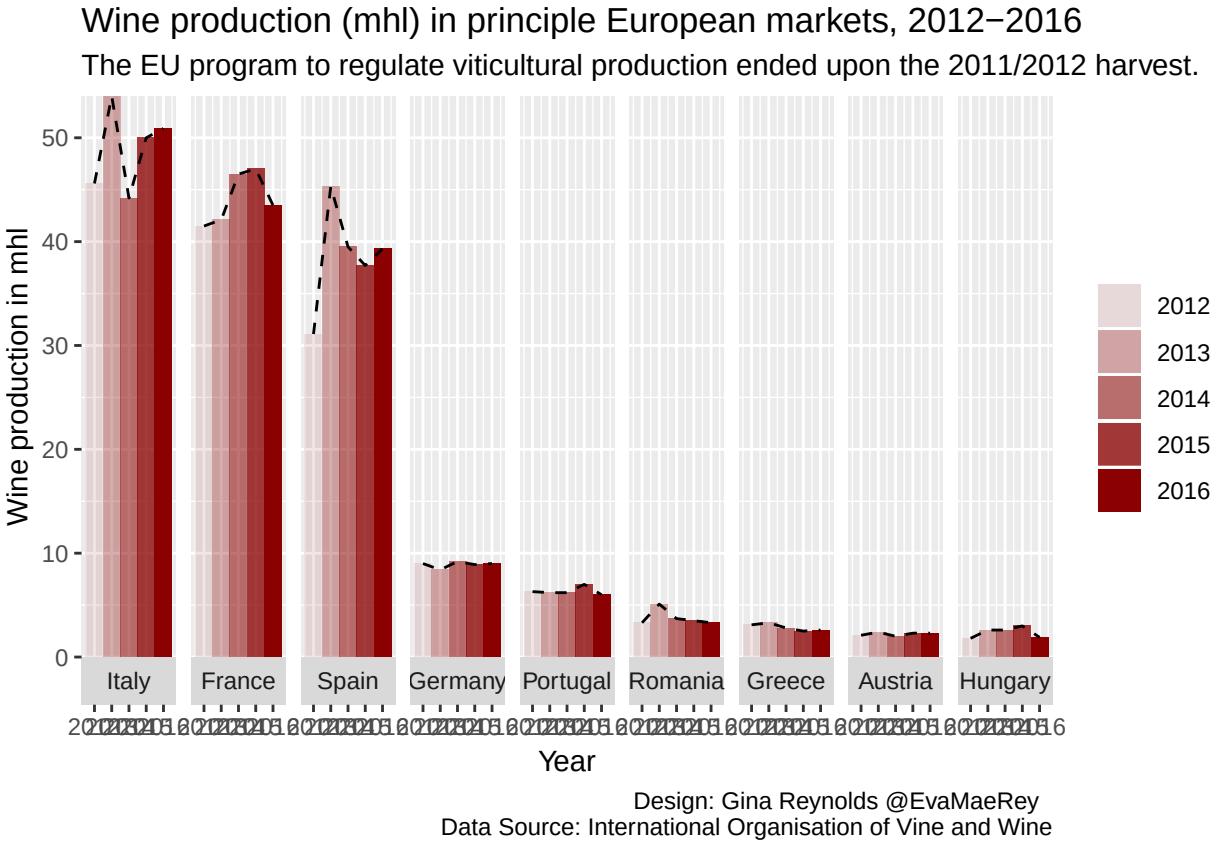
```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`) +
facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1) +
geom_col(aes(alpha = Year), position = "dodge", fill = "darkred", width = 1) +
geom_line(col = "black", lty = 2) +
scale_y_continuous(expand = c(0, 0)) +
labs(fill = "") +
labs(alpha = "")+
labs(title = "Wine production (mhl) in principle European markets, 2012-2016")
```



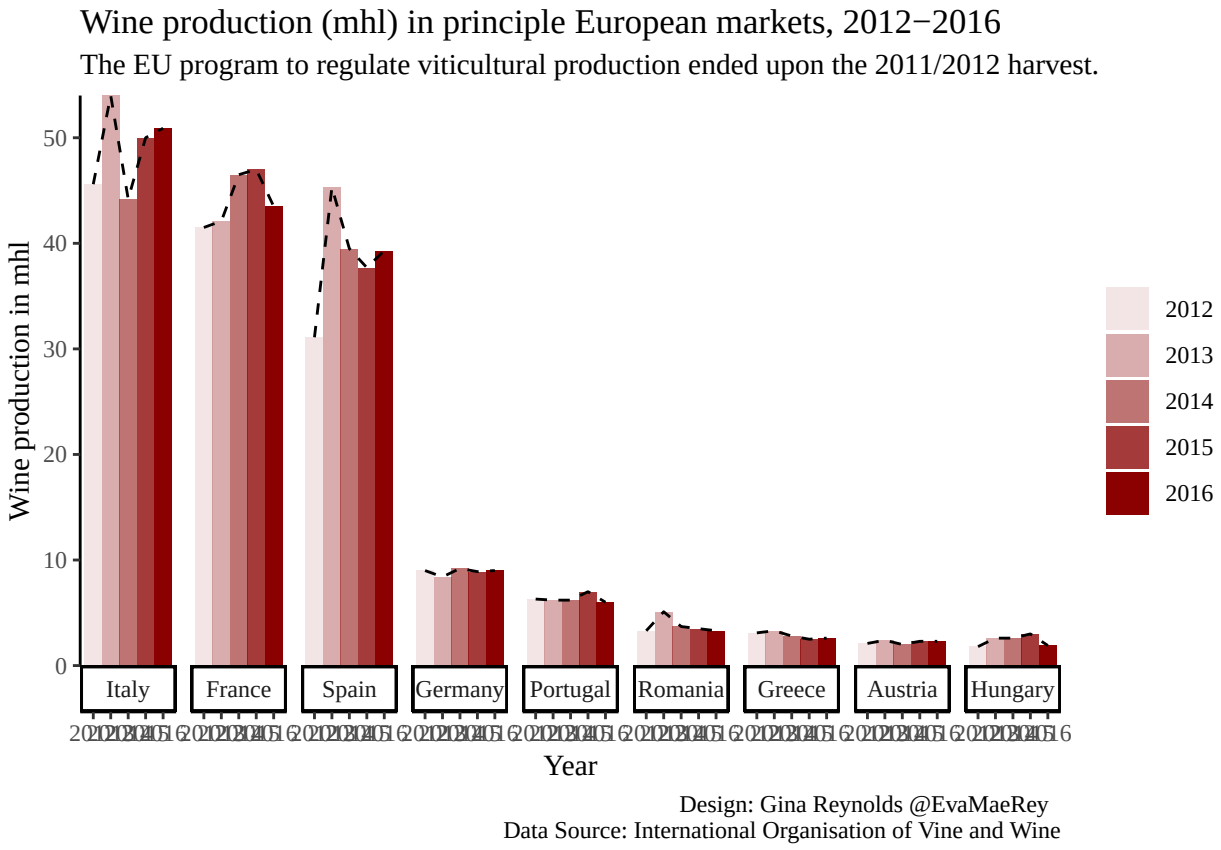
```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`) +
facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1) +
geom_col(aes(alpha = Year), position = "dodge", fill = "darkred", width = 1) +
geom_line(col = "black", lty = 2) +
scale_y_continuous(expand = c(0, 0)) +
labs(fill = "") +
labs(alpha = "")+
labs(title = "Wine production (mhl) in principle European markets, 2012-2016") +
labs(subtitle = "The EU program to regulate viticultural production ended upon the 2011/2012 harvest.
```



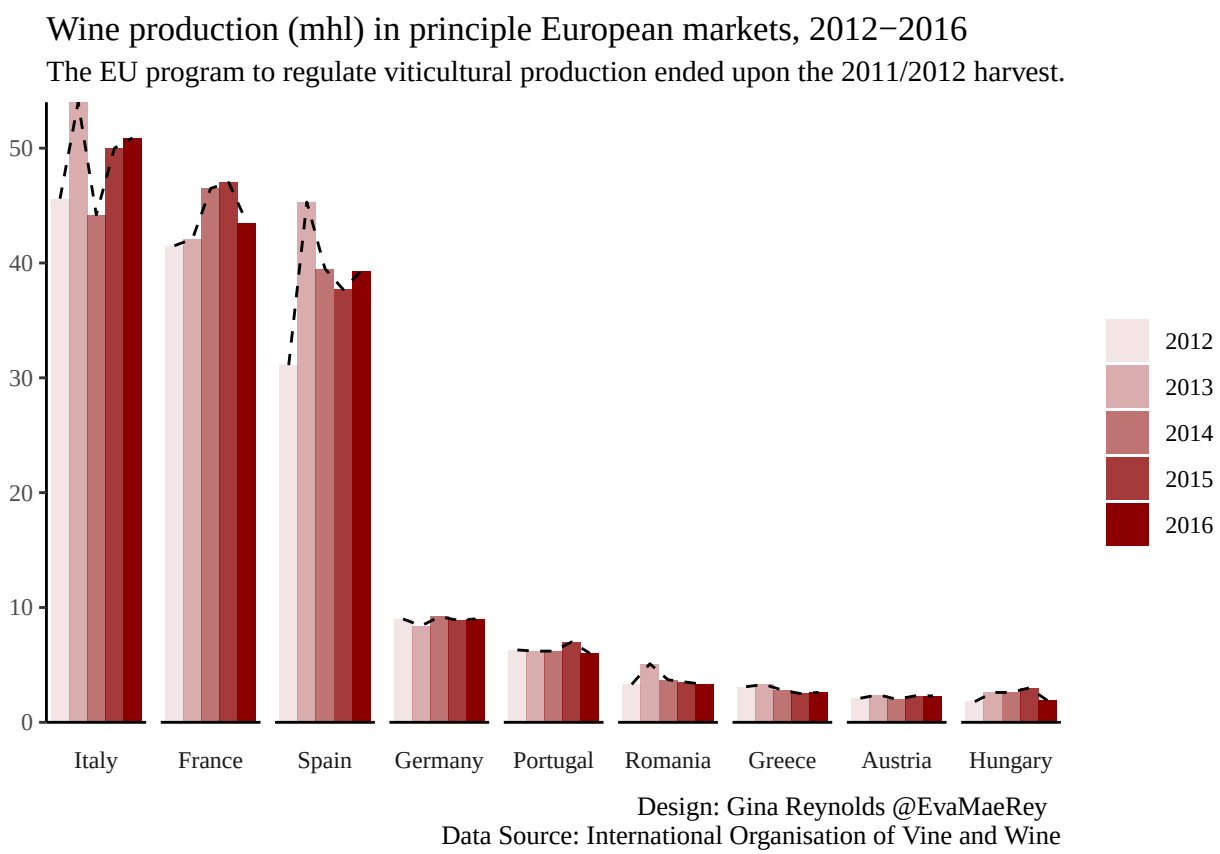
```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`) +
facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1) +
geom_col(aes(alpha = Year), position = "dodge", fill = "darkred", width = 1) +
geom_line(col = "black", lty = 2) +
scale_y_continuous(expand = c(0, 0)) +
labs(fill = "") +
labs(alpha = "")+
labs(title = "Wine production (mhl) in principle European markets, 2012-2016") +
labs(subtitle = "The EU program to regulate viticultural production ended upon the 2011/2012 harvest.
labs(caption = "Design: Gina Reynolds @EvaMaeRey  \nData Source: International Organisation of Vine a
```



```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`) +
facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1) +
geom_col(aes(alpha = Year), position = "dodge", fill = "darkred", width = 1) +
geom_line(col = "black", lty = 2) +
scale_y_continuous(expand = c(0, 0)) +
labs(fill = "") +
labs(alpha = "")+
labs(title = "Wine production (mhl) in principle European markets, 2012-2016") +
labs(subtitle = "The EU program to regulate viticultural production ended upon the 2011/2012 harvest.
labs(caption = "Design: Gina Reynolds @EvaMaeRey  \nData Source: International Organisation of Vine a
theme_classic(base_family = "Times")
```



```
ggplot(df %>% filter(Country %in% Europe)) +
aes(x = Year) +
aes(y = `Wine production in mhl`) +
facet_wrap(~ fct_inorder(Country), strip.position = "bottom", nrow = 1) +
geom_col(aes(alpha = Year), position = "dodge", fill = "darkred", width = 1) +
geom_line(col = "black", lty = 2) +
scale_y_continuous(expand = c(0, 0)) +
labs(fill = "") +
labs(alpha = "")+
labs(title = "Wine production (mhl) in principle European markets, 2012-2016") +
labs(subtitle = "The EU program to regulate viticultural production ended upon the 2011/2012 harvest.
labs(caption = "Design: Gina Reynolds @EvaMaeRey  \nData Source: International Organisation of Vine a
theme_classic(base_family = "Times") +
theme_opts
```



