

Slow ggplot2

Evangeline Reynolds

2018-11-02

Contents

1	Introduction	5
2	Baseball, WAR, and Ethnicity	7
3	Christmas Trees	19
4	Officials' beliefs about women's representation	23
5	Maternal Leave	45
6	Traits	47
7	Salaries of Trump and Obama White House Employees	49
8	Winter Games	51
9	Brexit	55
10	Curry in London	59
11	Life Expectancy Increases	63
12	Myers Briggs	65
13	Wine	67
14	Arctic Ice	69

Chapter 1

Introduction

In late 2016 I heard Andy Kreibel (and ____) being interviewed on the Podcast “Data Stories”. He was invited onto the show because he had started the initiative #MakeoverMonday, in which he would find the data for an existing data visualization that he had come across in the media, and would recreate the visualization using Tableau, a tool that they were both expert in, but found that they weren’t practicing as regularly due to the grind of administrative jobs. The two friends shared their makeovers products with each other, but also with the world on Twitter. Soon, more people expressed interest in joining, and they two started a more organized initiative - posting the original graph and data every Sunday, so that whoever wanted to could participate in #Makeovermonday.

My first submission was late 2016, after catching wind of the exciting project via the podcast. I made a scrappy little graph about motorway casualties; sad topic, but fun graph making.

I was using base R at that time. Then in the summer of 2017 I went to a conference in Zurich, the women’s summer school for political methodology. There was a session on ggplot2. I internalized some of the basics, and decided that if I wanted to learn that (powerful - as everyone kept calling it) graphing system, then I could do it via the #MakeoverMonday weekly exercises (not that I participated weekly). Even though most folks were using Tablaeu, the administrators didn’t seem to mind a few R and ggplot submission here and there. I got a little hooked.

Early this year Andy and Eva Murry sent a number of the participants a private message on Twitter. “We’re writing a book: #MakeoverMonday”. They were putting together a collection of a visualizations that resulted from the project, and were seeking perspectives of participants as well as permission to use some of the visualizations produced for the initiatives. Cool. I was pleased to participate. For me #MakeoverMonday allowed me to focus on the visualization task. Usually visualization comes at the end of, sometimes arguous, data cleaning — and you might already be a little spent. Having rather clean data delivered, and seeing the approaches of many other (many brilliant) data visualizers was a treat. I still need to buy my copy of the book, which contains a visualization of food prices in London as a function of how far a restaurant is from the Big Ben.

And now, using the magic of RStudio and Yihui Xie’s bookdown, I’m putting together my own little collection. Of course there is a bit of curation involved — I’m not including every plot. And, I’m revising the exact code that creates the plots in many cases, to be more consistent across plots, and also, I think, to make communicating about how the plot was built easy. This involves:

- using fewer functions (`labs(title =)` instead of `ggtitle()`)
- pulling out `aes()` from the `ggplot()` function
- multiply using functions `aes(x = ?) + aes(y = ?)` rather than `aes(x = , y =)`
- 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

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:

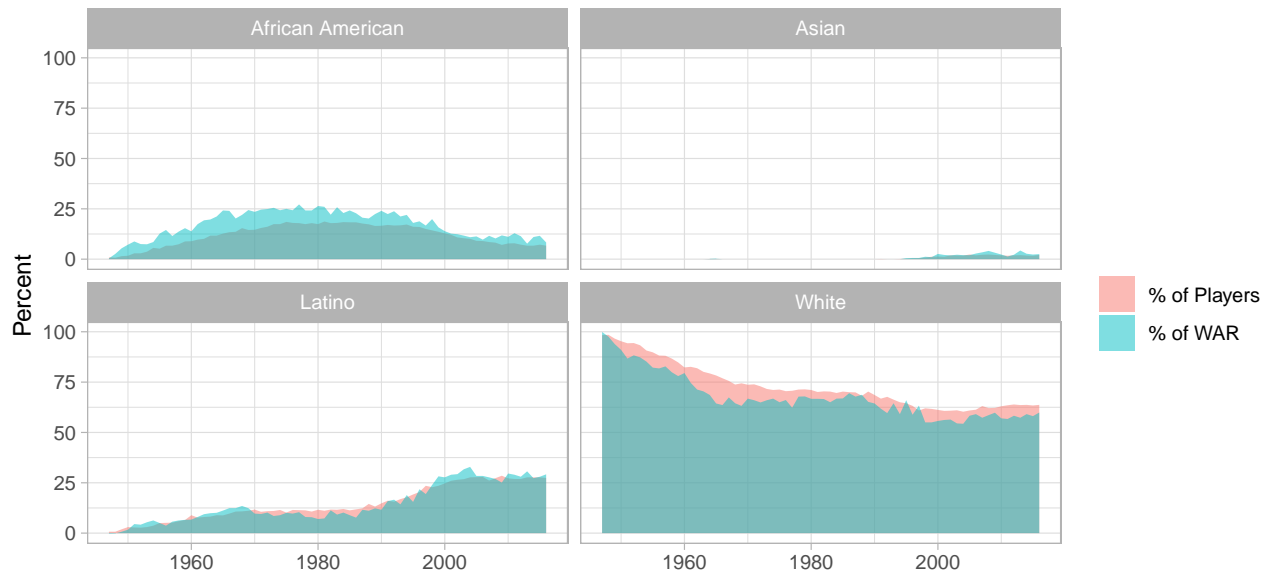
Year	Ethnicity	type	Percent
1993	Asian	% of Players	0.0
1988	Latino	% of WAR	11.1
1965	African American	% of Players	12.7
1998	White	% of Players	62.0
1948	Asian	% of Players	0.0

```
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



class: split-50 count: false

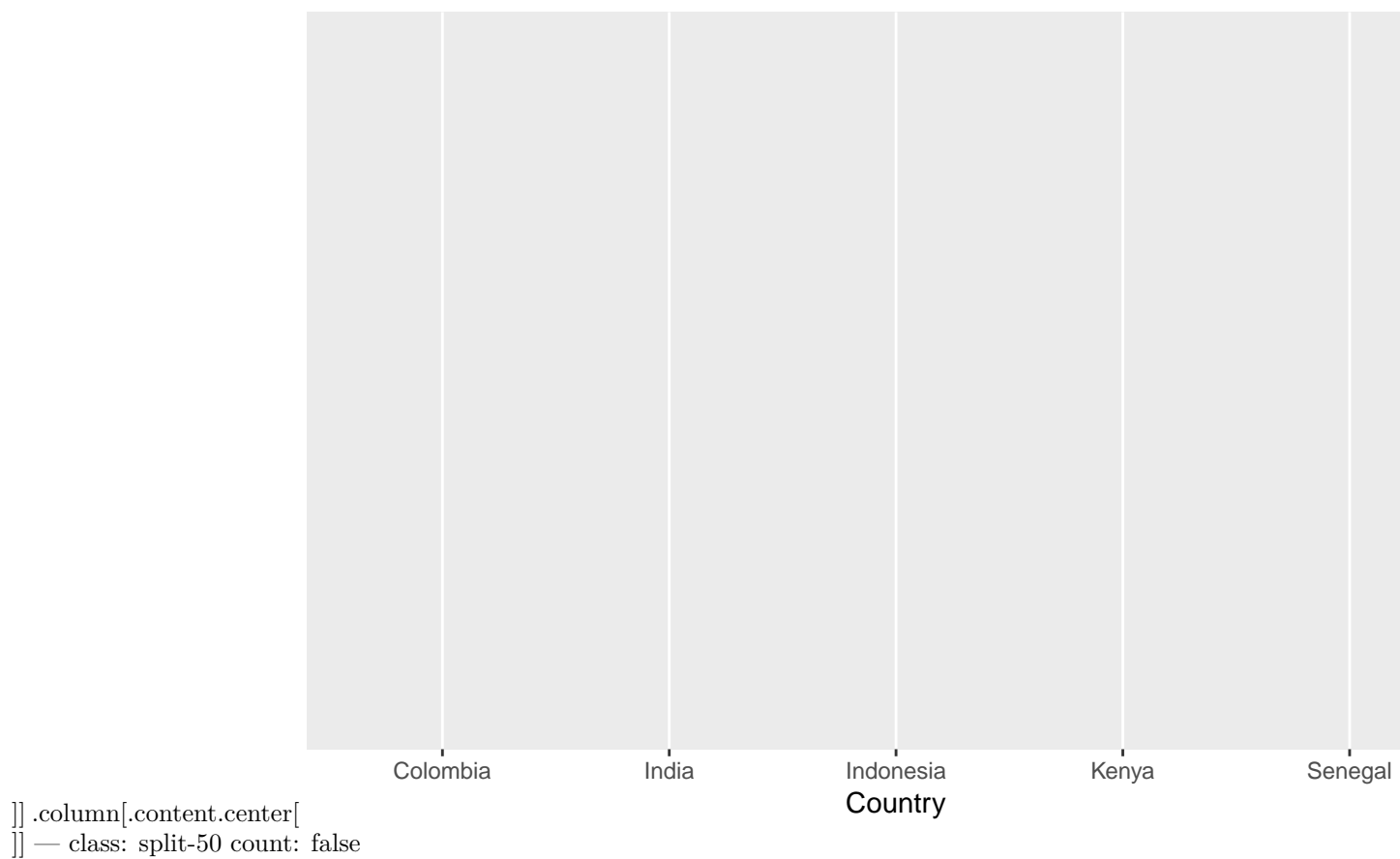
.column[.content[

ggplot(df_gather) #<<

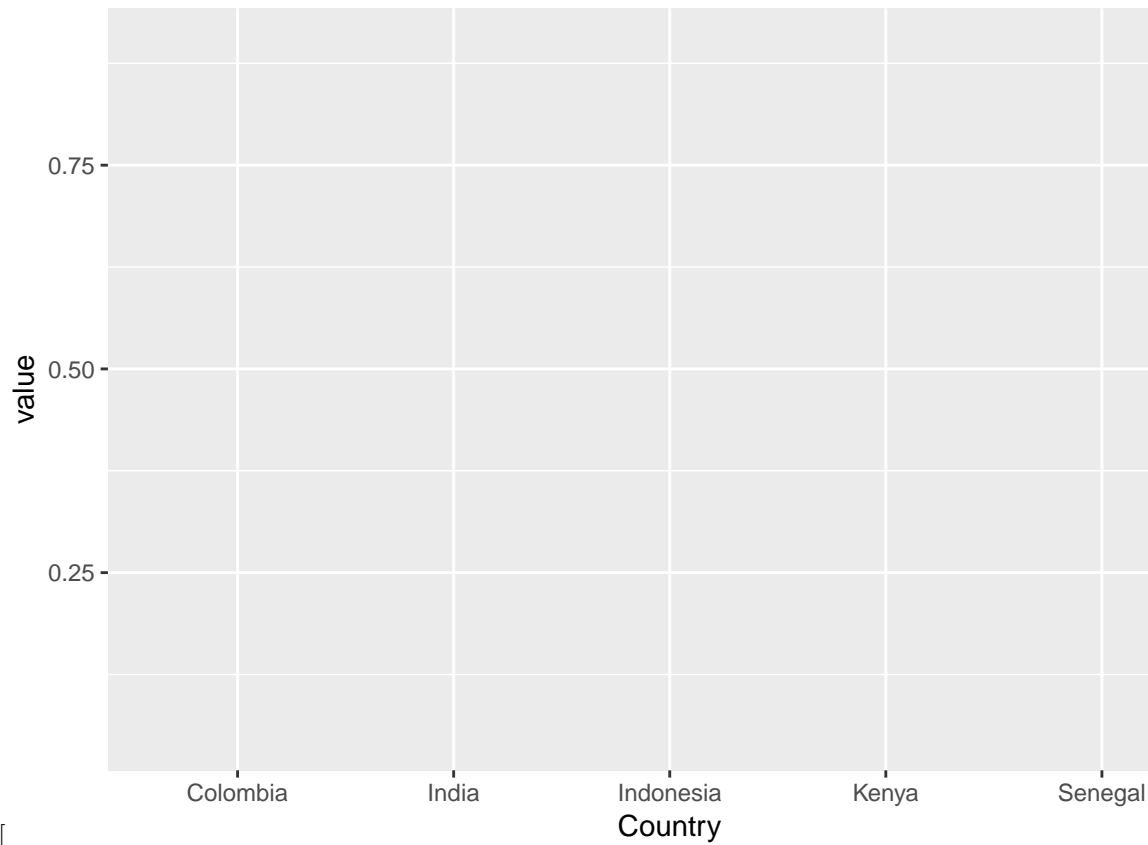
]] .column[.content.center[

]] — class: split-50 count: false