Chapter 14

Arctic Ice

This visualization shows the trend in Arctic Ice Sea Extent, data from the National Snow and Ice Data Center. If I recall correctly, the definition for coverage is the case where at least 15 percent of the sea is ice.

The visualization shows melting and freezing cycles, in accordance with the seasons — and the disconcerting trend of a general decrease in ice extent over the years.

One problem that arises is due to inconsistant number of days in each year. There is a measurement for every day, but leap years contain a extra day. Which means that plotting years over years leads to imperfect alignment. My solution was just to pretend that all the data come from a single year, 2000, and plot each of the years on that scale. The earliest year cycle and last year cycle are highlighted in white.

A random sample from the data set:

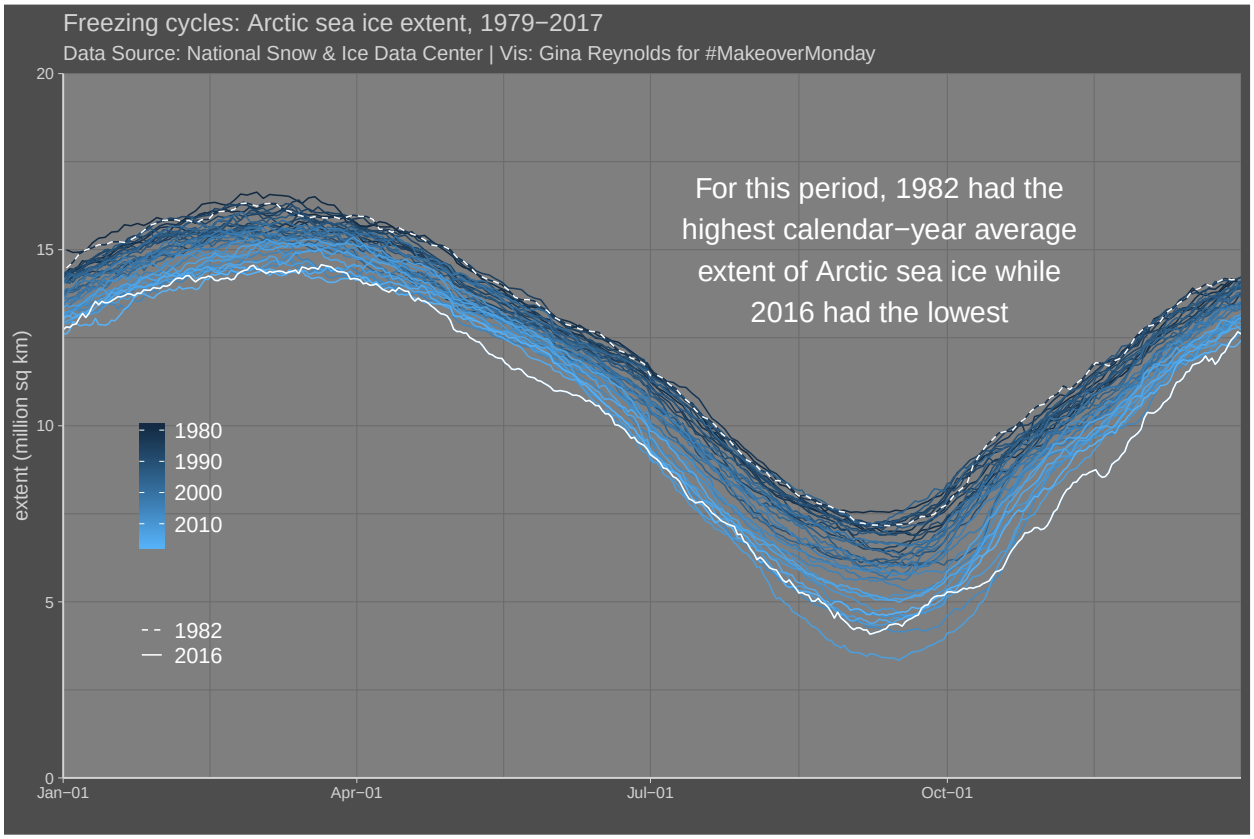
Date	Extent (million sq km)	year	month_day	month_day_plus	proportion_ocean_covered_in_ice	mea
1996-11-19	10.486	1996	11-19	2000-11-19	0.0291278	
2014-10-15	6.926	2014	10-15	2000-10-15	0.0192389	
1993-05-06	13.902	1993	05-06	2000-05-06	0.0386167	
2005-03-15	14.708	2005	03-15	2000-03-15	0.0408556	
2000-02-02	14.959	2000	02-02	2000-02-02	0.0415528	

year	average_coverage	num_days	average_day
1982	12.43945	182	1982-07-02 00:00:00
2016	10.15069	366	2016-07-01 12:00:00

```
theme_opts <- theme(
    legend.background = element_blank(),
    legend.position = c(0.1, .35),
    legend.text = element_text(colour = "white", size = 15),
    plot.background = element_rect(fill = "grey30"),
    plot.title = element_text(colour = "lightgrey"),
    plot.subtitle = element_text(colour = "lightgrey"),
    axis.title = element_text(colour = "lightgrey"),
    axis.line = element_line(colour = "lightgrey"),
    axis.text = element_text(colour = "lightgrey"),
    axis.ticks = element_line(colour = "lightgrey")
  )

# breaks for x axis.
br <- as.numeric(lubridate::ymd(c(
  "2000-01-01", "2000-04-01",
```

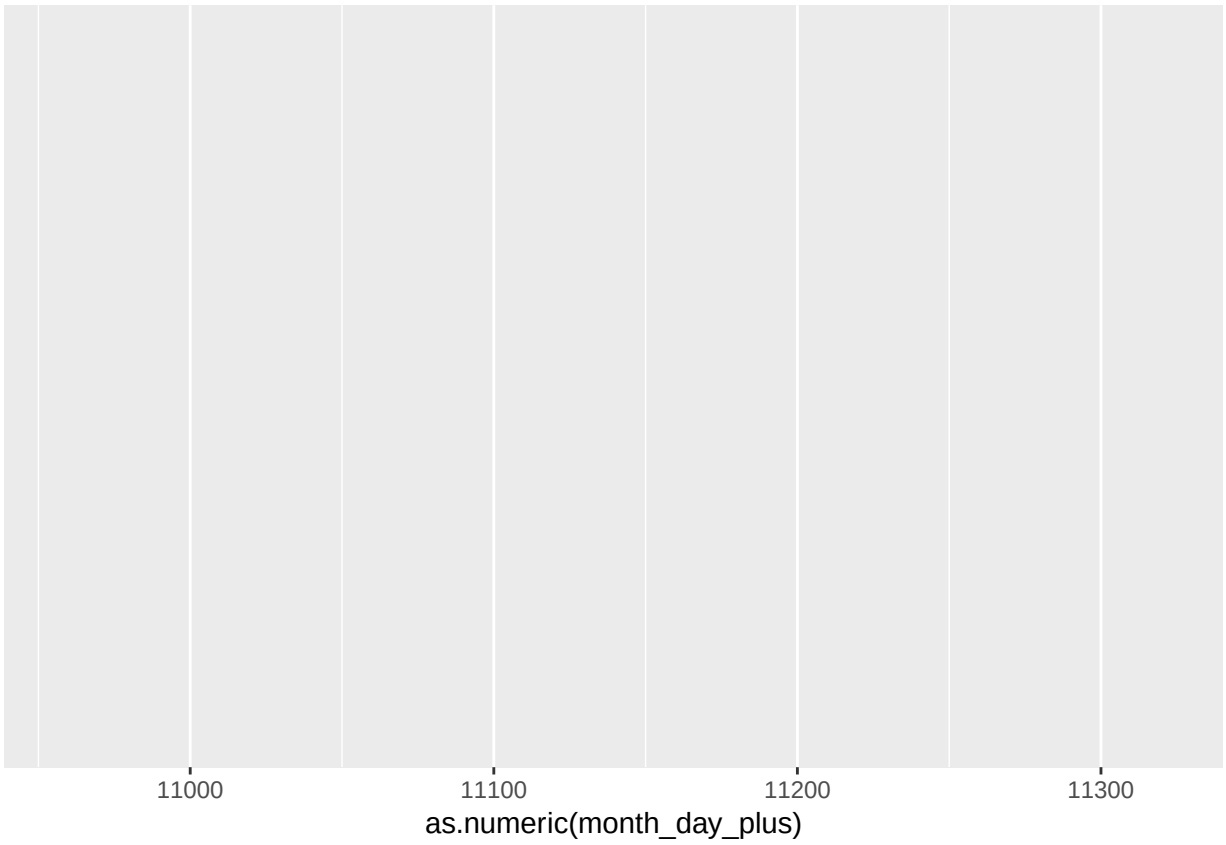
```
  "2000-07-01", "2000-10-01", "2001-01-01"
)))
```

```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") ) +
  annotate( geom = "text", x = 11210, y = 15, label = str_wrap("For this period, 1982 had the highest ca
  labs(x = "") +
  labs(y = "extent (million sq km)") +
  labs(col = "") +
  labs(lty = "") +
  labs(title = "Freezing cycles: Arctic sea ice extent, 1979-2017") +
  labs(subtitle = "Data Source: National Snow & Ice Data Center | Vis: Gina Reynolds for #MakeoverMonday
  theme_dark(base_size = 14) +
  theme_opts
```

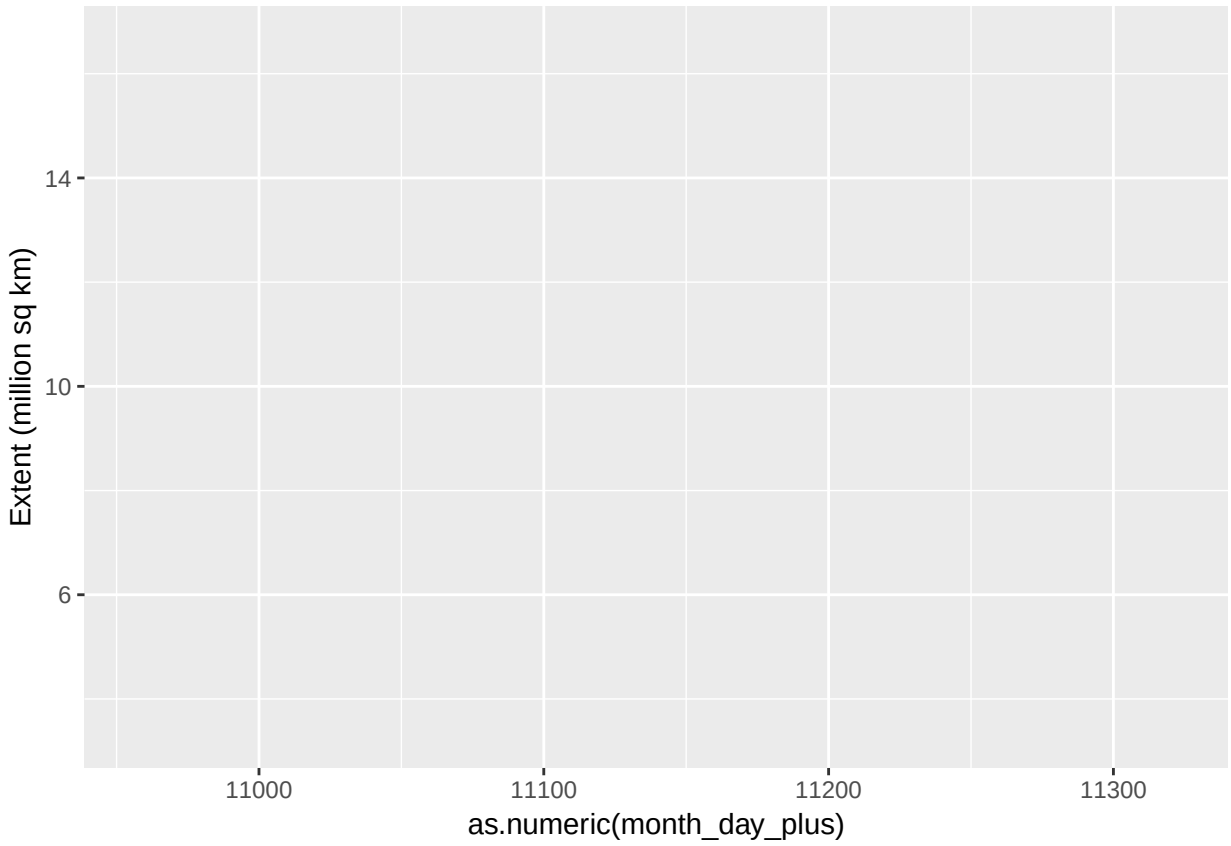


```
ggplot(df)
```

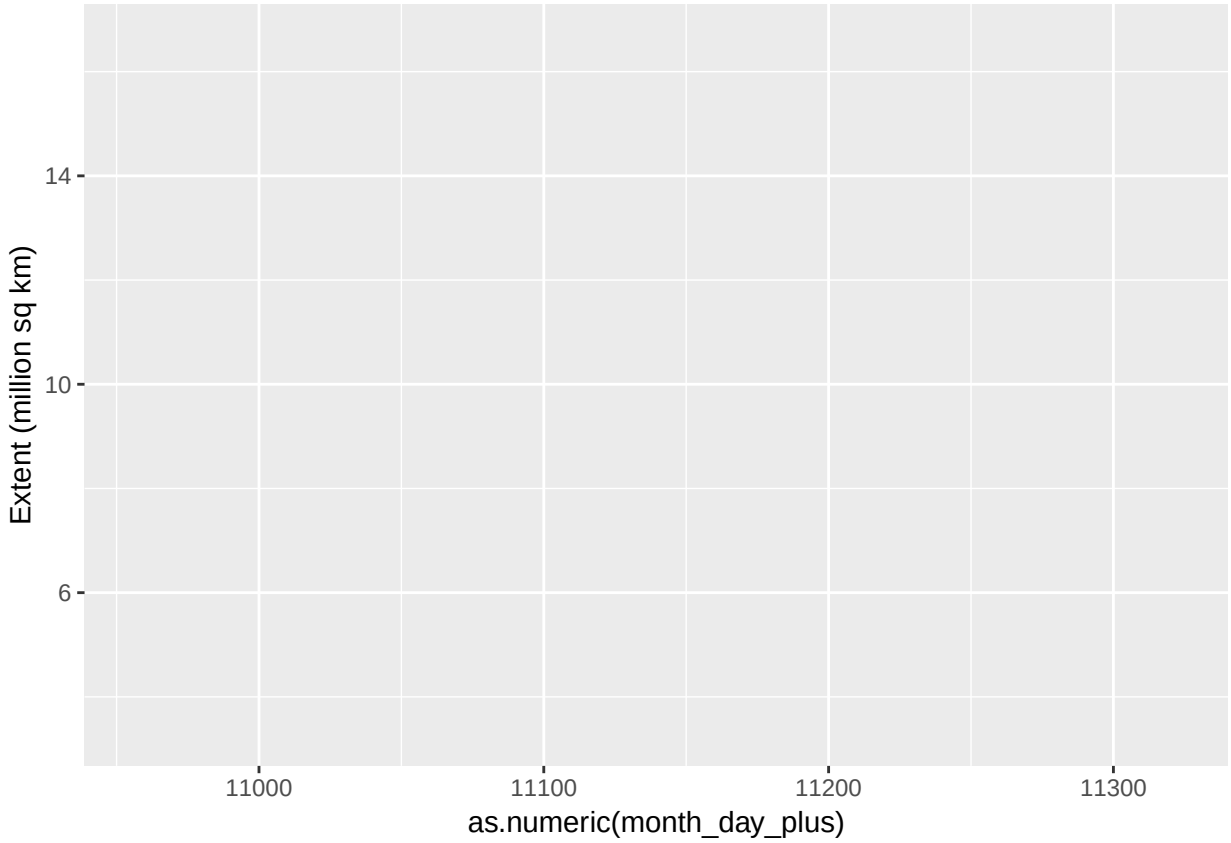
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus))
```



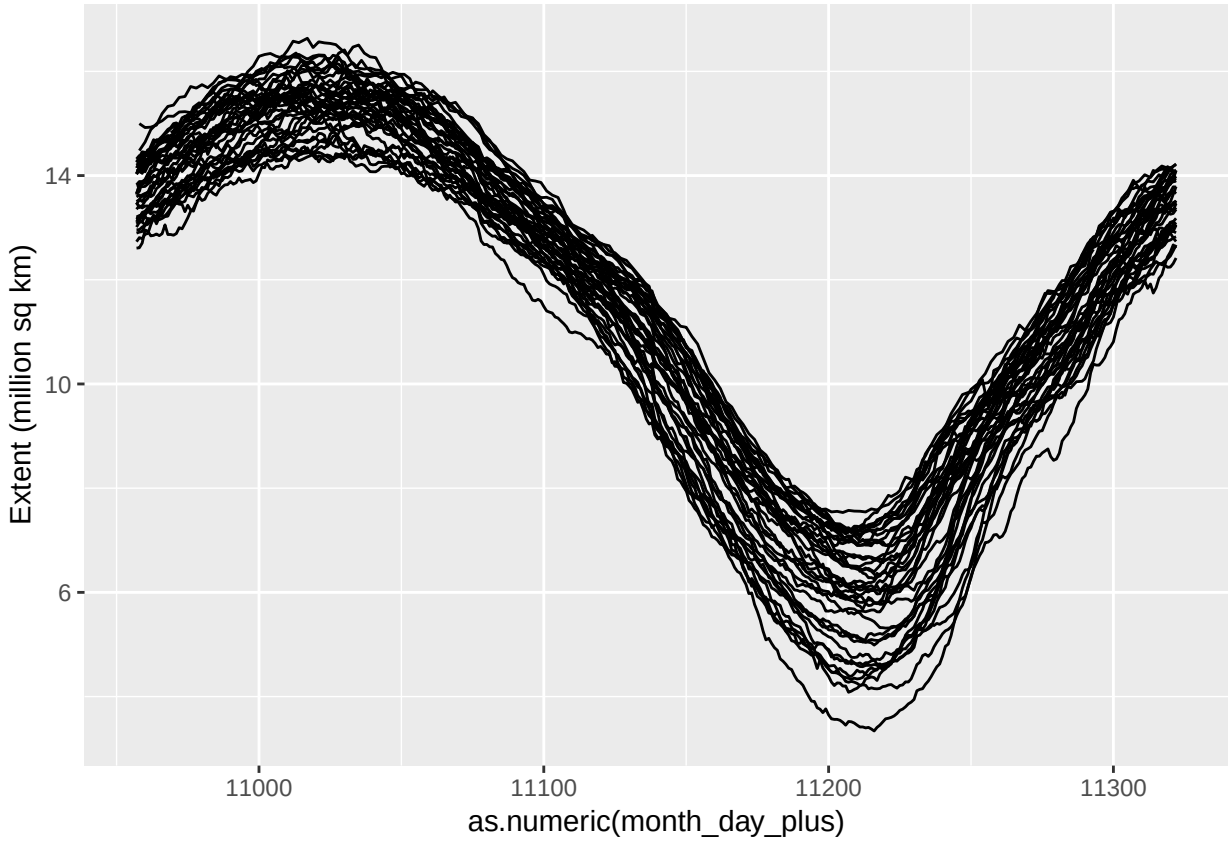
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`)
```