```
.column[.content[
ggplot(df_gather) +
  aes(x = Year) #<<
```

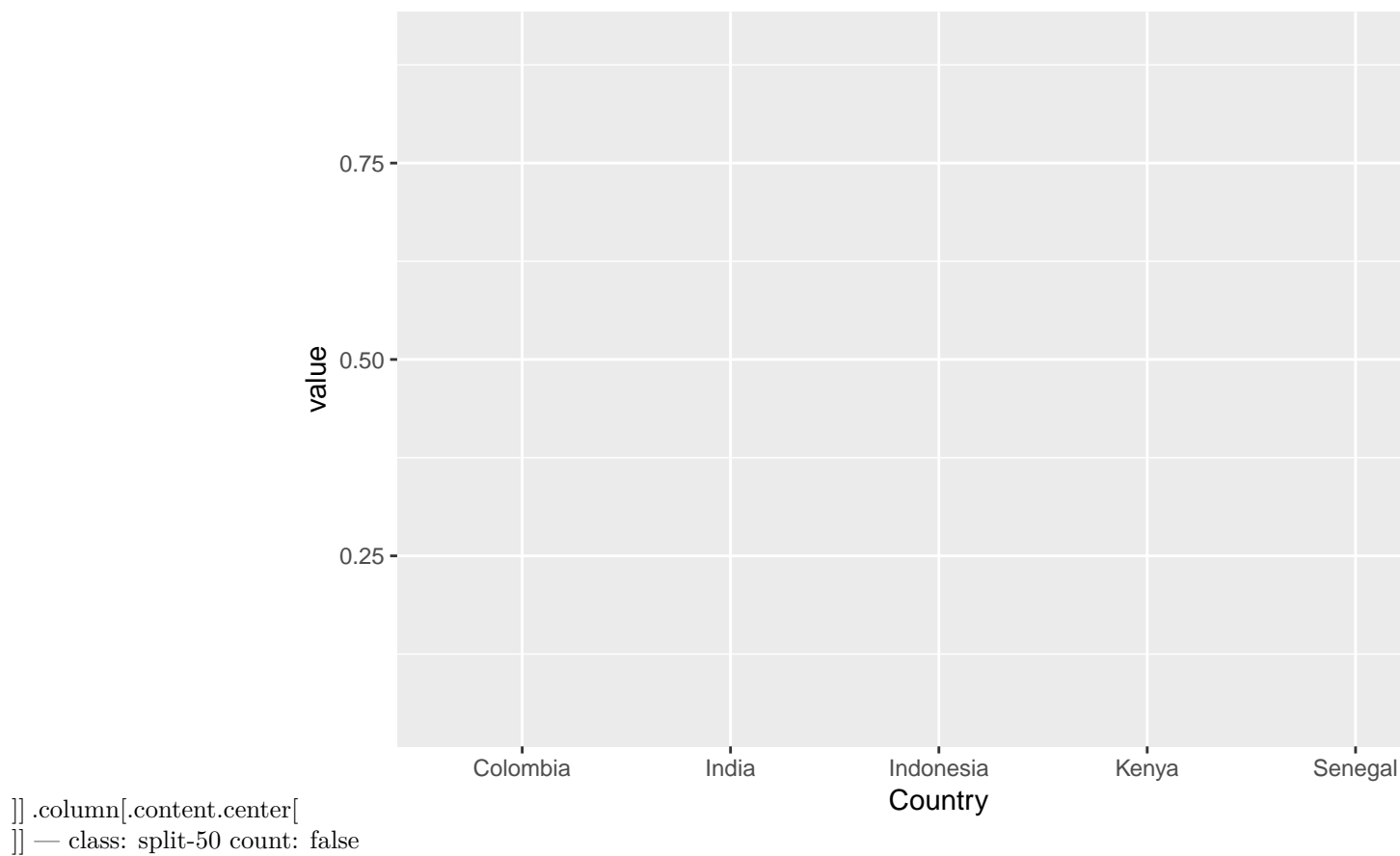


```
.column[.content[
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) #<<
```

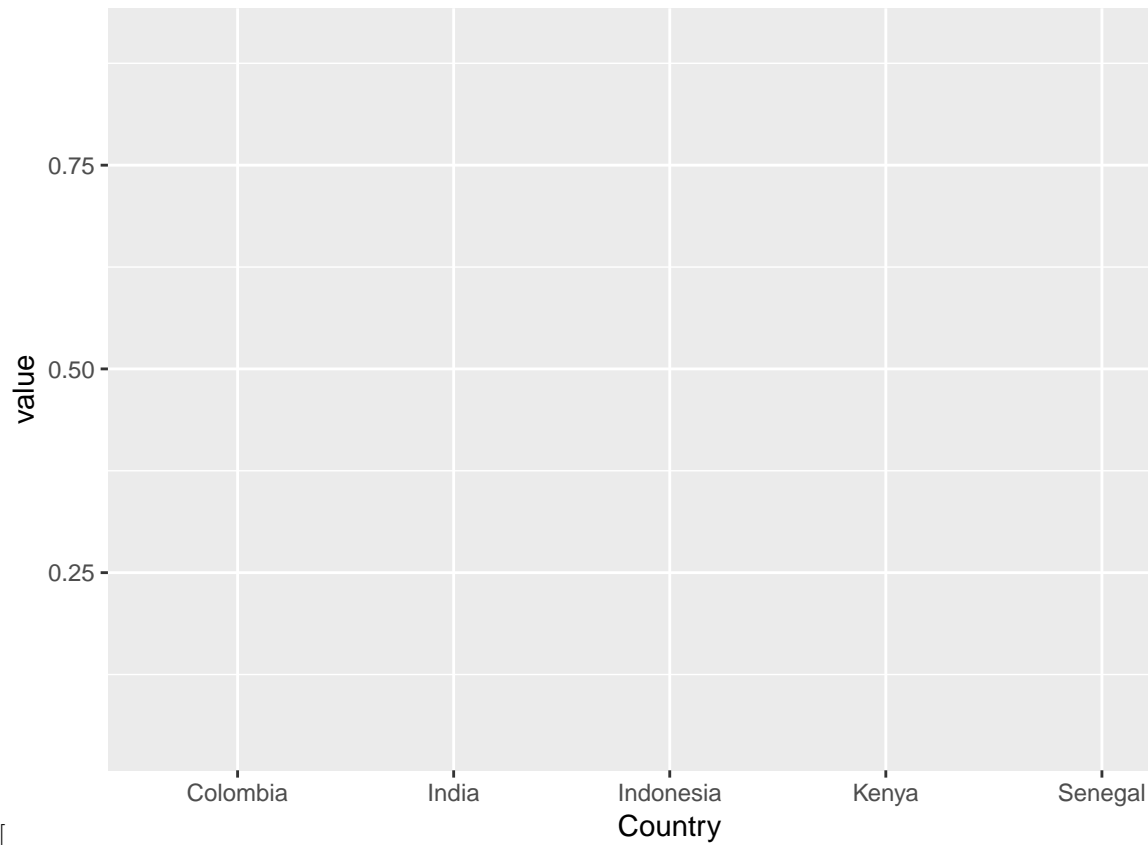


```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type) #<<
```

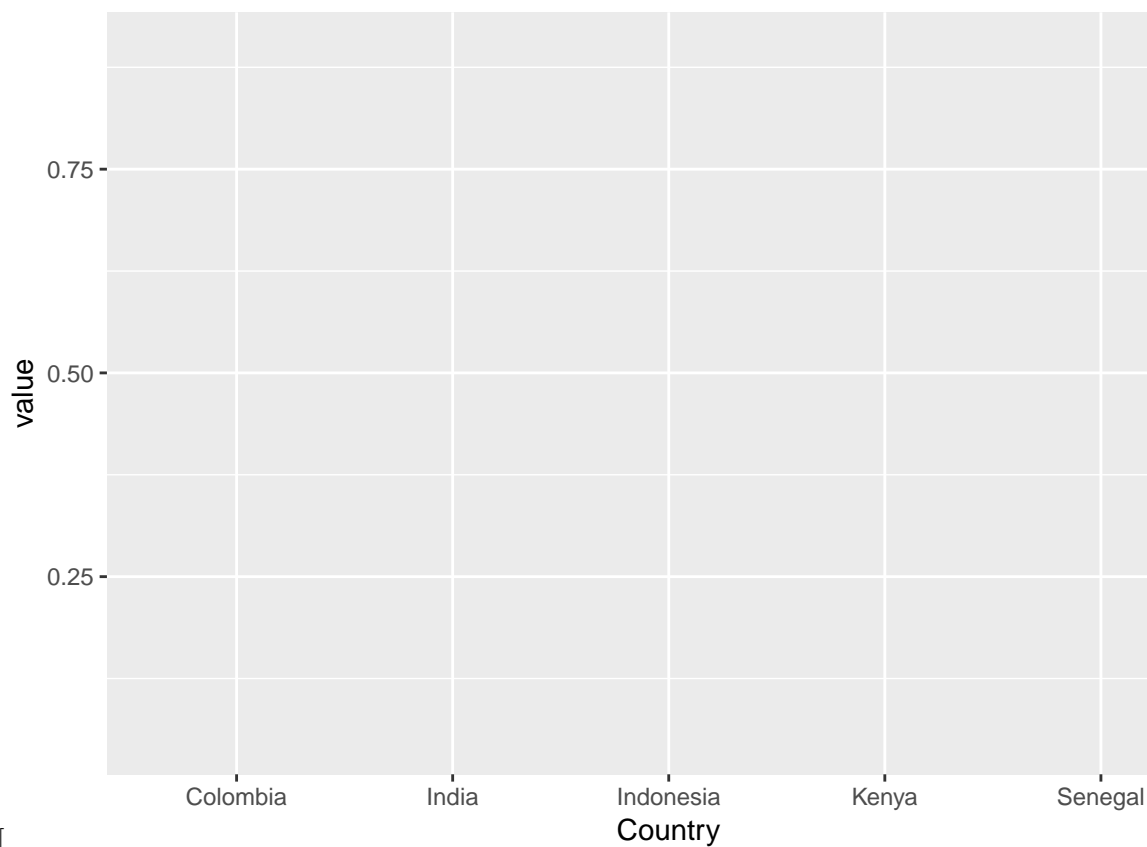


```
.column[.content[
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type) +
  facet_wrap(~ Ethnicity) #<<
```



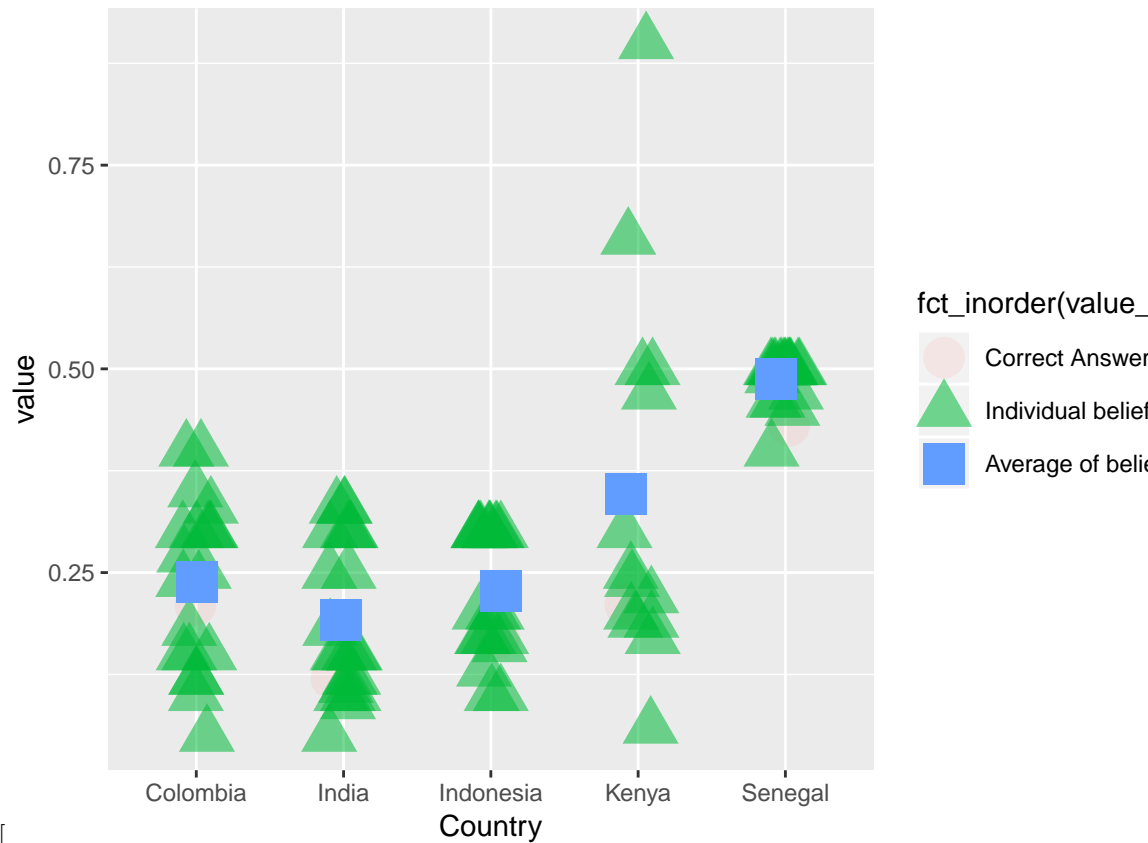
```
]] .column[.content.center[
]] — class: split-50 count: false
```

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



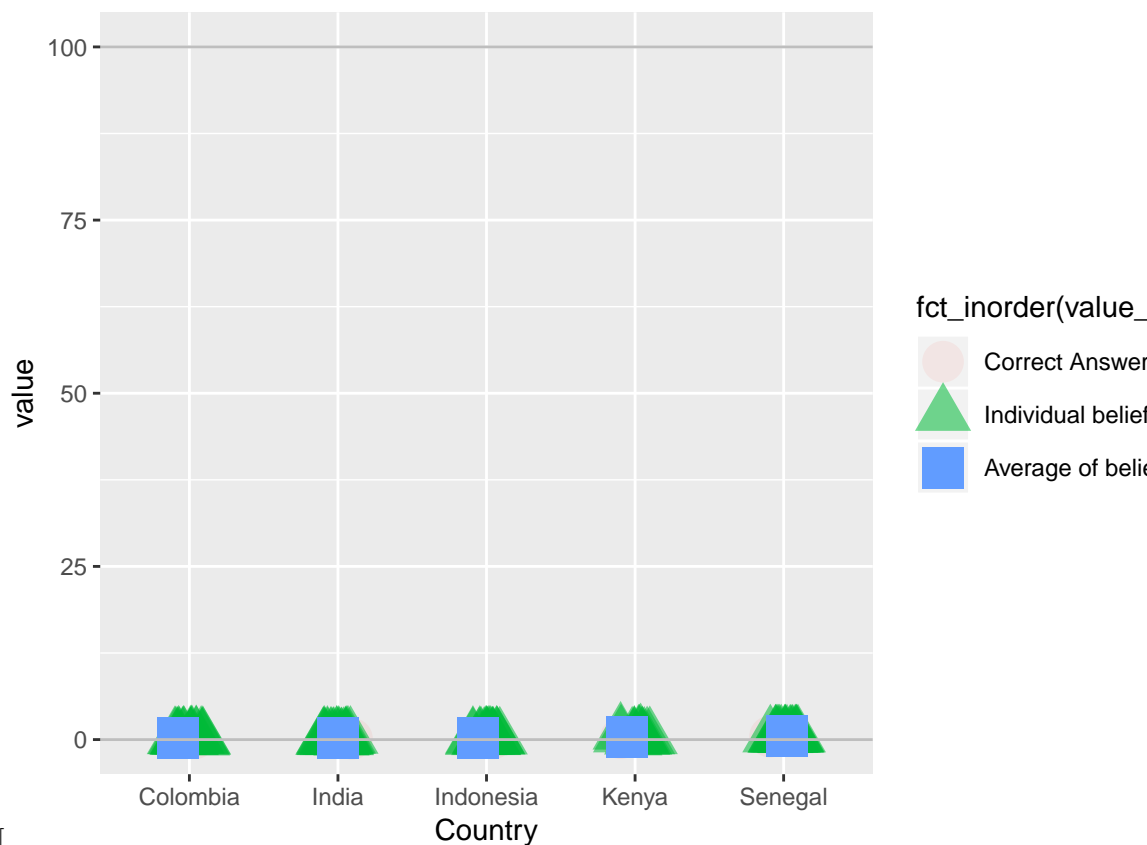
```
]] .column[.content.center[
]] — class: split-50 count: false
```

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



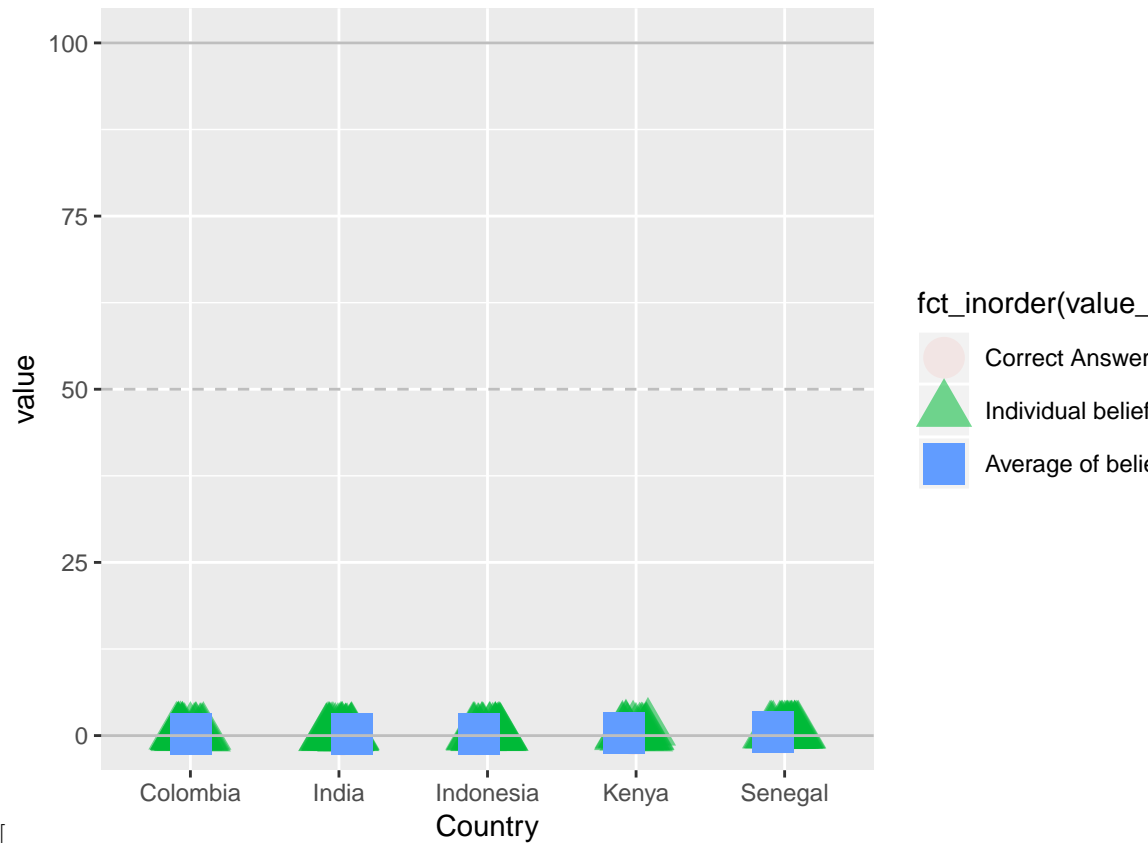
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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 = "") #<<
```



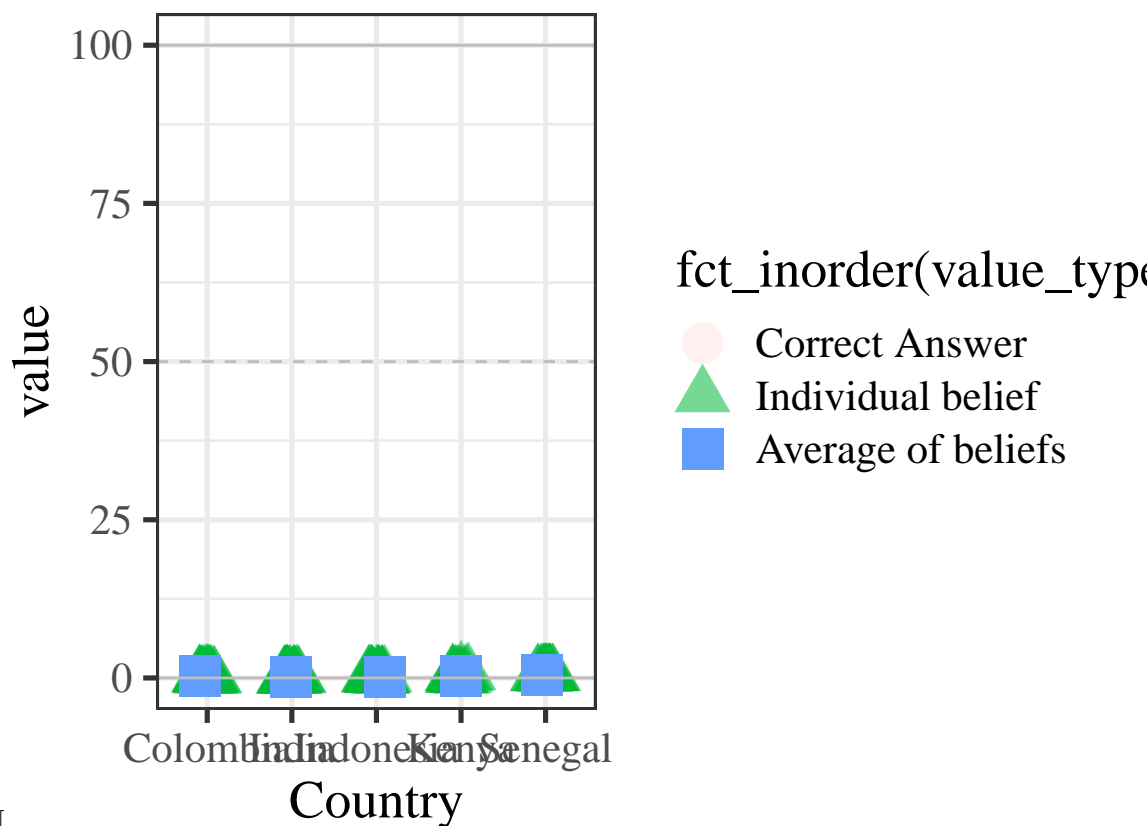
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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") #<<
```



```
]] .column[.content.center[
]] — class: split-50 count: false
```

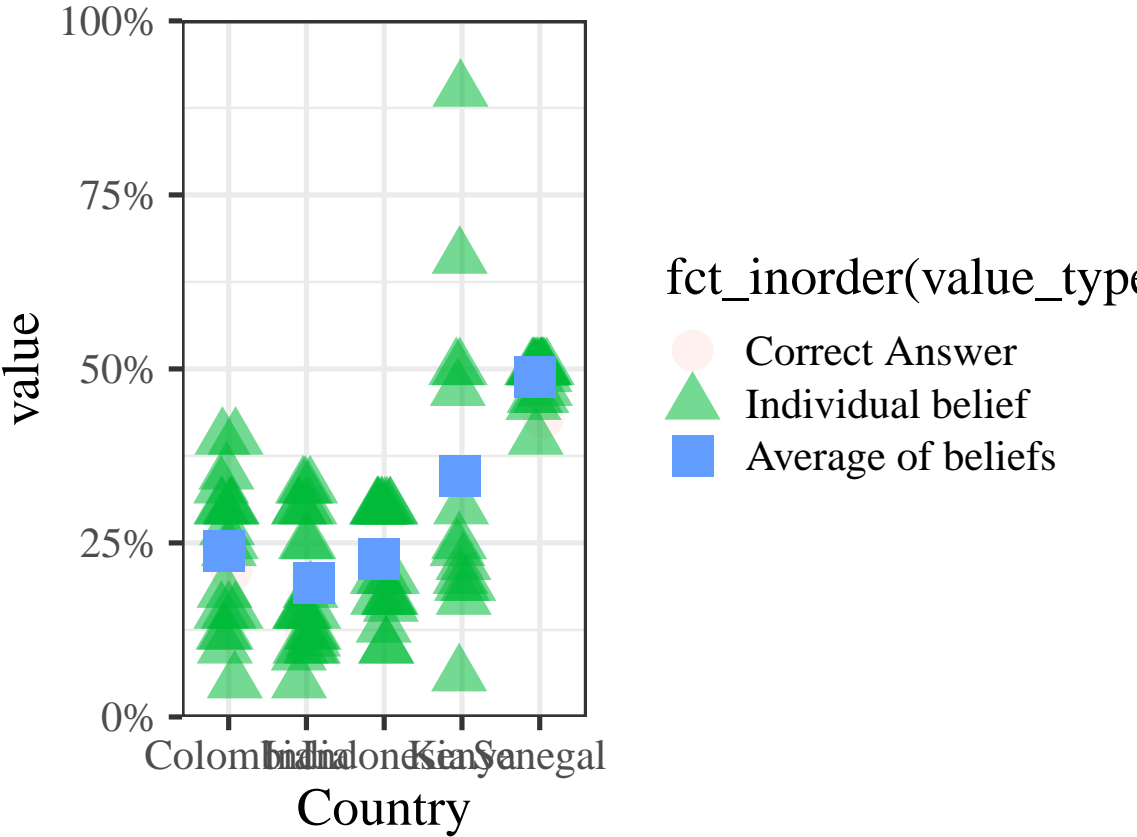
```
.column[.content[
ggplot(df_gather) +
  aes(x = Year) +
  aes(y = Percent) +
  aes(fill = type) +
  facet_wrap(~ Ethnicity) +
  geom_area(alpha = .5, position = "dodge") +
  labs(fill = "") +
  labs(x = "") +
  labs(title = "American Baseball Demographics 1947-2016") +
  labs(subtitle = "Percentage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

```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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() #<<
```

```
]] .column[.content.center[  
]]
```



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!

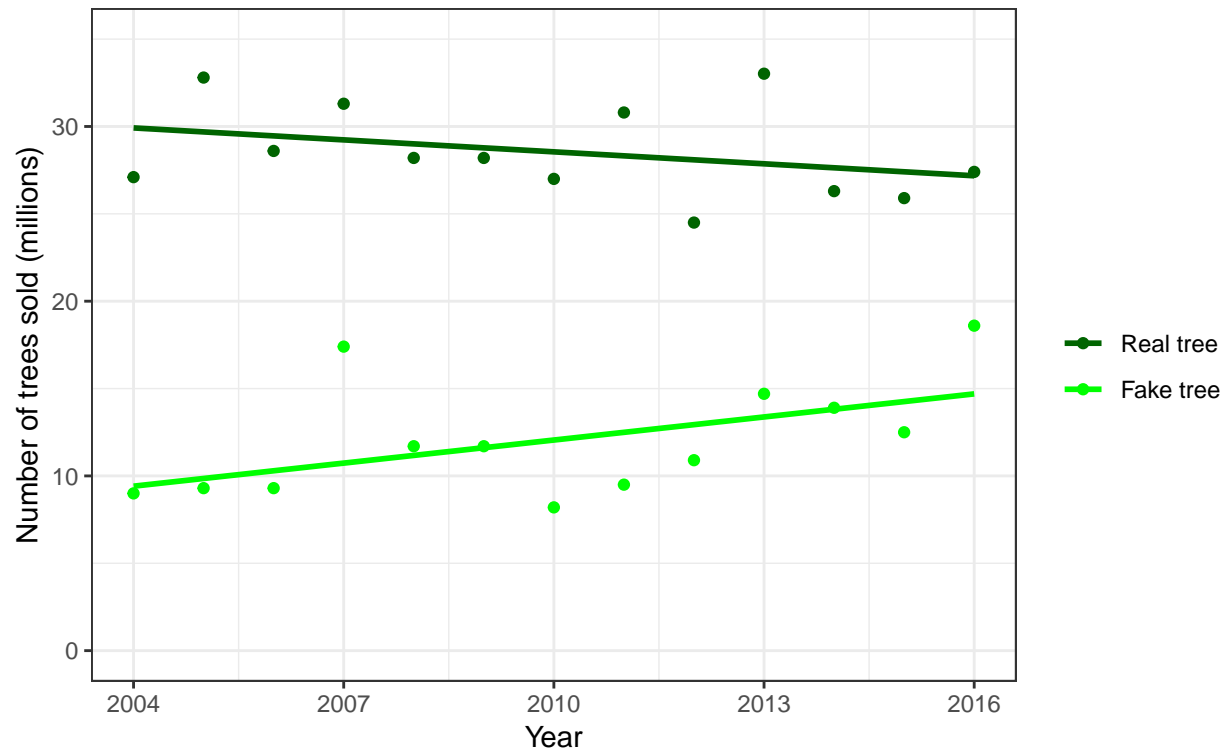
A random sample from the data set:

Year	Number of trees sold	Type of tree	Number of trees sold (millions)
2010	27000000	Real tree	27.0
2006	28600000	Real tree	28.6
2013	14700000	Fake tree	14.7
2008	11700000	Fake tree	11.7
2008	28200000	Real tree	28.2

```
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



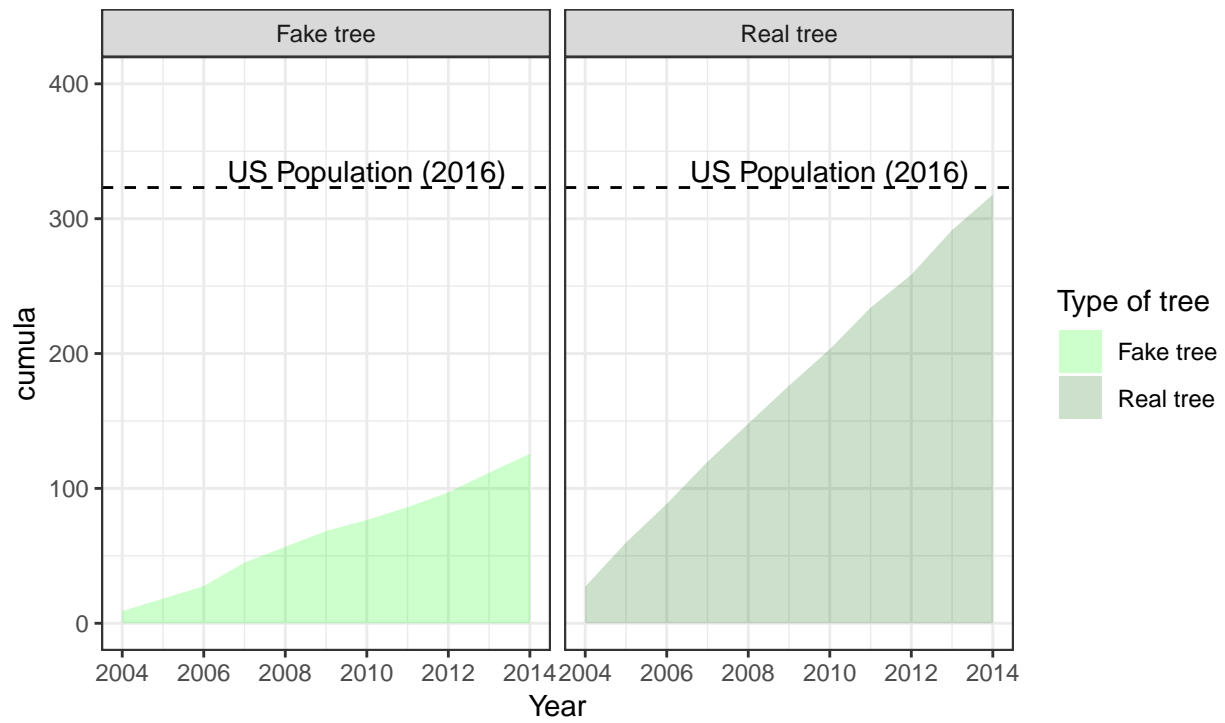
```
dta <- dta %>%
  group_by(`Type of tree`) %>%
  mutate(cumula = cumsum(`Number of trees sold (millions)`))

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 = "Cumulative real and fake Christmas trees sold in the US\nData Source: Statista | @EvaMaeRey") +
  scale_fill_manual(values = c("green", "darkgreen")) +
  theme_bw() +
  ylim(c(0, 400))
```

Ten years of trees.

Cummulative real and fake Christmas trees sold in the US

Data Source: Statista | @EvaMaeRey



Chapter 4

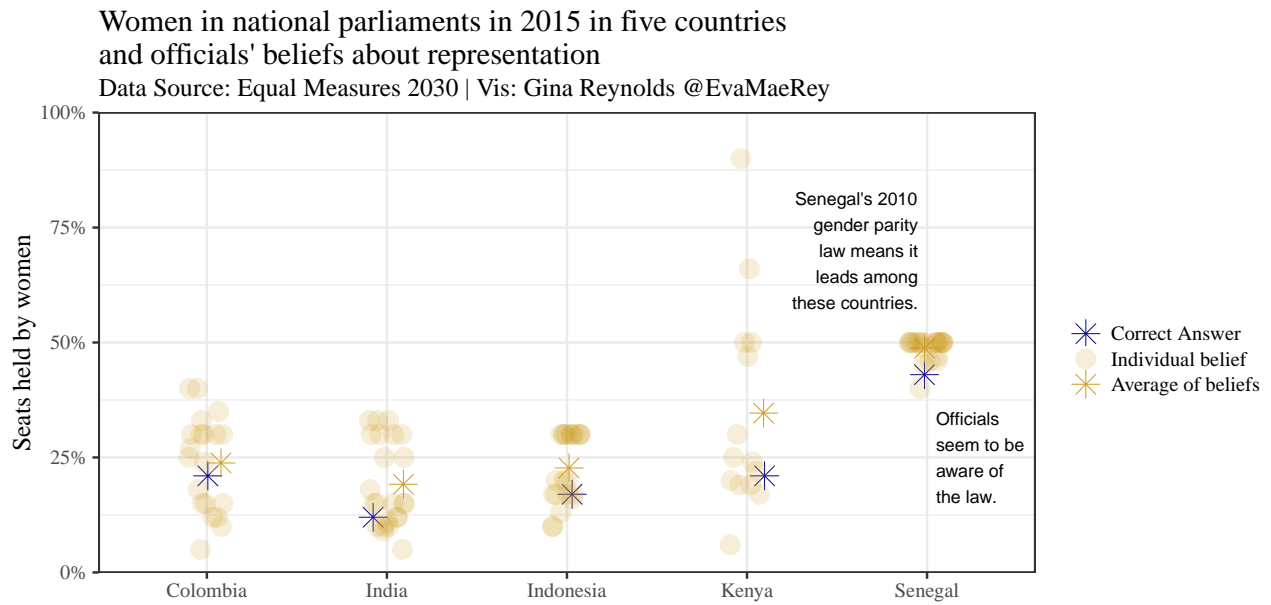
Officials' beliefs about women's representation

The data provided is based on a small survey of elite officials in five less developed countries. The question that arises 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
Indonesia	Share of seats held by women	0.17	Correct Answer	0.7
Colombia	Share of seats held by women	0.33	Individual belief	0.3
Colombia	Share of seats held by women	0.30	Individual belief	0.3
Indonesia	Share of seats held by women	0.30	Individual belief	0.3
Colombia	Share of seats held by women	0.15	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")
```



```
class: split-50 count: false
```

```
.column[.content[
```

```
ggplot(data = df_all) #<<
```

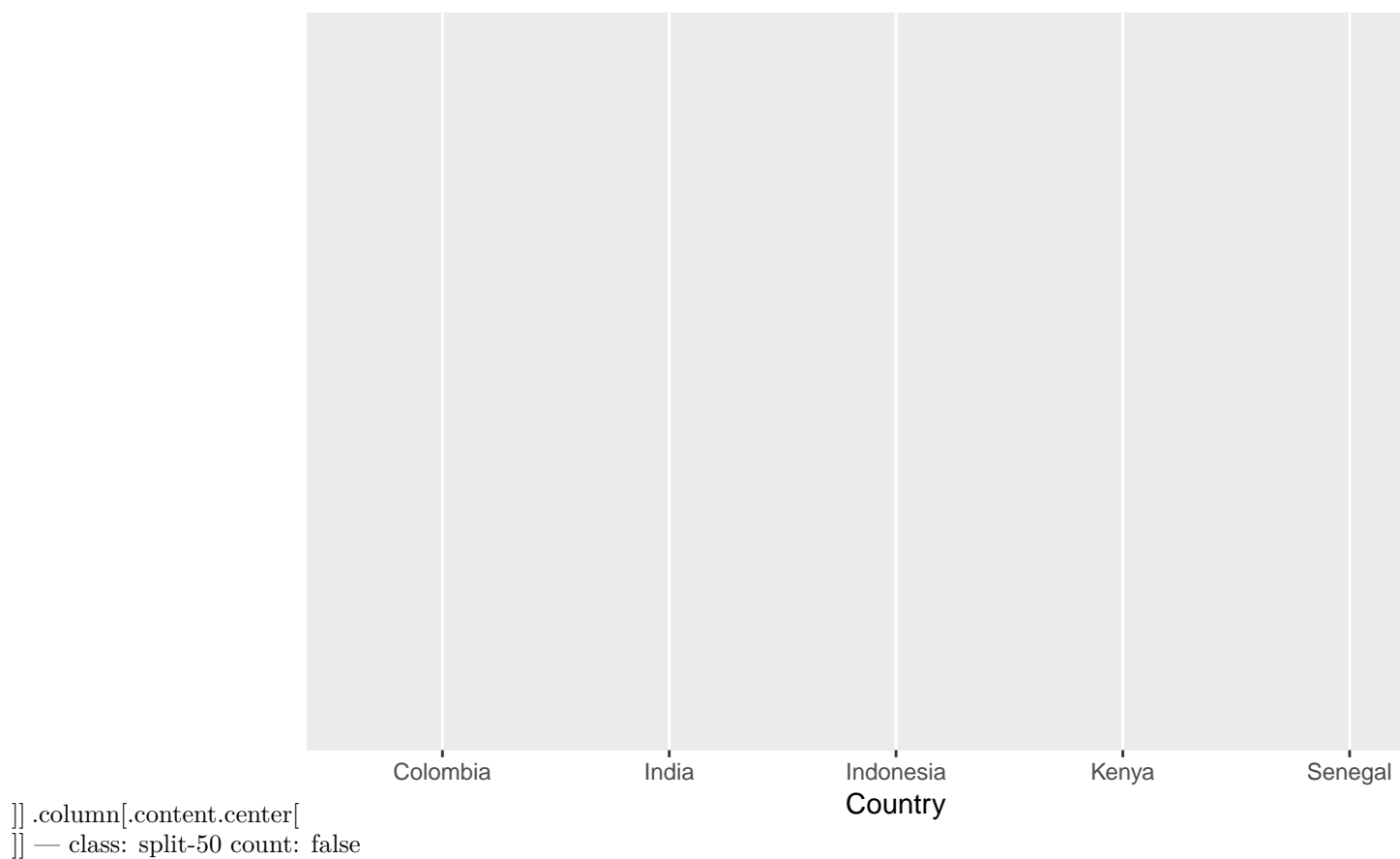
```
]] .column[.content.center[
```

```
]] — class: split-50 count: false
```