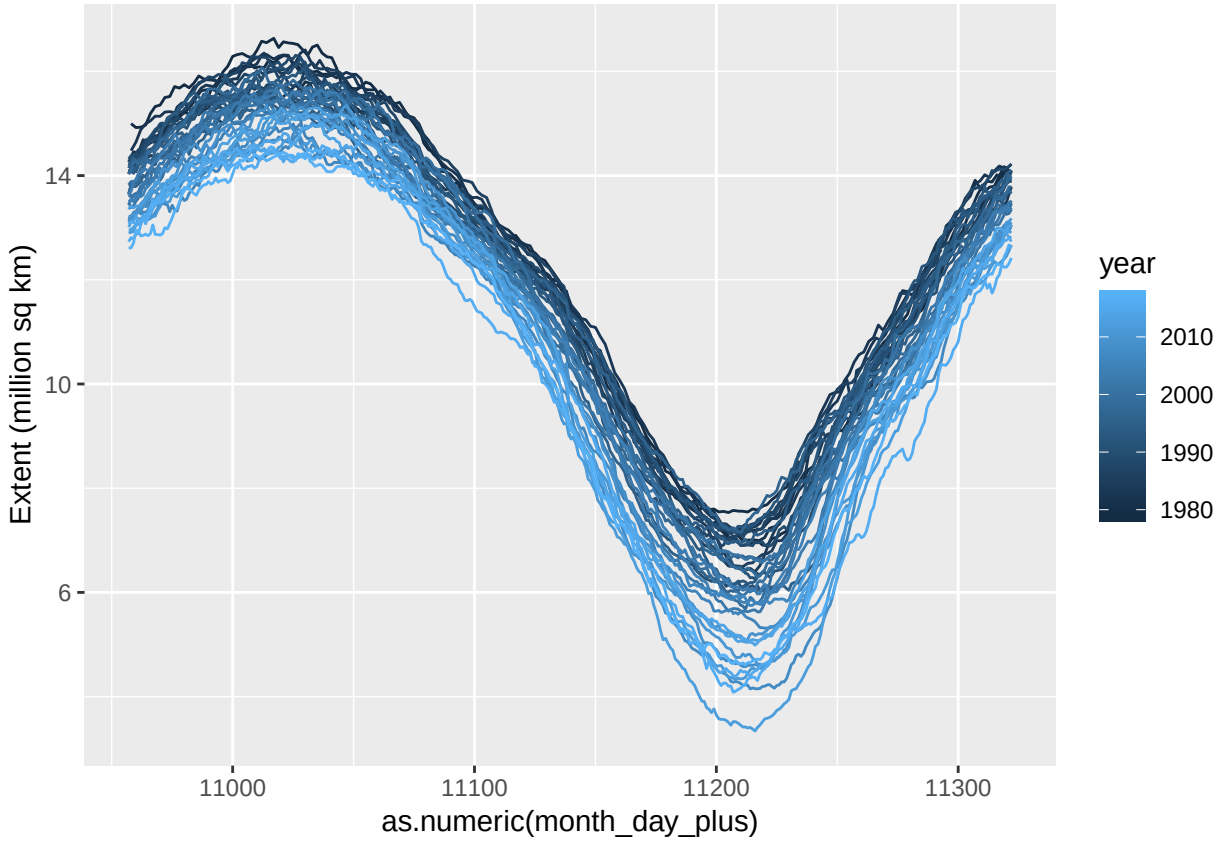
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year)
```



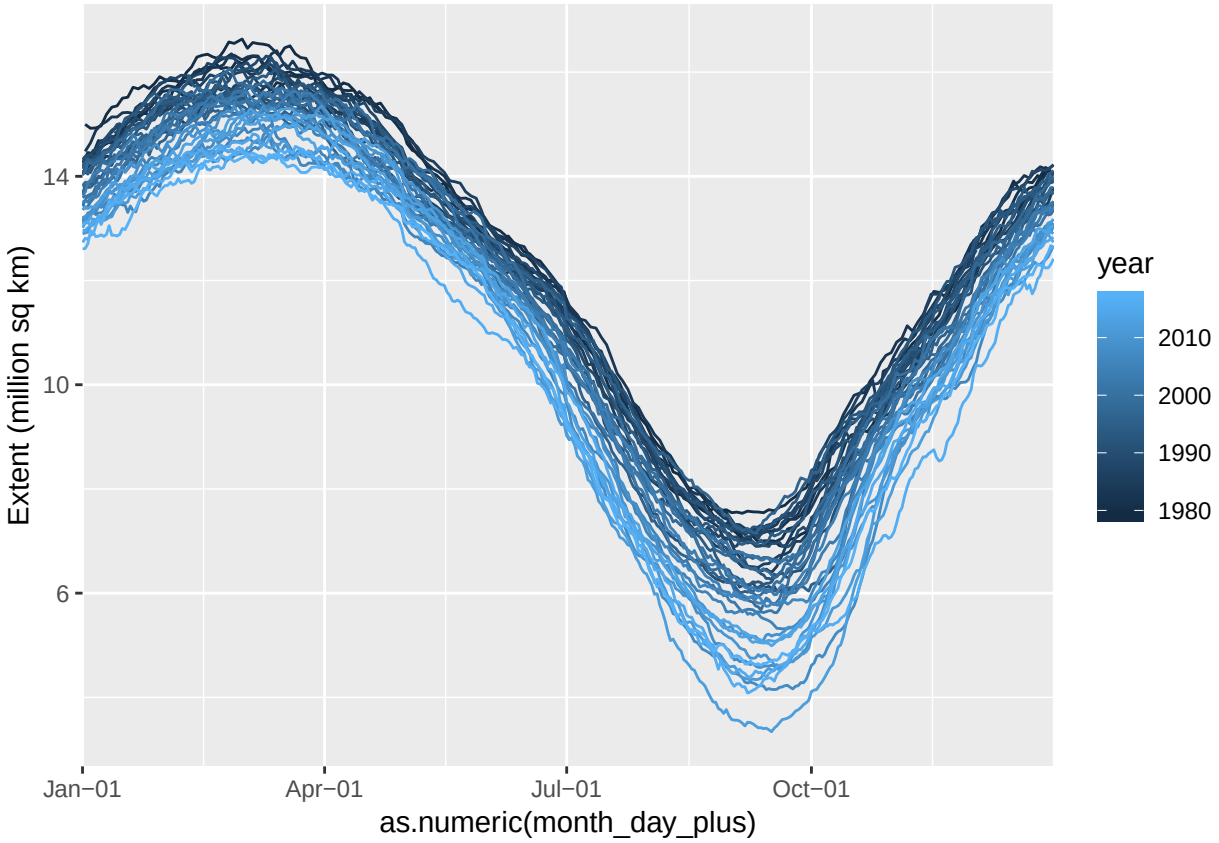
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line()
```



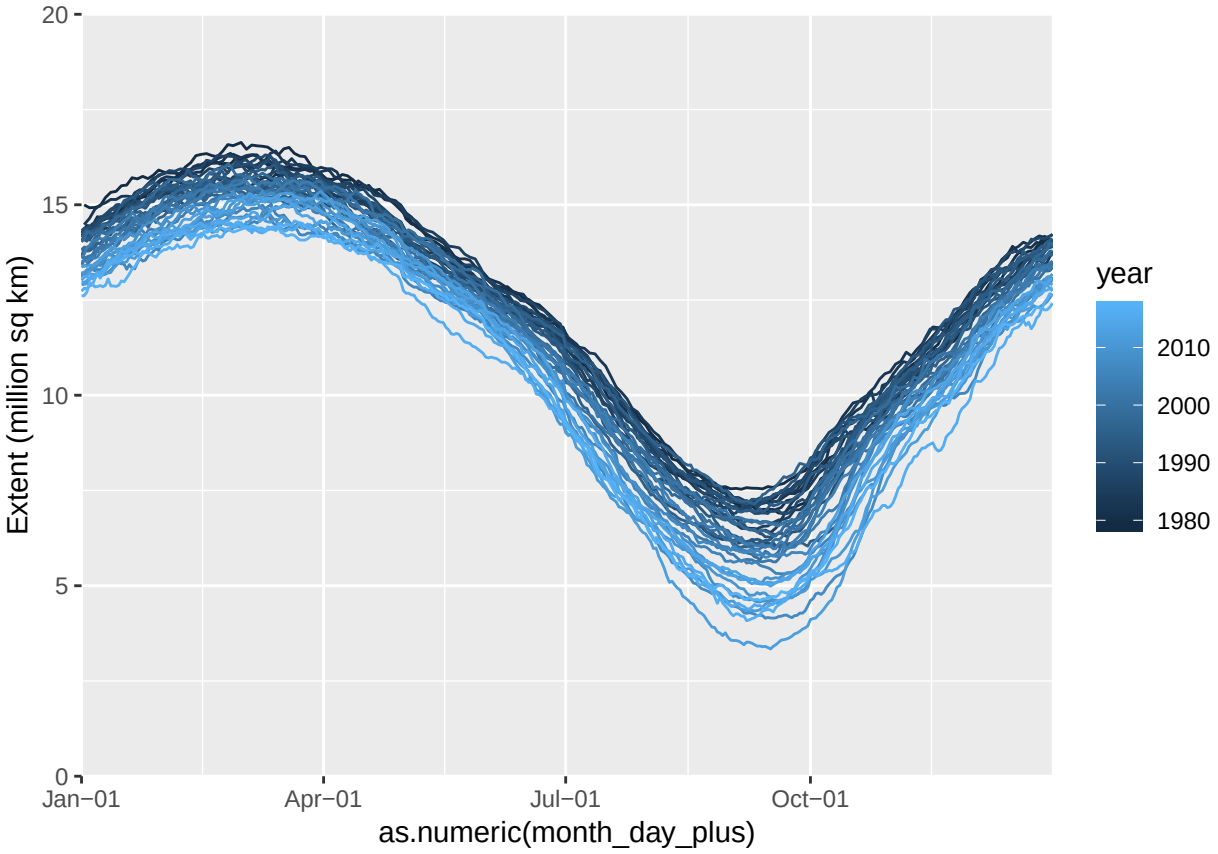
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year)
```



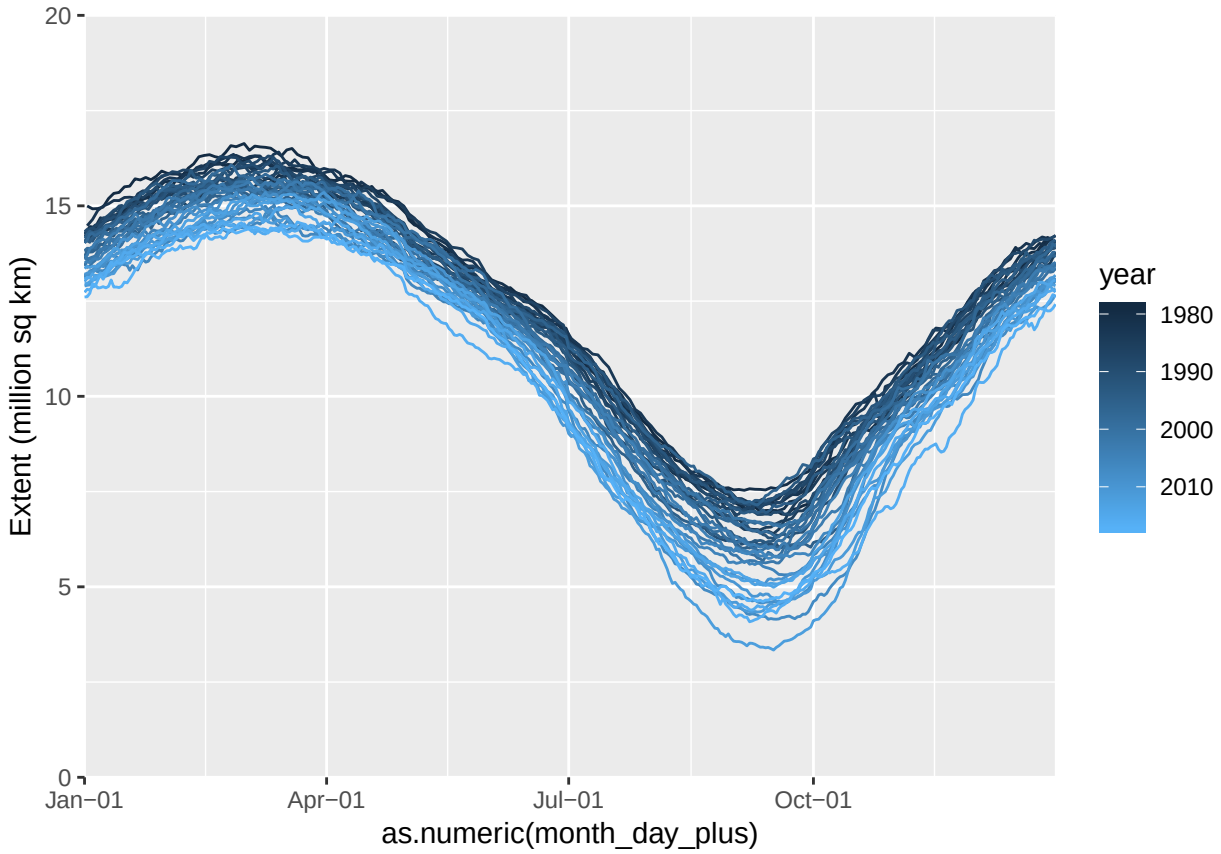
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
```