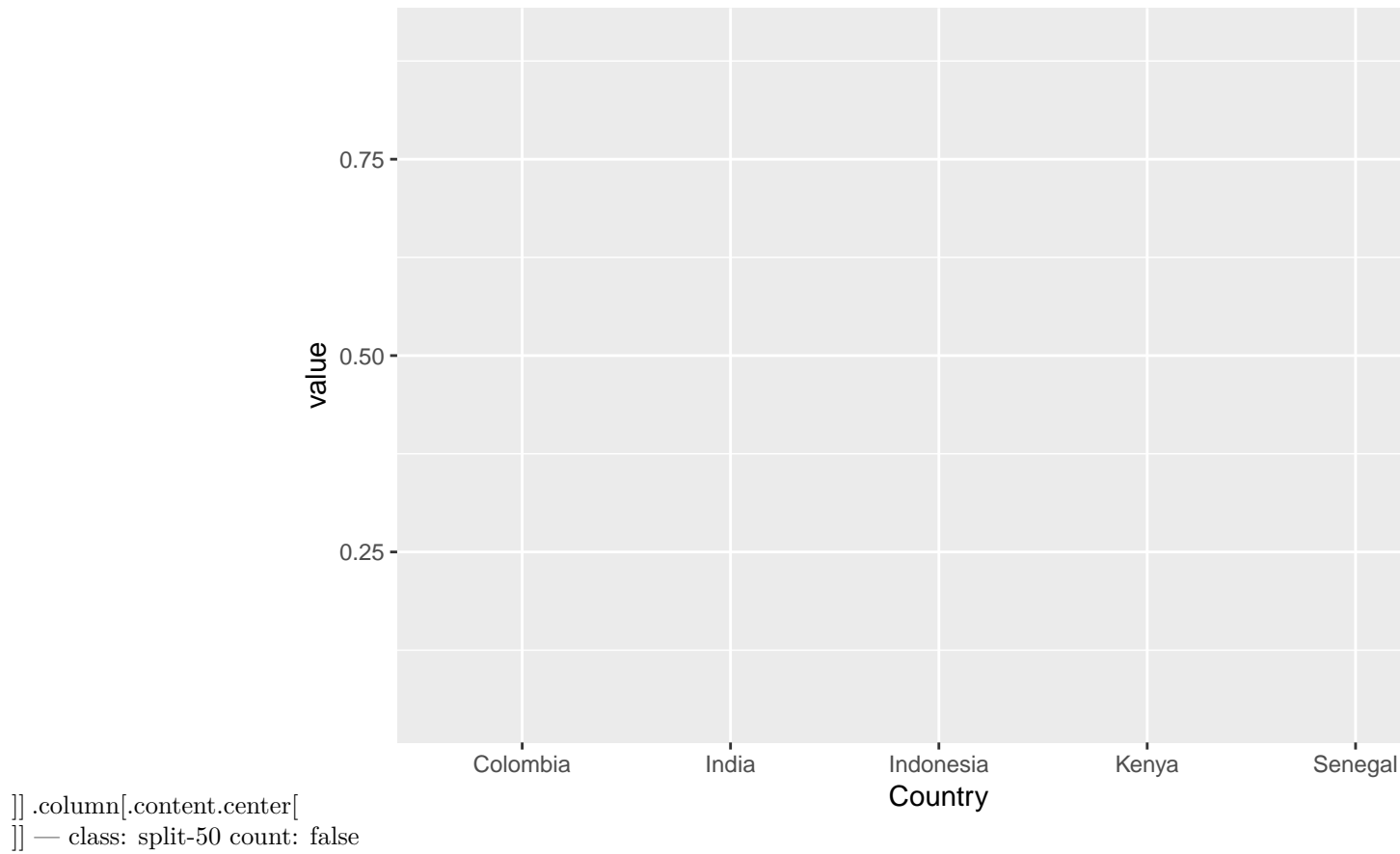
```
.column[.content[
```



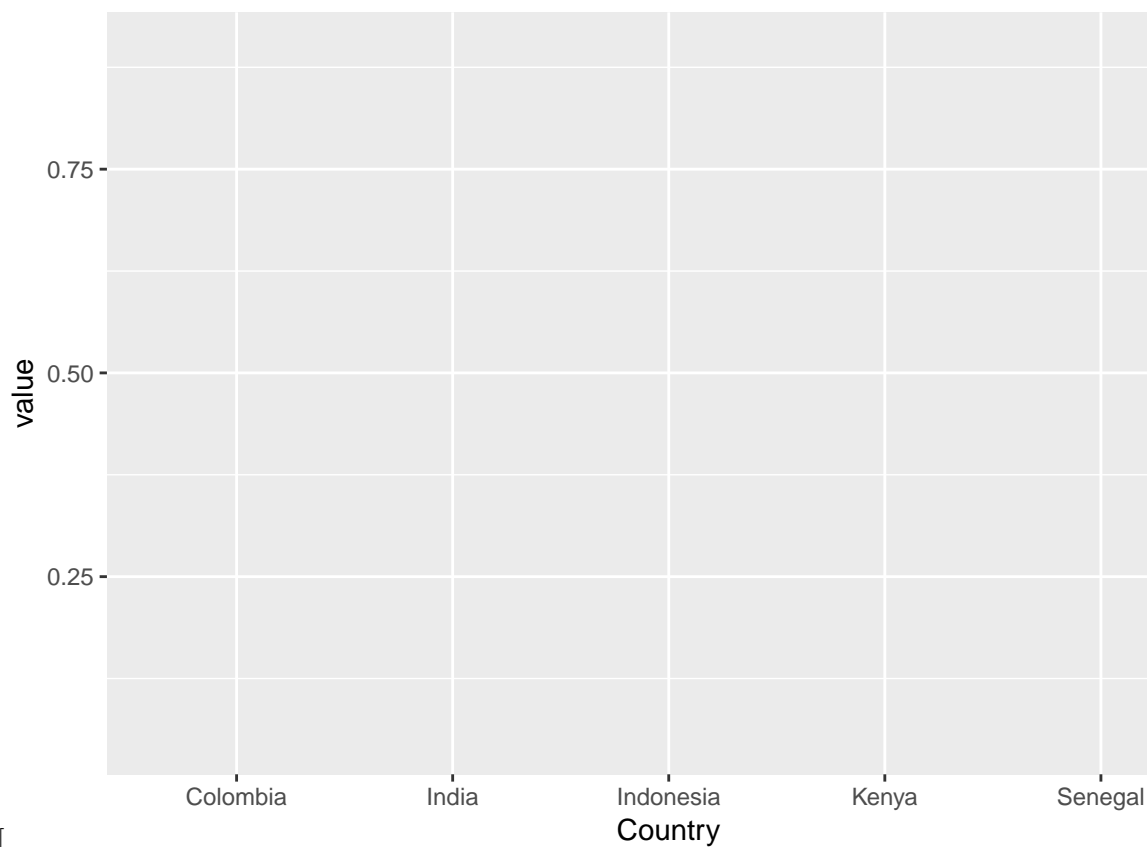
```
ggplot(data = df_all) +
  aes(x = Country) #<<
```



```
.column[.content[
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) #<<
```

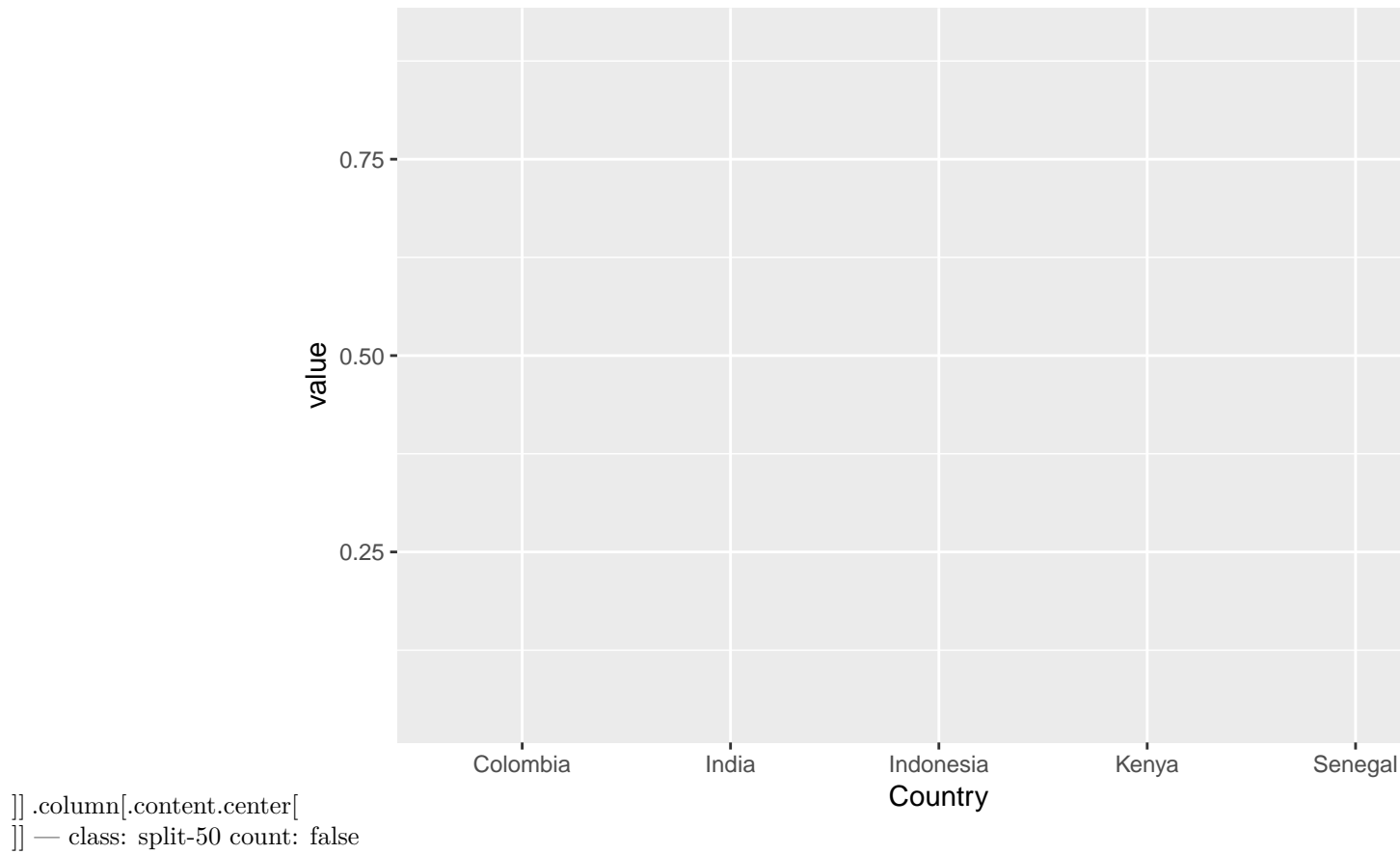


```
.column[.content[
  ggplot(data = df_all) +
    aes(x = Country) +
    aes(y = value) +
    aes(col = fct_inorder(value_type)) #<<
```

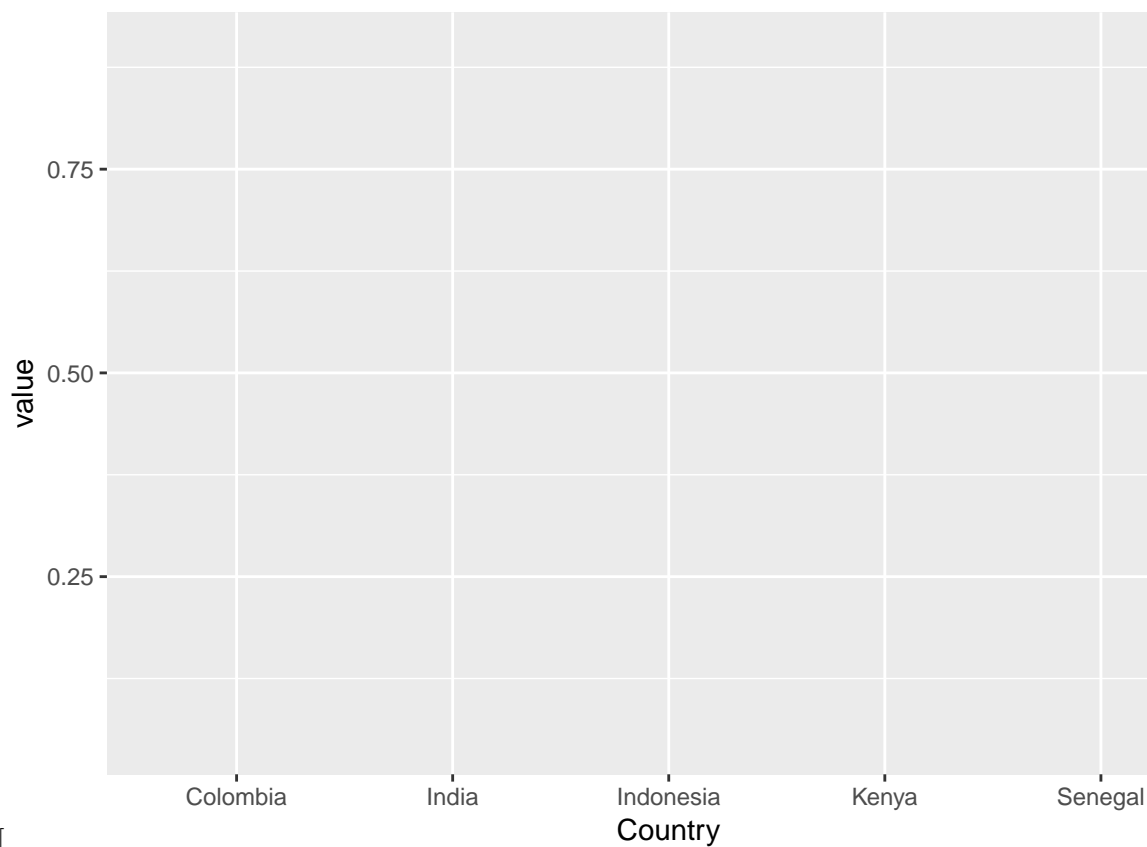


```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) #<<
```

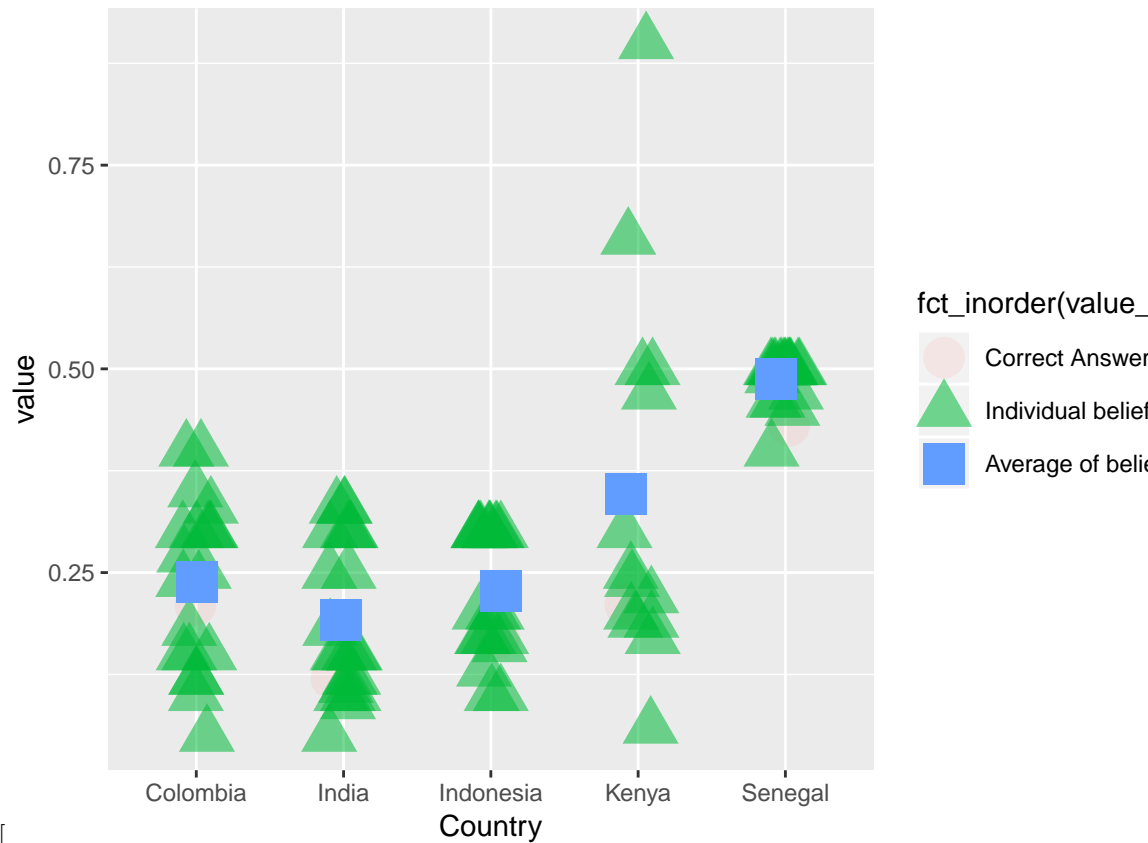


```
.column[.content[
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) #<<
```