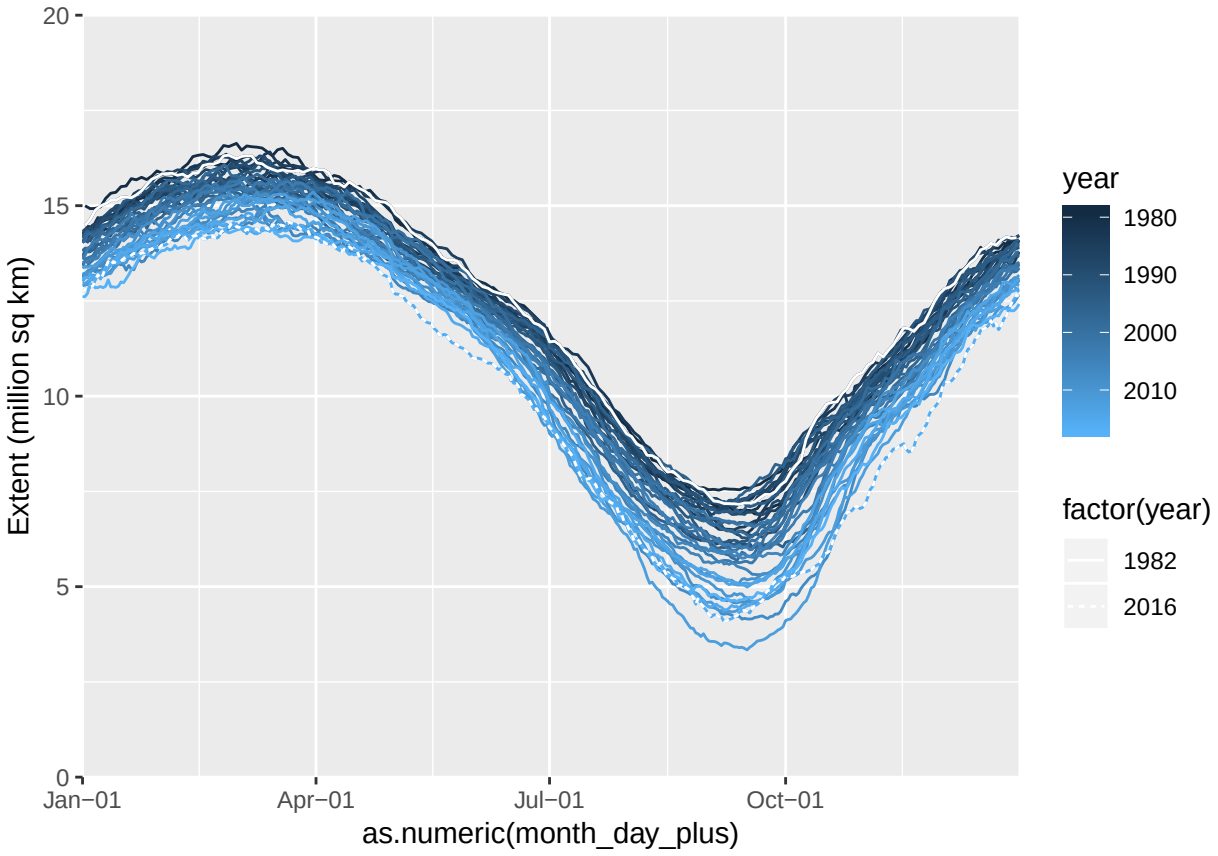
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20))
```



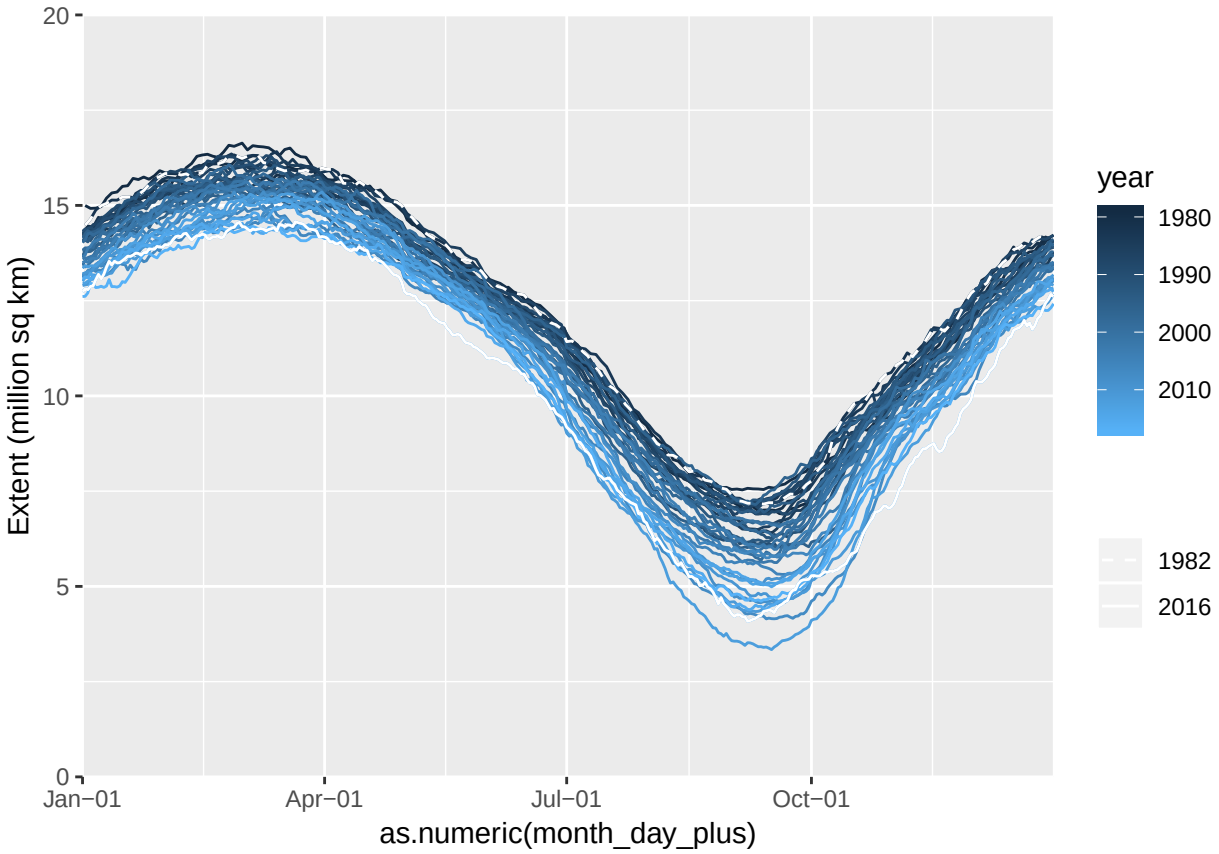
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10))
```



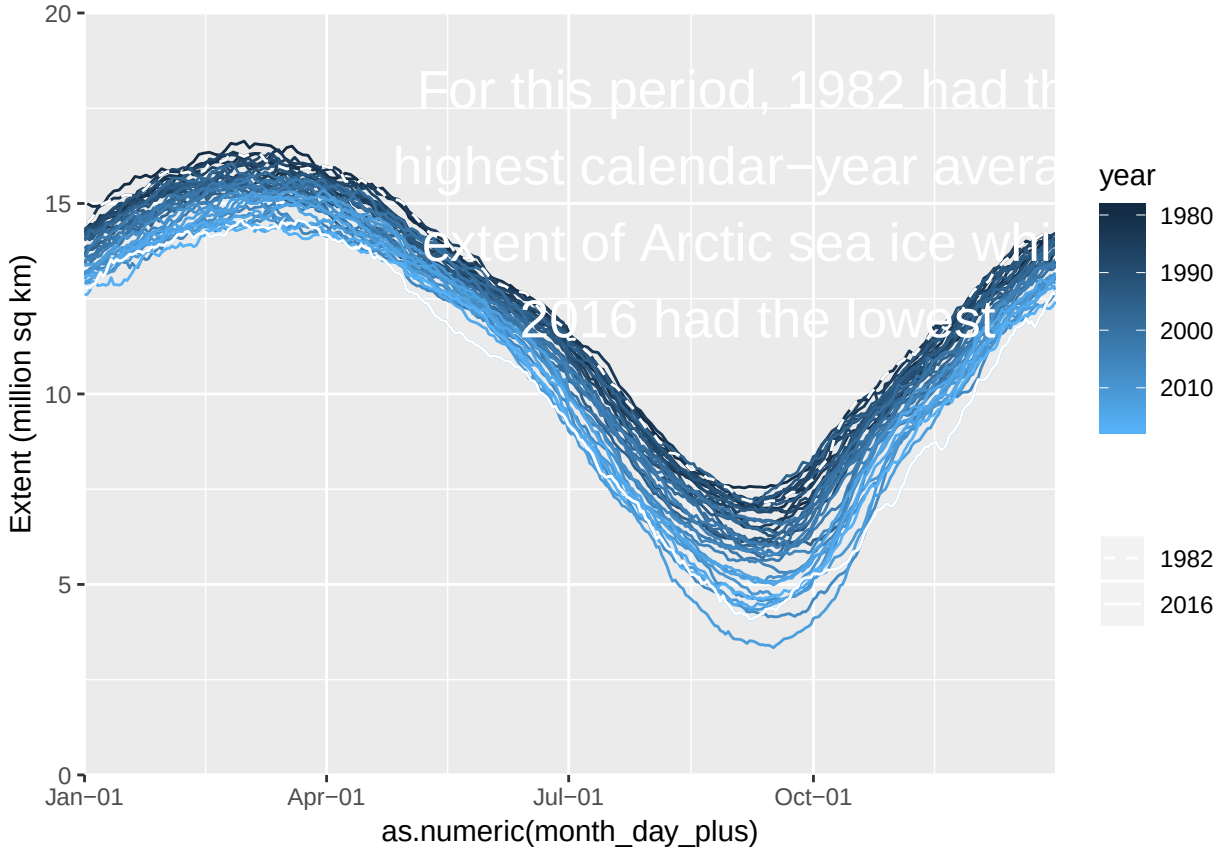
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
```



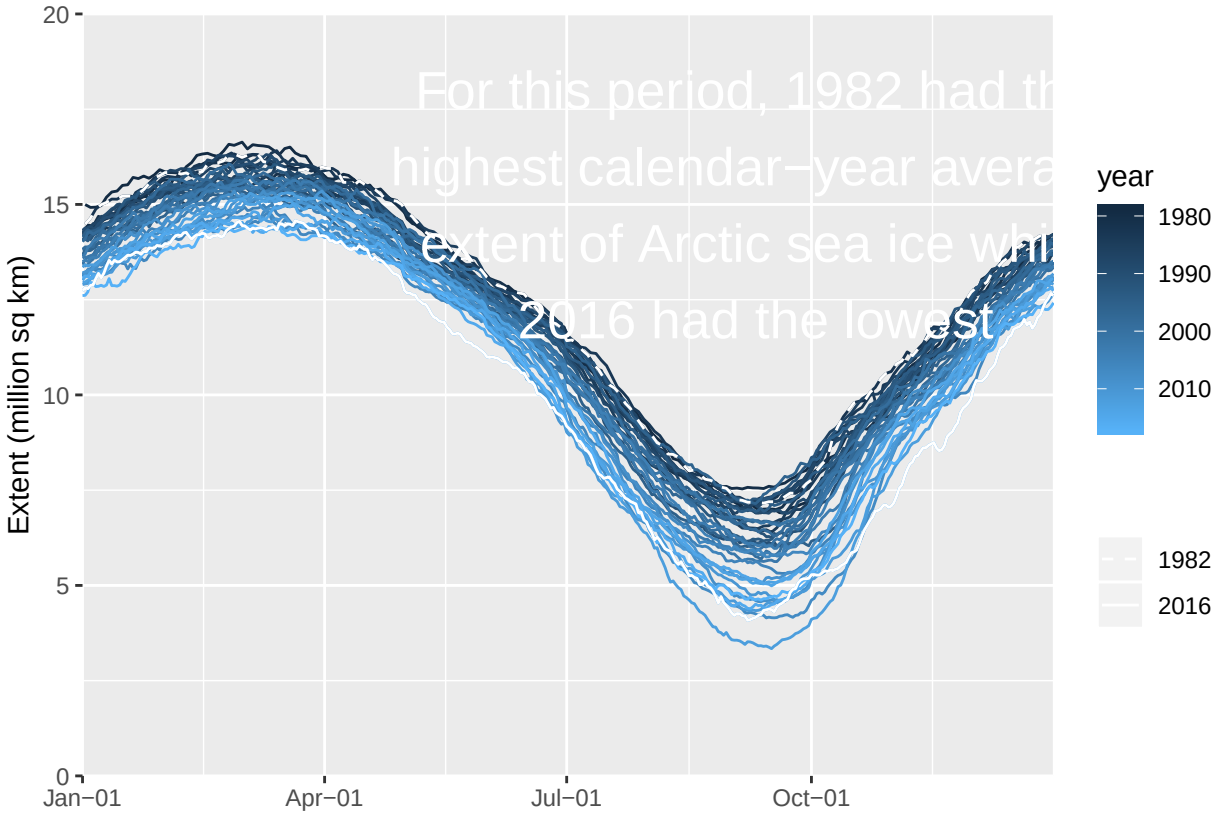
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") )
```



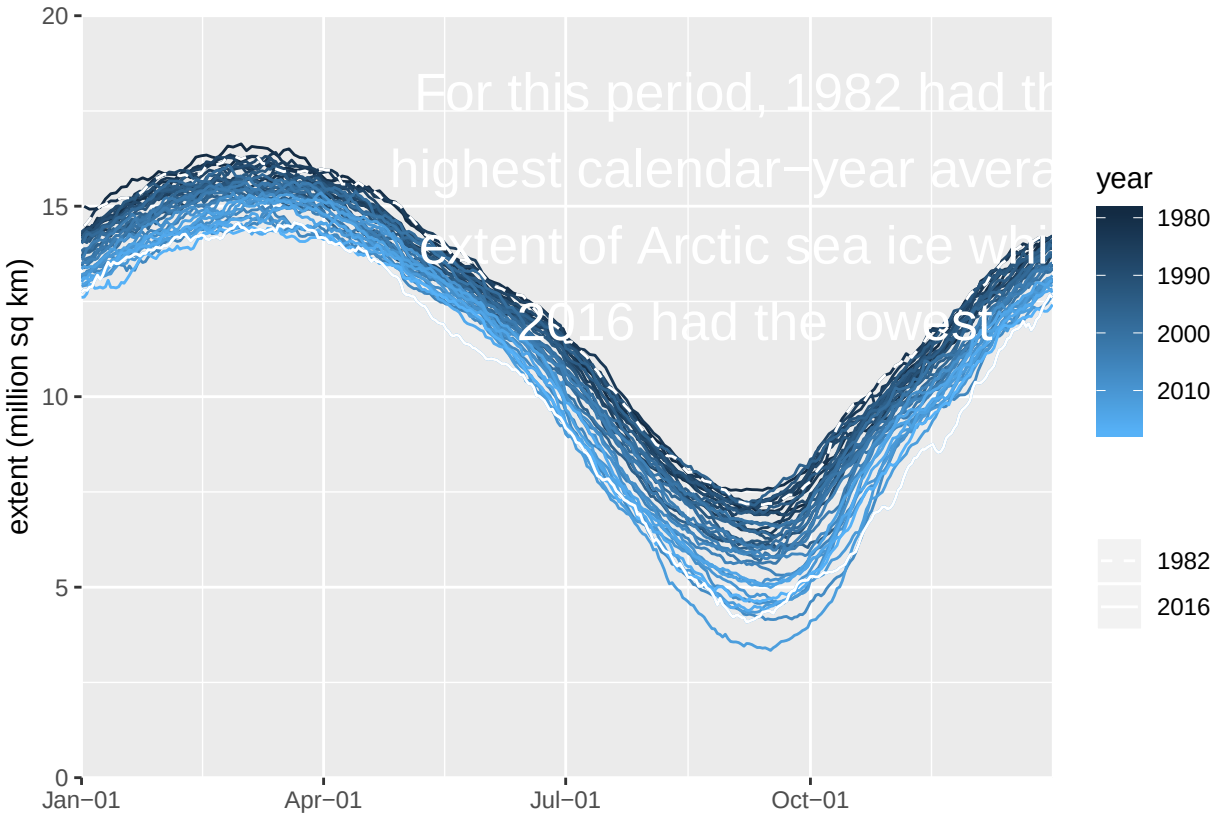
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") ) +
  annotate( geom = "text", x = 11210, y = 15, label = str_wrap("For this period, 1982 had the highest ca
```



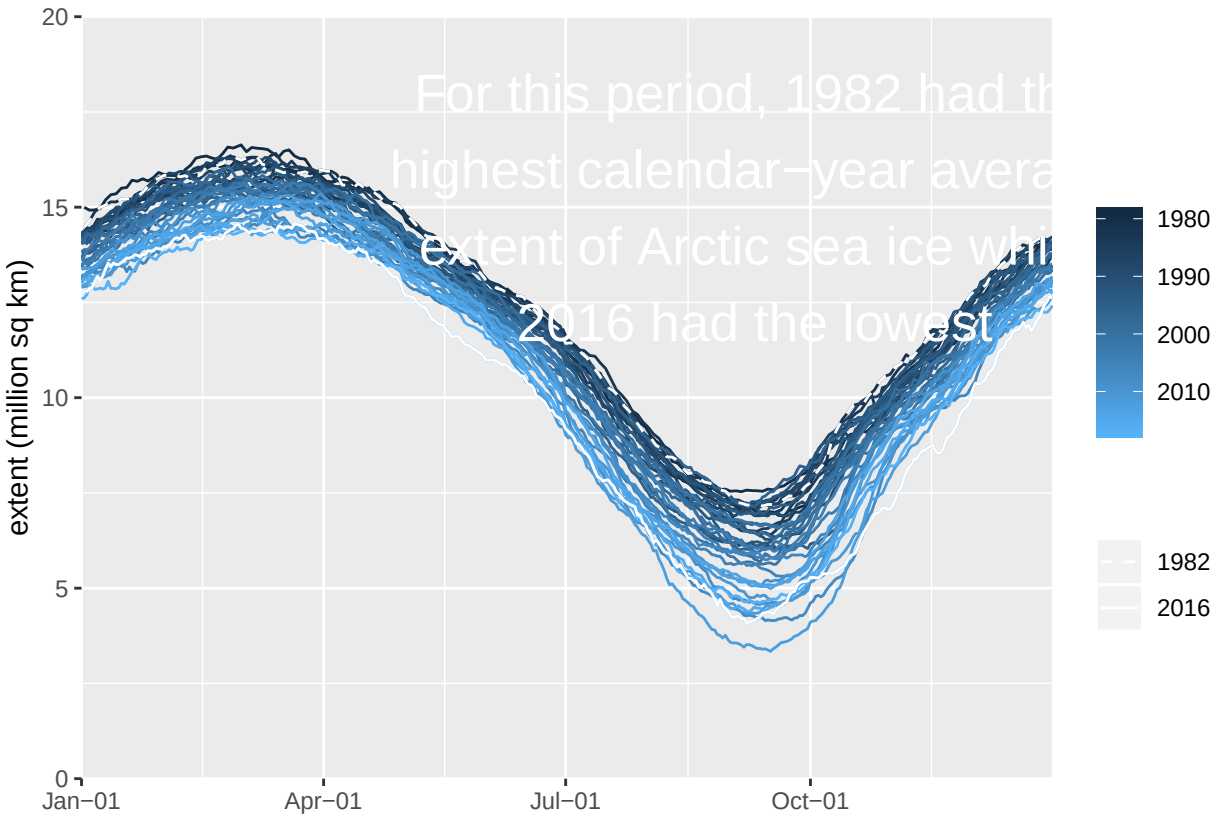
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") ) +
  annotate( geom = "text", x = 11210, y = 15, label = str_wrap("For this period, 1982 had the highest ca
  labs(x = "")
```