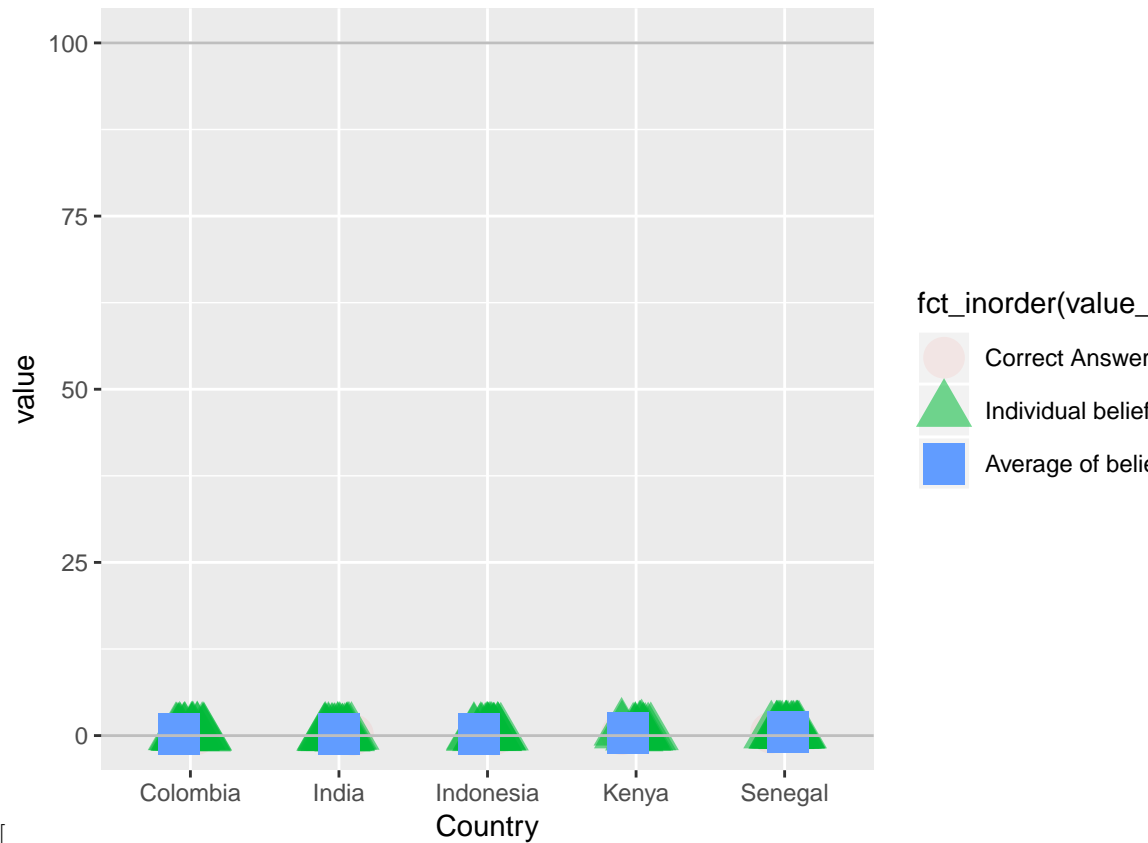
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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) #<<
```



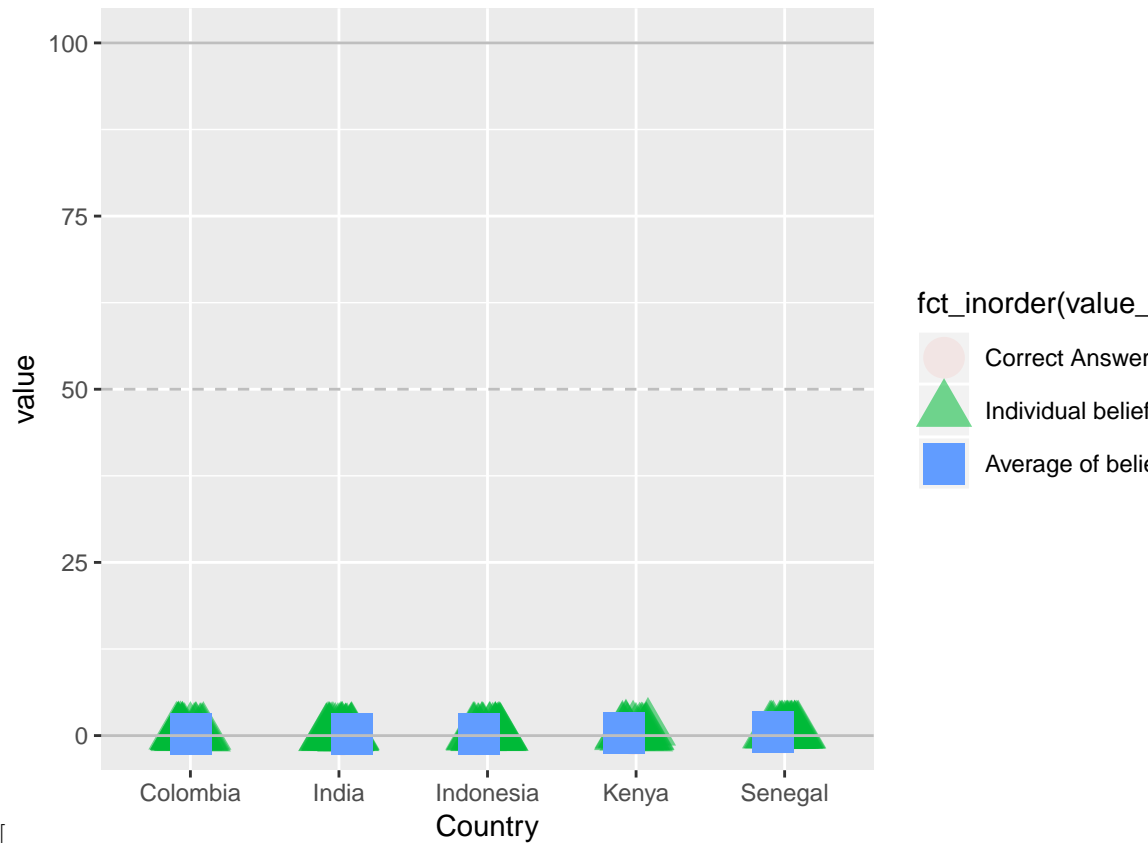
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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") #<<
```



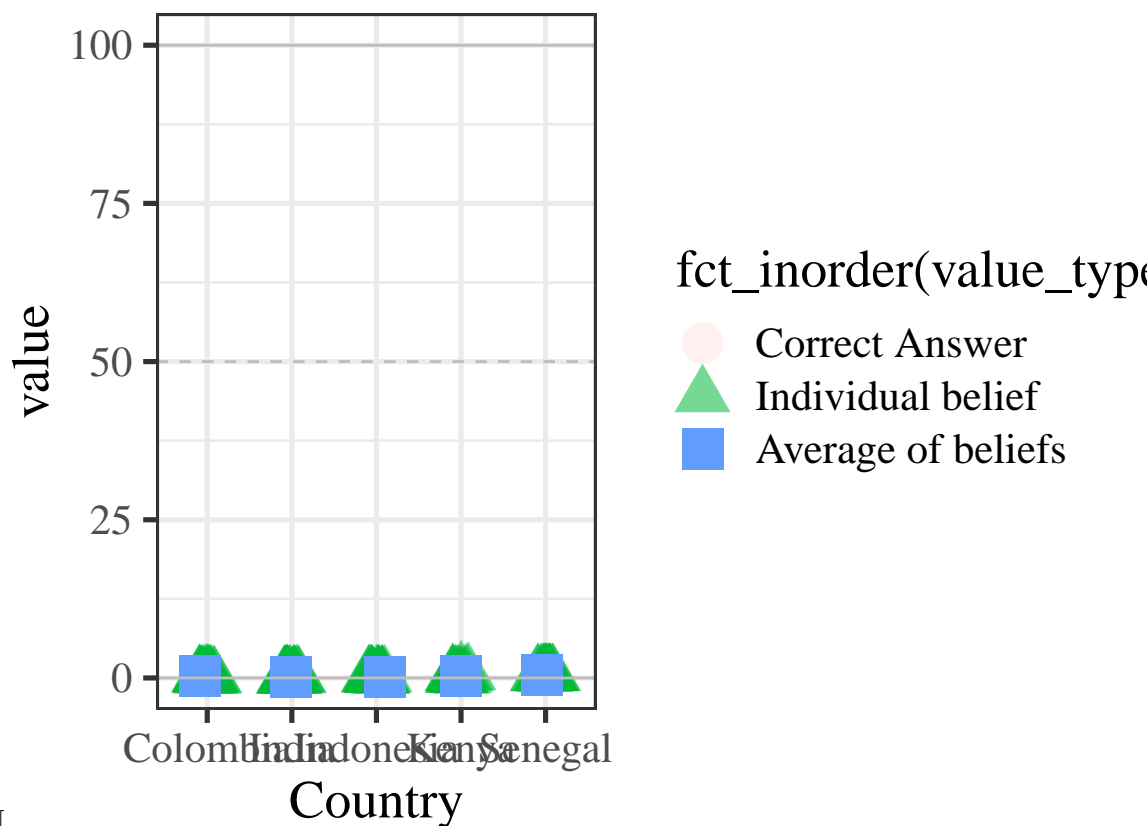
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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") #<<
```



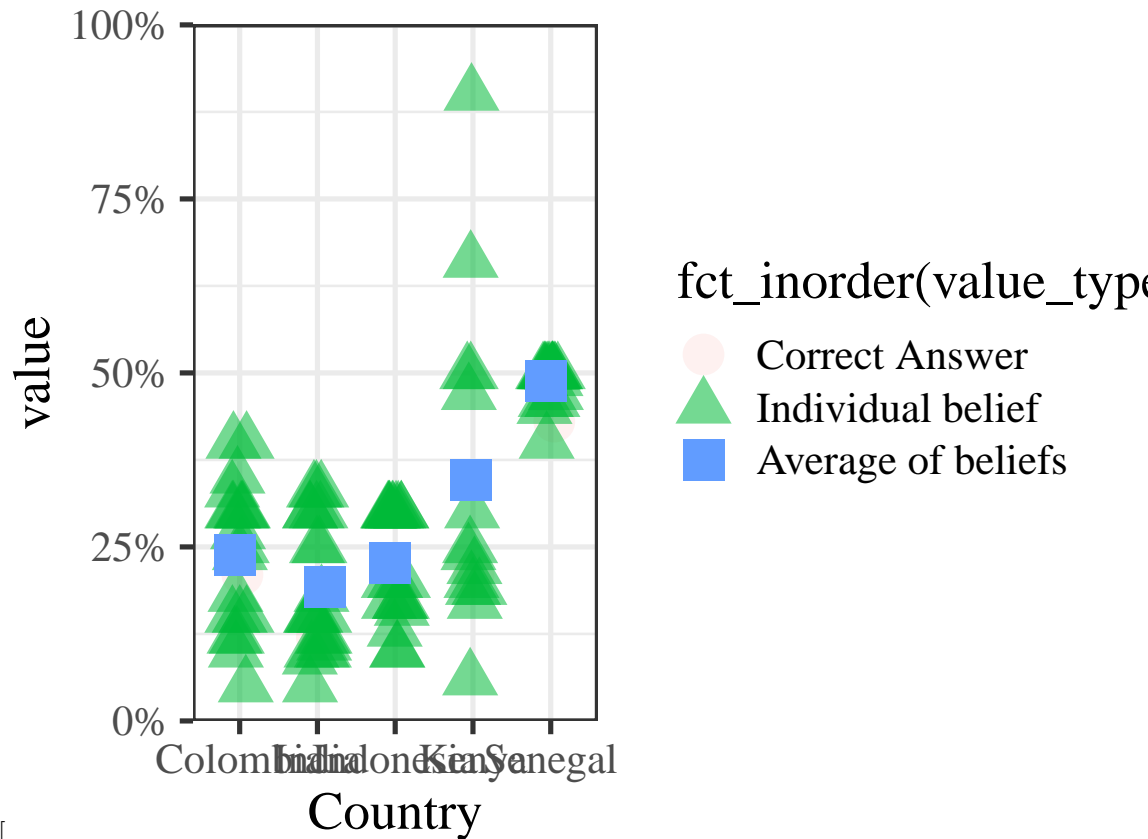
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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") #<<
```

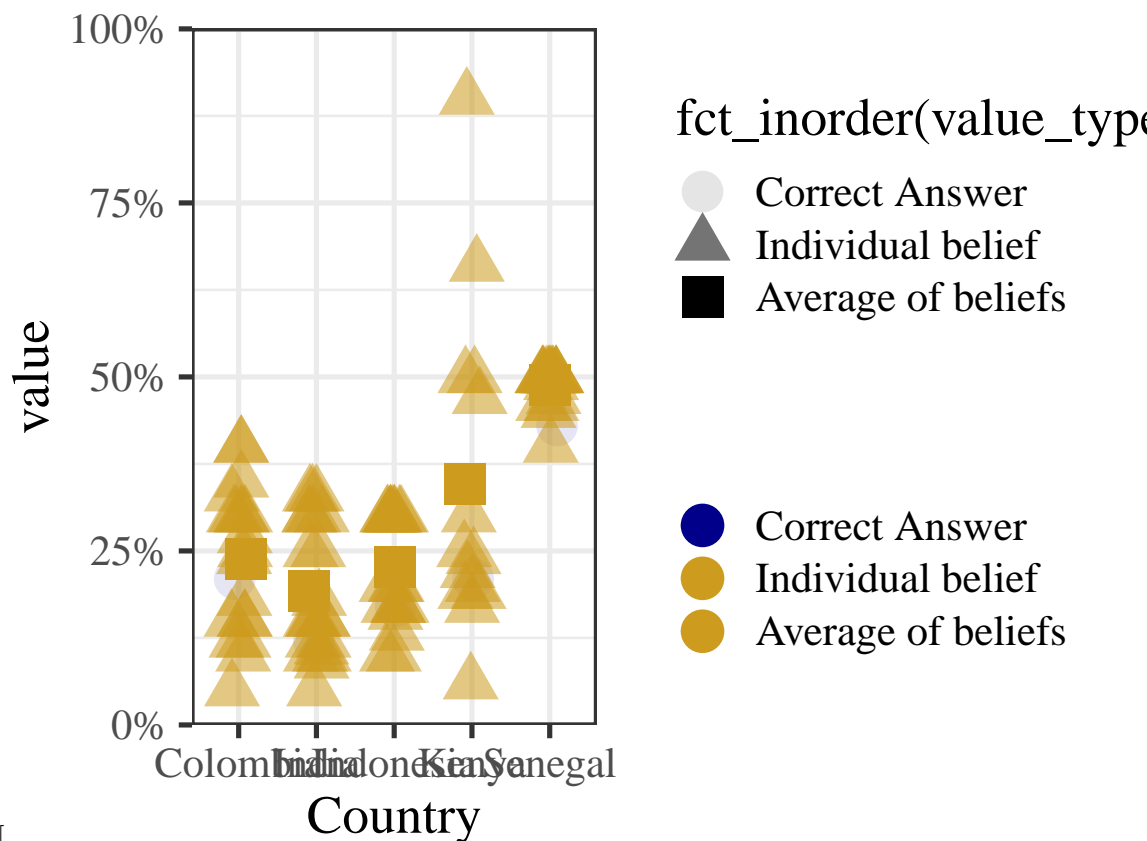
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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) #<<
```



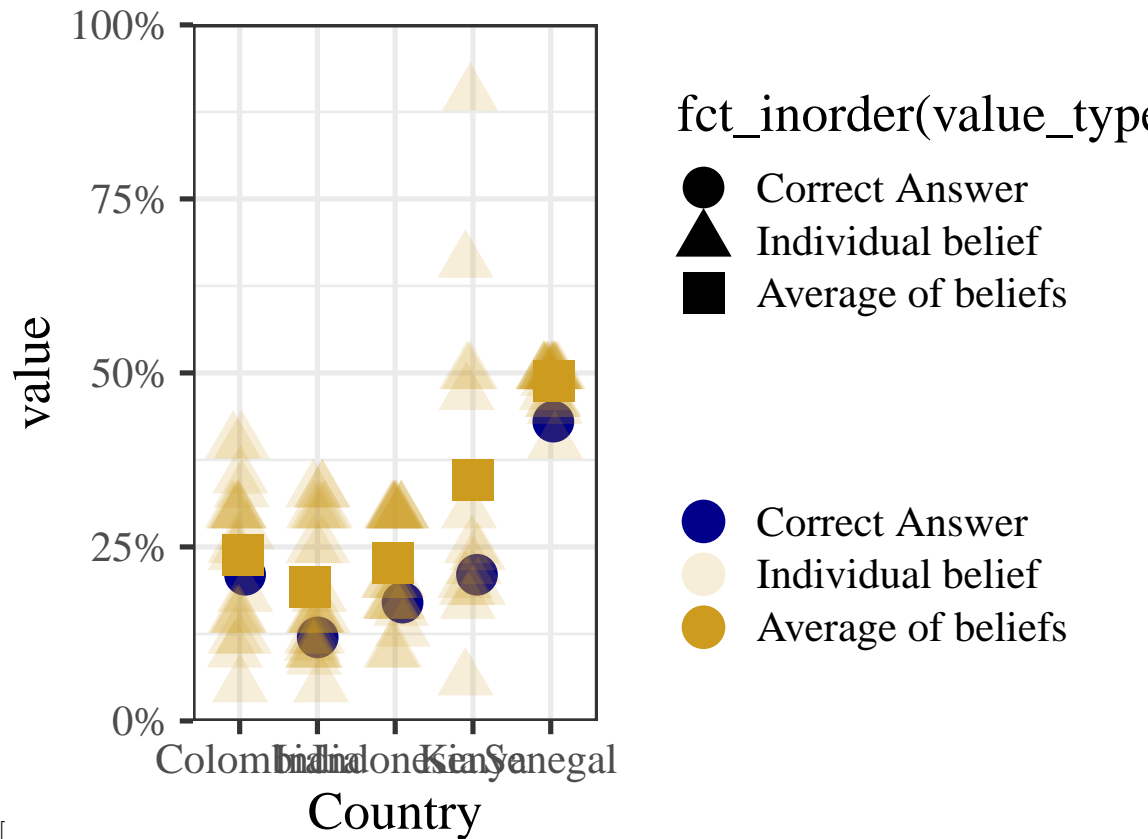
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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")) #<<
```



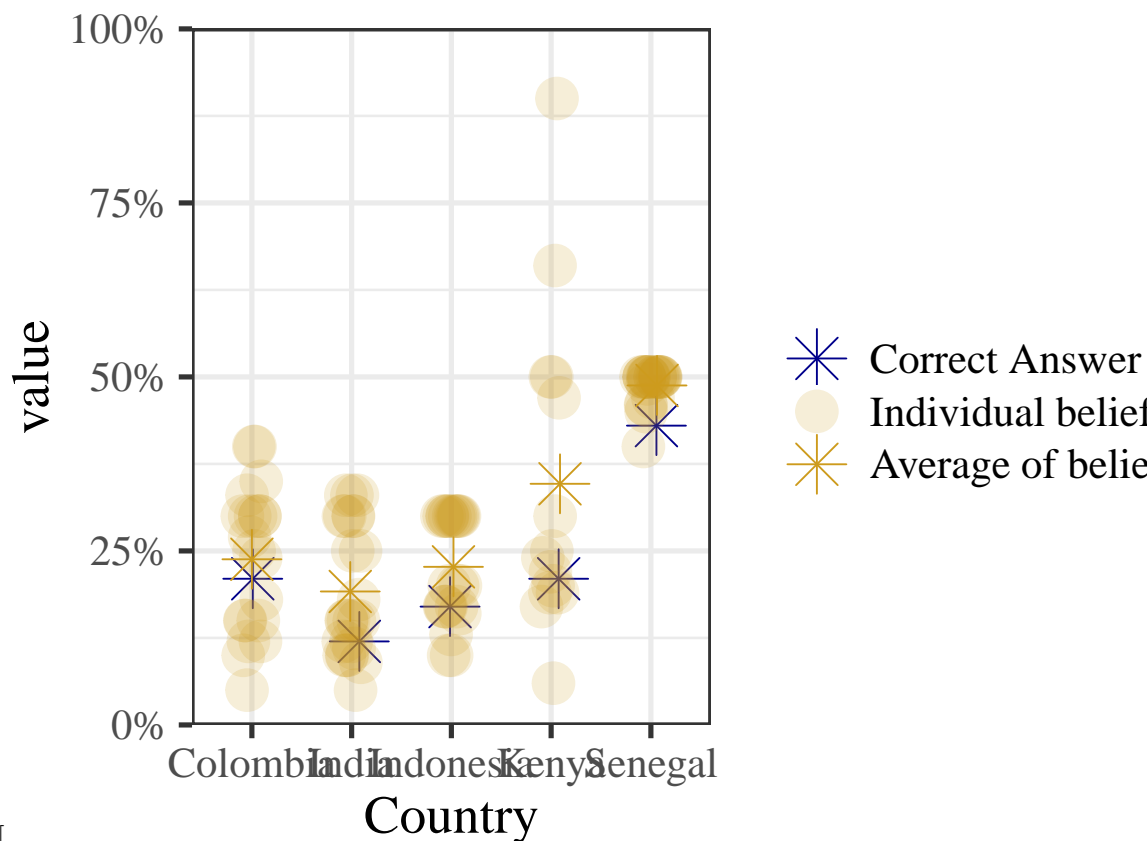
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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)) #<<
```



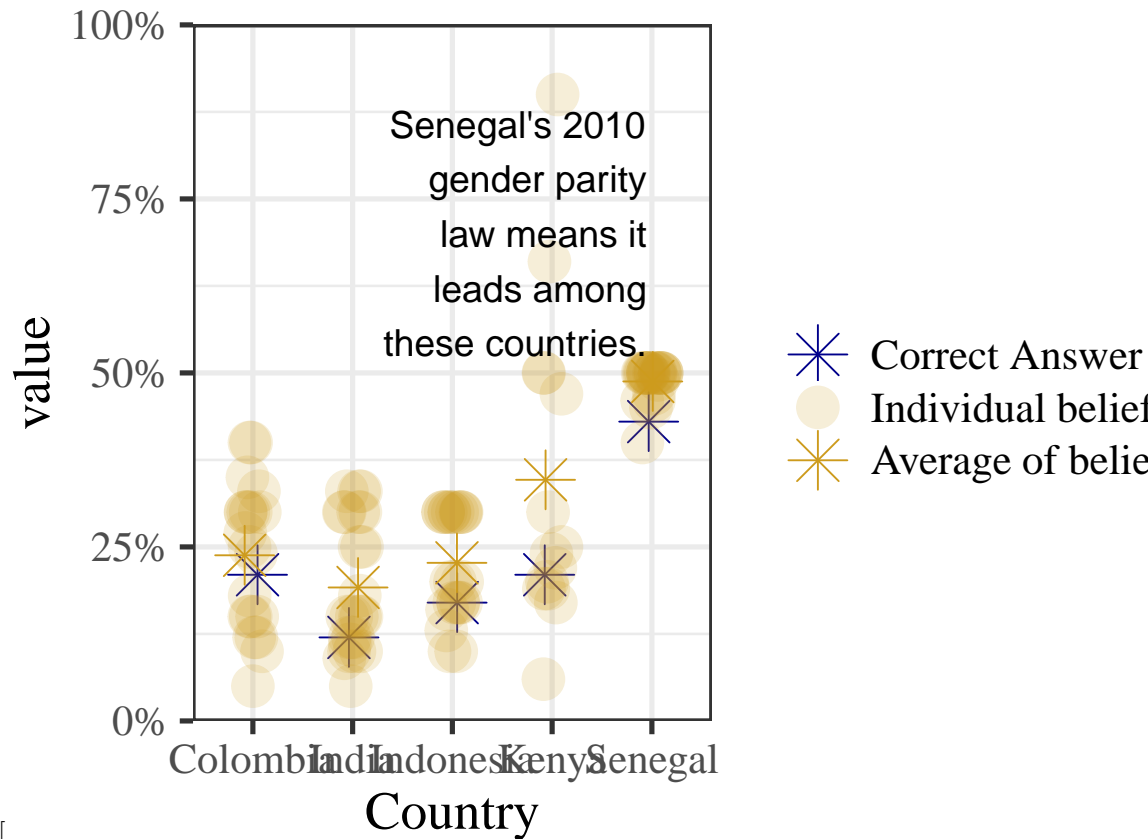
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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)) #<<
```



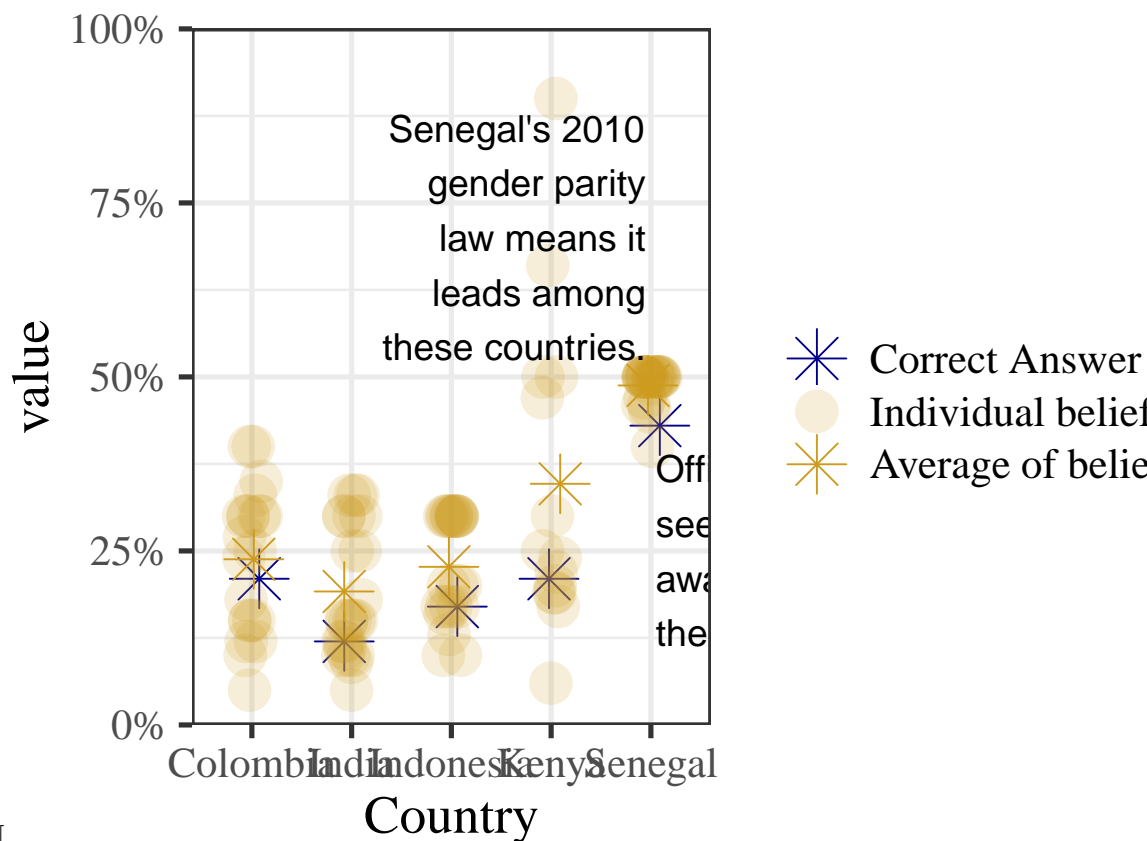
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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
```



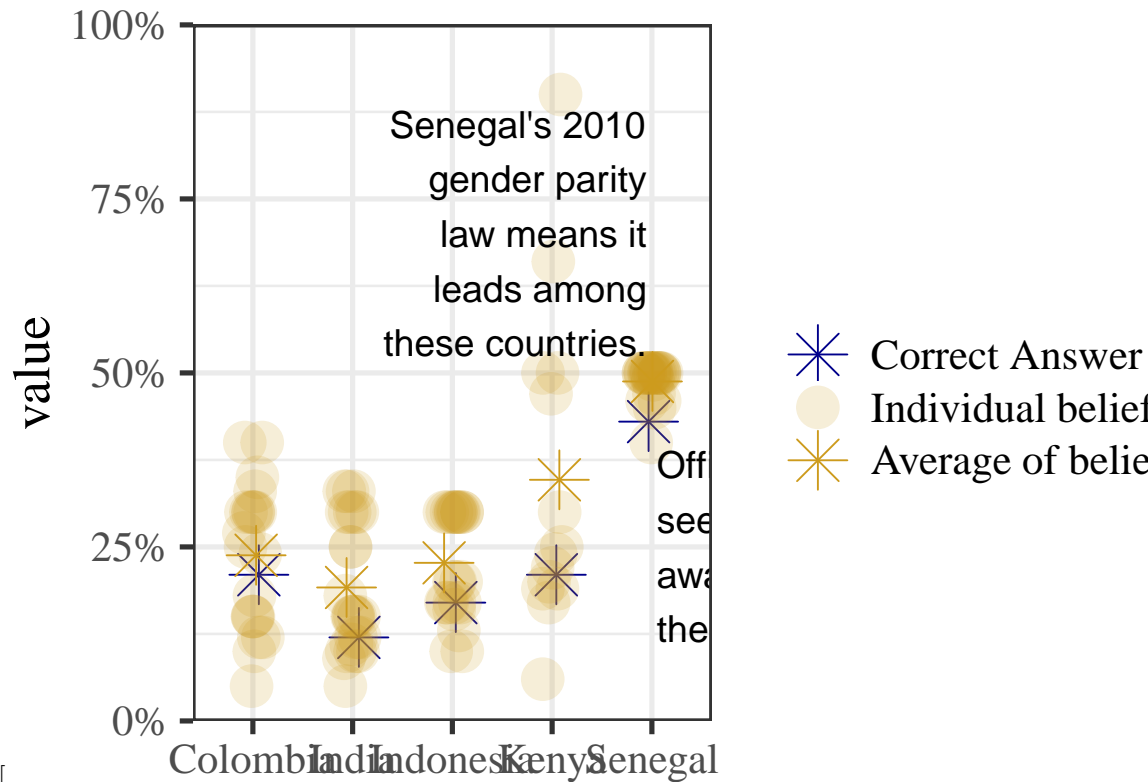
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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 leads among these countries.")) +
  annotate(geom = "text", x = 5.05, y = .250, label = str_wrap("Officials seem to be aware of the law."))
```