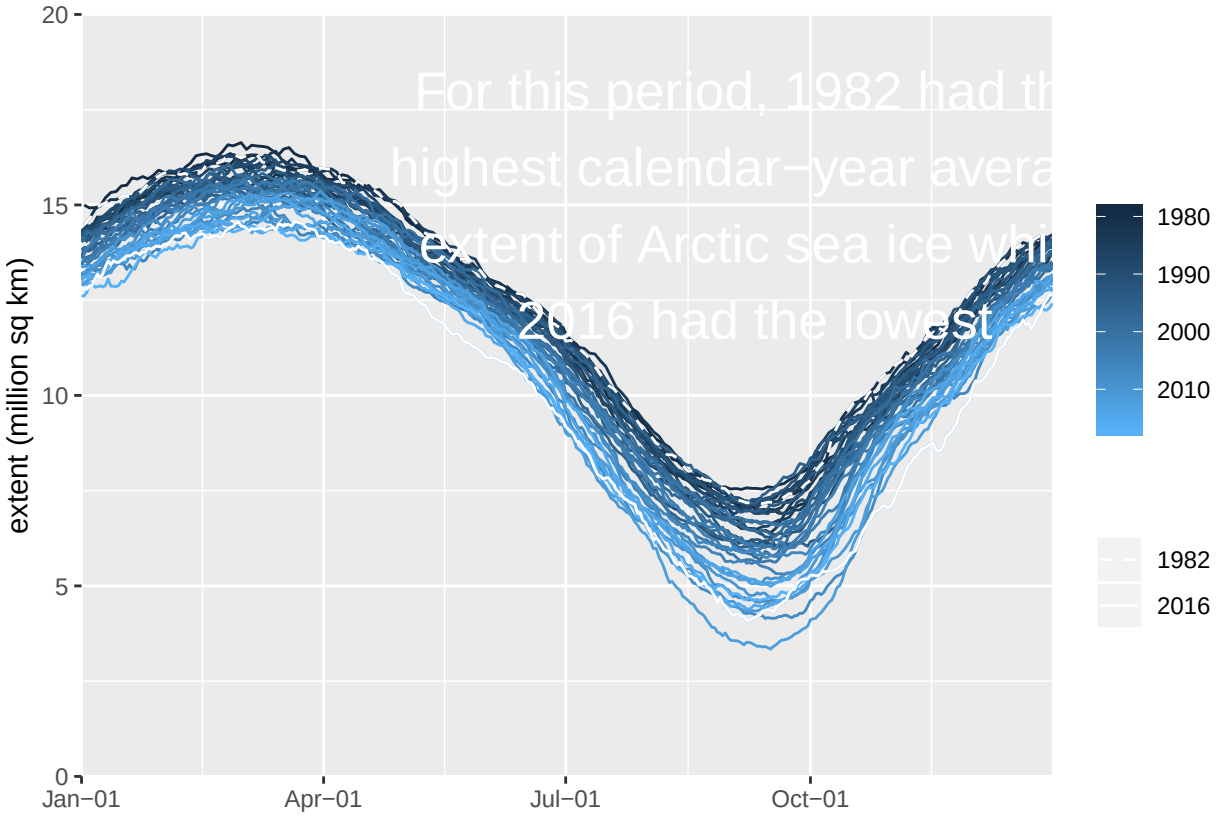
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") ) +
  annotate( geom = "text", x = 11210, y = 15, label = str_wrap("For this period, 1982 had the highest ca
  labs(x = "") +
  labs(y = "extent (million sq km)")
```



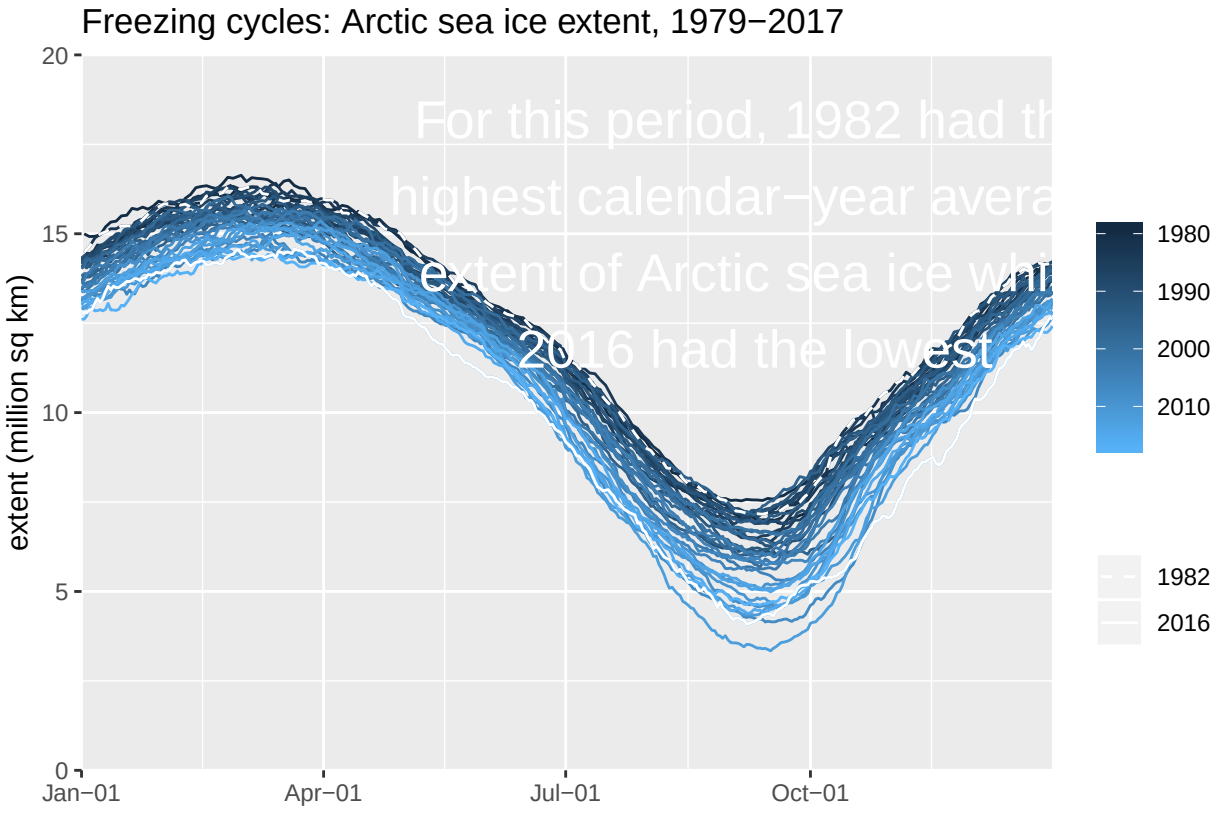
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") ) +
  annotate( geom = "text", x = 11210, y = 15, label = str_wrap("For this period, 1982 had the highest ca
  labs(x = "") +
  labs(y = "extent (million sq km)") +
  labs(col = "")
```



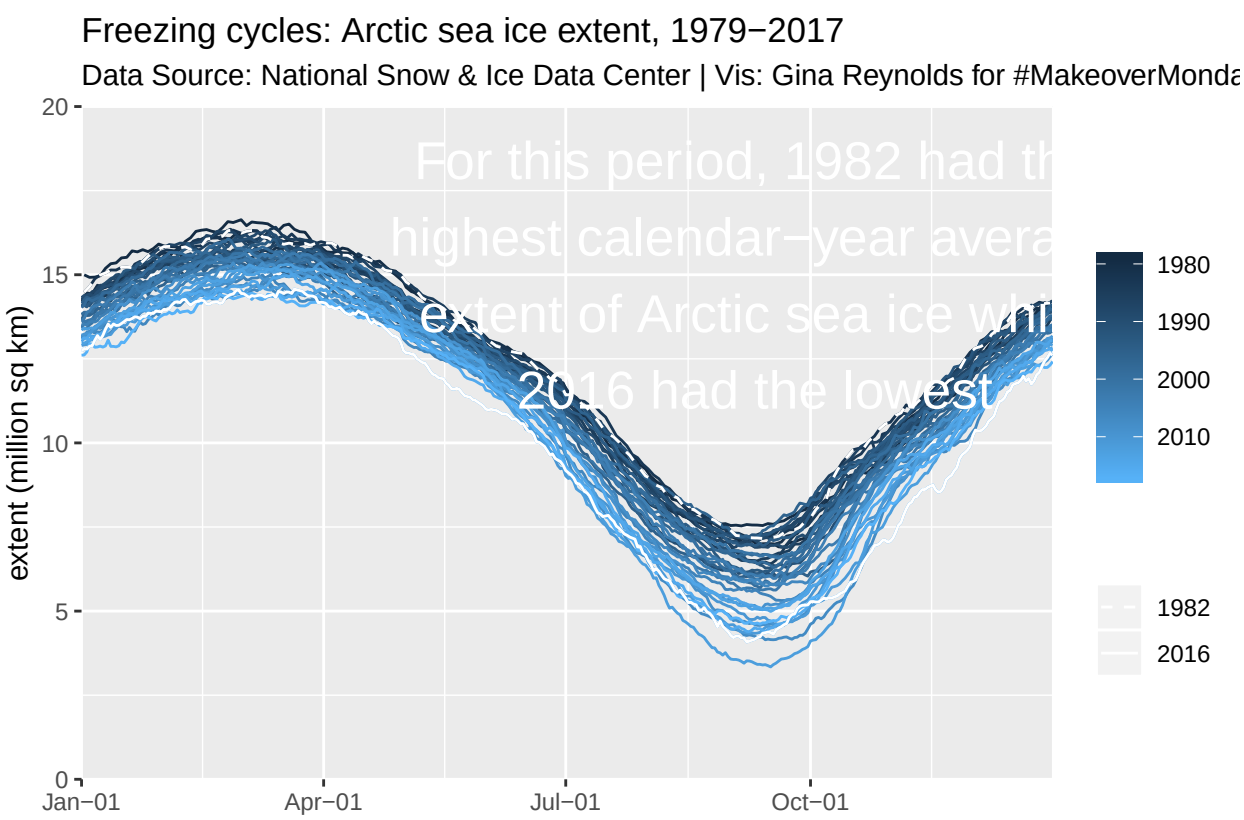
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") ) +
  annotate( geom = "text", x = 11210, y = 15, label = str_wrap("For this period, 1982 had the highest ca
  labs(x = "") +
  labs(y = "extent (million sq km)") +
  labs(col = "") +
  labs(lty = "")
```



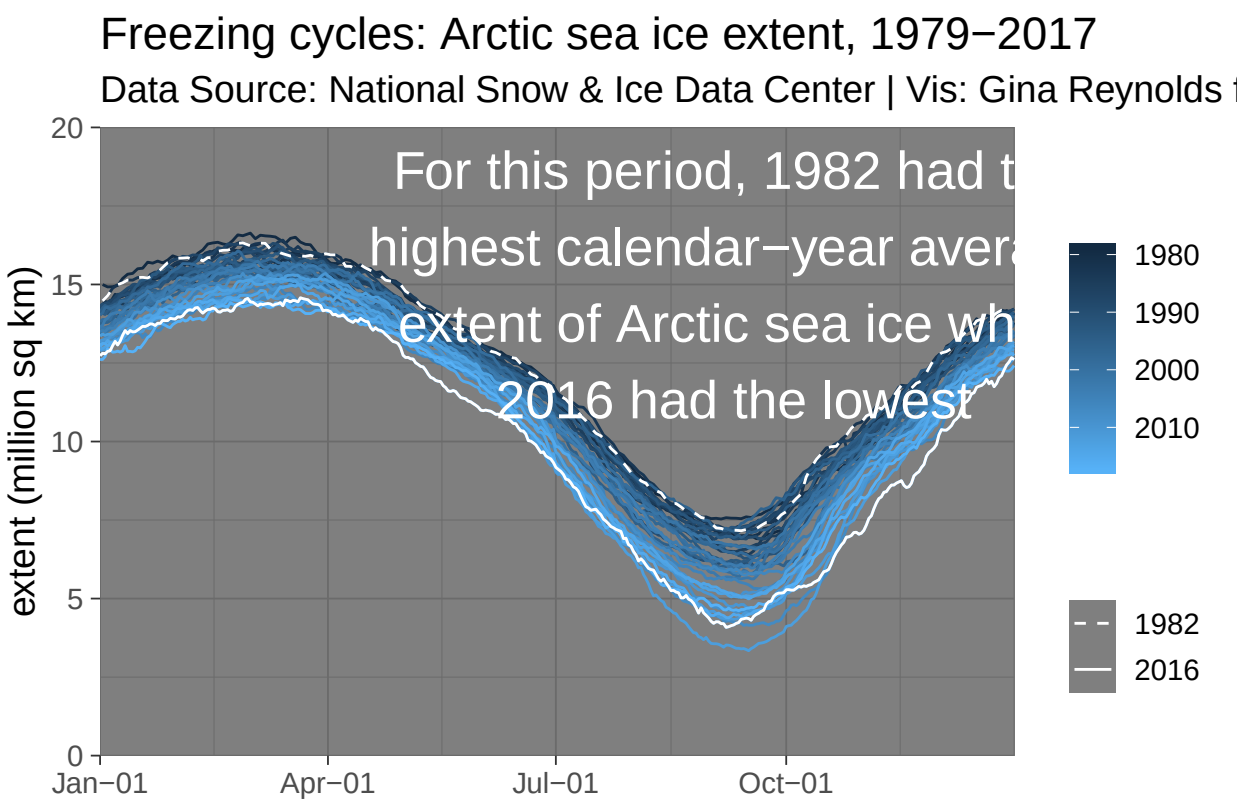
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") ) +
  annotate( geom = "text", x = 11210, y = 15, label = str_wrap("For this period, 1982 had the highest ca
  labs(x = "") +
  labs(y = "extent (million sq km)") +
  labs(col = "") +
  labs(lty = "") +
  labs(title = "Freezing cycles: Arctic sea ice extent, 1979-2017")
```



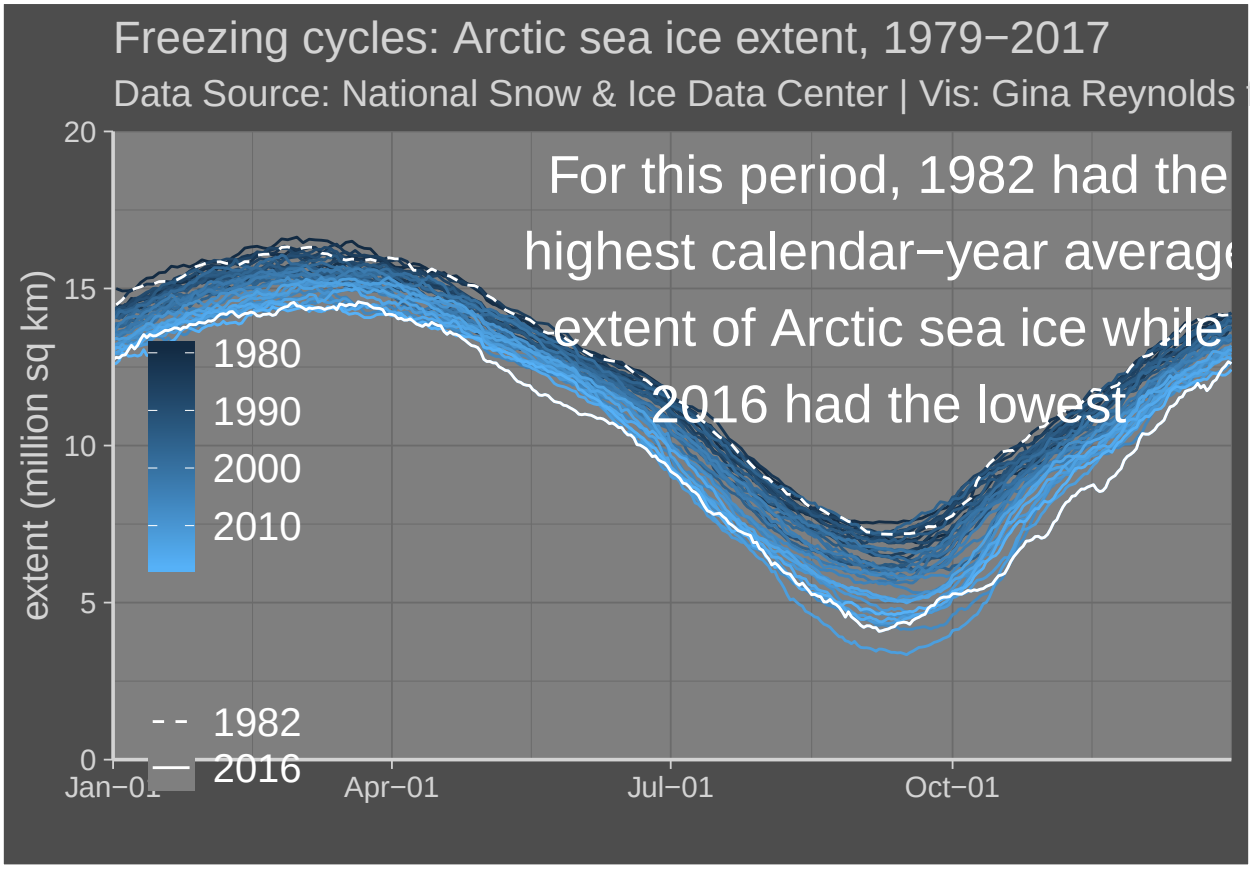
```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") ) +
  annotate( geom = "text", x = 11210, y = 15, label = str_wrap("For this period, 1982 had the highest ca
  labs(x = "") +
  labs(y = "extent (million sq km)") +
  labs(col = "") +
  labs(lty = "") +
  labs(title = "Freezing cycles: Arctic sea ice extent, 1979-2017") +
  labs(subtitle = "Data Source: National Snow & Ice Data Center | Vis: Gina Reynolds for #MakeoverMonday
```



```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") ) +
  annotate( geom = "text", x = 11210, y = 15, label = str_wrap("For this period, 1982 had the highest ca
  labs(x = "") +
  labs(y = "extent (million sq km)") +
  labs(col = "") +
  labs(lty = "") +
  labs(title = "Freezing cycles: Arctic sea ice extent, 1979-2017") +
  labs(subtitle = "Data Source: National Snow & Ice Data Center | Vis: Gina Reynolds for #MakeoverMonday
  theme_dark(base_size = 14)
```



```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") ) +
  annotate( geom = "text", x = 11210, y = 15, label = str_wrap("For this period, 1982 had the highest ca
  labs(x = "") +
  labs(y = "extent (million sq km)") +
  labs(col = "") +
  labs(lty = "") +
  labs(title = "Freezing cycles: Arctic sea ice extent, 1979-2017") +
  labs(subtitle = "Data Source: National Snow & Ice Data Center | Vis: Gina Reynolds for #MakeoverMonday
  theme_dark(base_size = 14) +
  theme_opts
```



```
ggplot(df) +
  aes(x = as.numeric(month_day_plus)) +
  aes(y = `Extent (million sq km)`) +
  aes(group = year) +
  geom_line() +
  aes(col = year) +
  scale_x_continuous(breaks = br, labels = c("Jan-01", "Apr-01", "Jul-01", "Oct-01", "Jan-01"), expand =
  scale_y_continuous(expand = c(0, 0), limits = c(0, 20)) +
  scale_color_continuous(guide = guide_colourbar(reverse = TRUE), breaks = seq(2010, 1980, -10)) +
  geom_line(aes(lty = factor(year)), data = df %>% filter(year == 2016 | year == 1982),  col = "white")
  scale_linetype_manual( name = "", values = c("dashed", "solid") ) +
  annotate( geom = "text", x = 11210, y = 15, label = str_wrap("For this period, 1982 had the highest ca
  labs(x = "") +
  labs(y = "extent (million sq km)") +
  labs(col = "") +
  labs(lty = "") +
  labs(title = "Freezing cycles: Arctic sea ice extent, 1979-2017") +
  labs(subtitle = "Data Source: National Snow & Ice Data Center | Vis: Gina Reynolds for #MakeoverMonday
  theme_dark(base_size = 14) +
  theme_opts
```