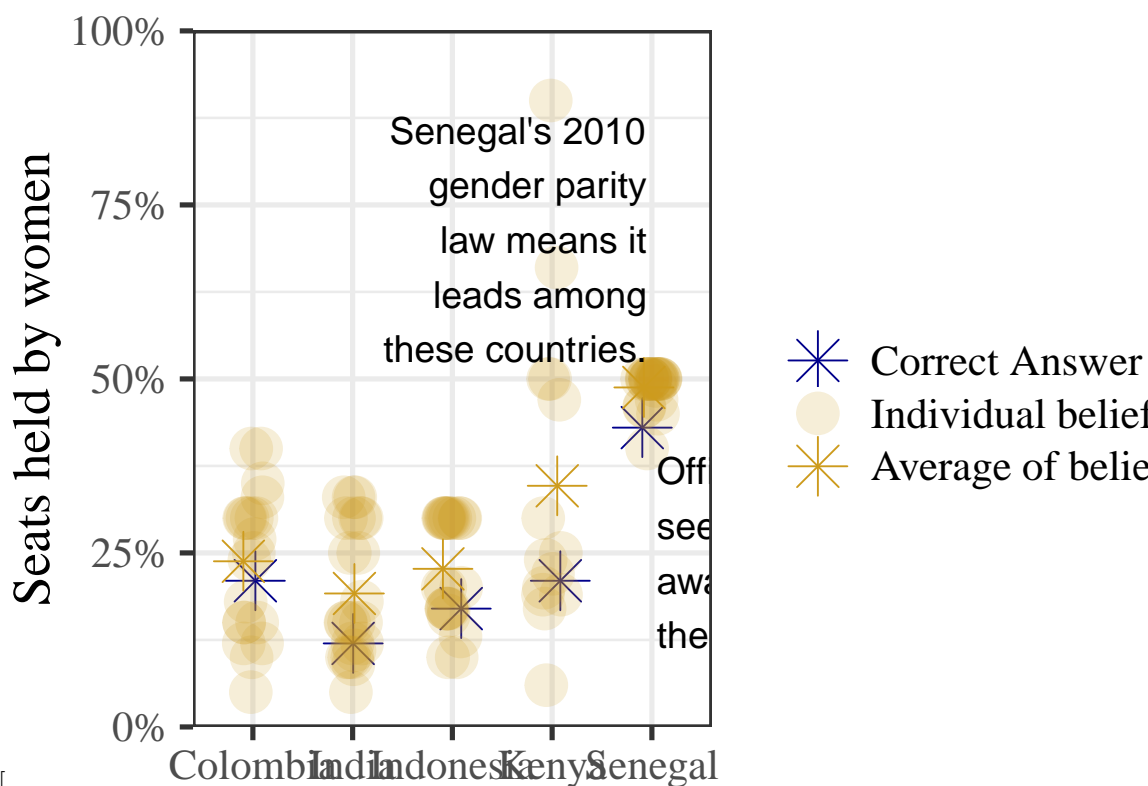
```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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 leads among these countries.")) +
  annotate(geom = "text", x = 5.05, y = .250, label = str_wrap("Officials seem to be aware of the law.")) +
  labs(x = "") #<<
```



```
]] .column[.content.center[
]] — class: split-50 count: false
```

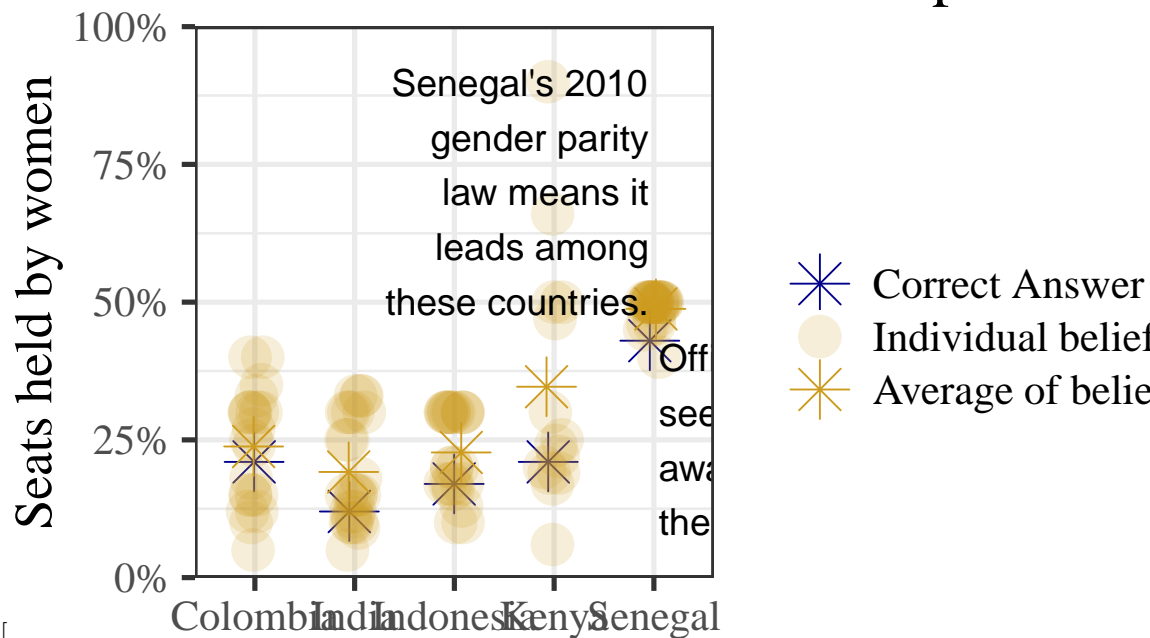
```
.column[.content[
ggplot(data = df_all) +
  aes(x = Country) +
  aes(y = value) +
  aes(col = fct_inorder(value_type)) +
  aes(alpha = fct_inorder(value_type)) +
  aes(shape = fct_inorder(value_type)) +
  geom_jitter(width = .1, height = 0, size = 7) +
  geom_hline(yintercept = c(0, 100), col = "grey") +
  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 leads among these countries.")) +
  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") #<<
```

```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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 ")
```

Women in national parliaments in 2015 and officials' beliefs about representation

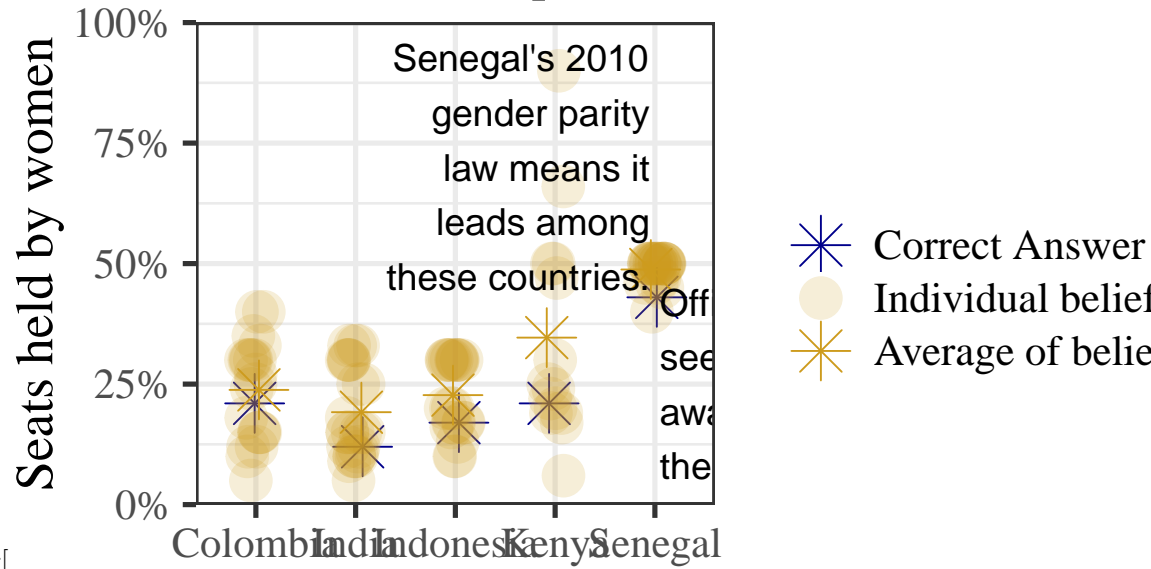


```
]] .column[.content.center[
]] — class: split-50 count: false
```

```
.column[.content[
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 leads among these countries.")) +
  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 representation") +
  labs(subtitle = "Data Source: Equal Measures 2030 | Vis: Gina Reynolds @EvaMaeRey") #<<
```

Women in national parliaments in 2010 and officials' beliefs about representation

Data Source: Equal Measures 2030 | Vis: Girardin



```
]] .column[.content.center[
]]
```

```
.remark-code{ line-height: 2; }
```


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:

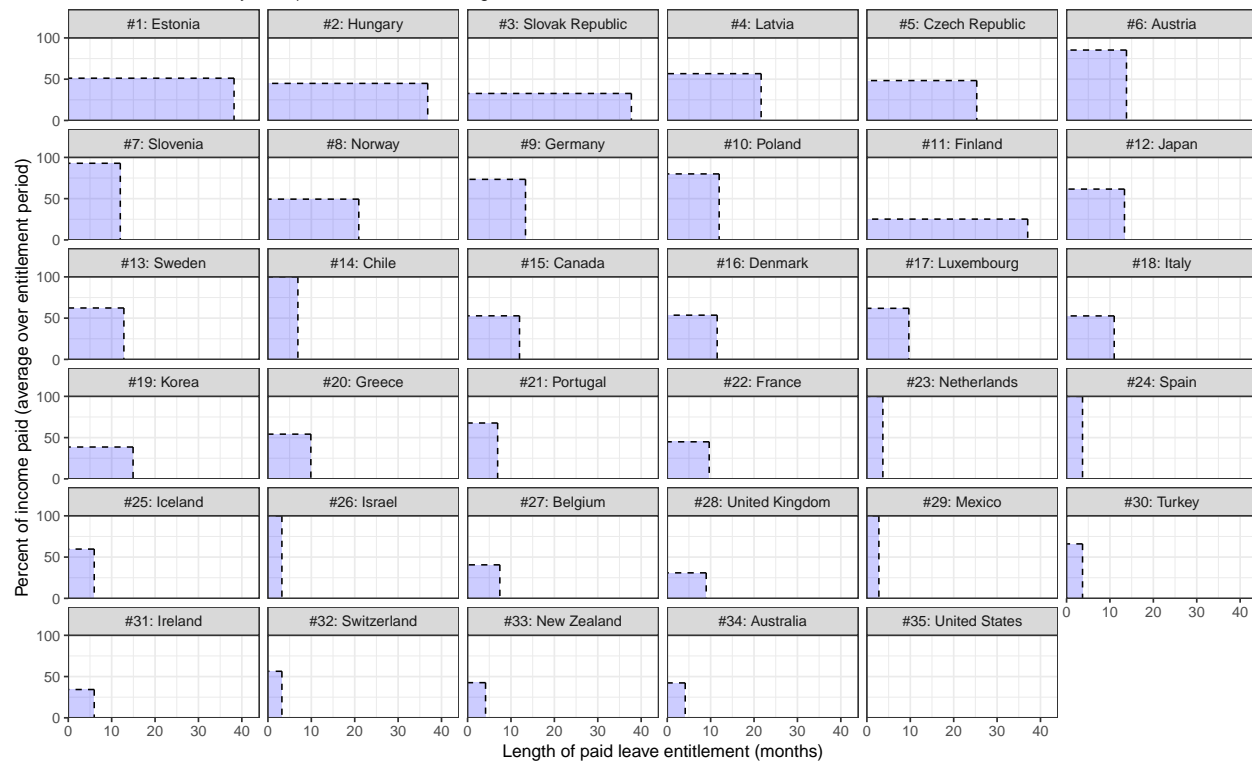
Country	Paid maternity leave avg payment rate (%)	Paid maternity leave full rate equivalent in weeks	Paid ma
Japan	67.0	9.4	
Sweden	77.6	10.0	
Denmark	53.6	9.6	
New Zealand	42.6	7.7	
Spain	100.0	16.0	

```
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)") +  
  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



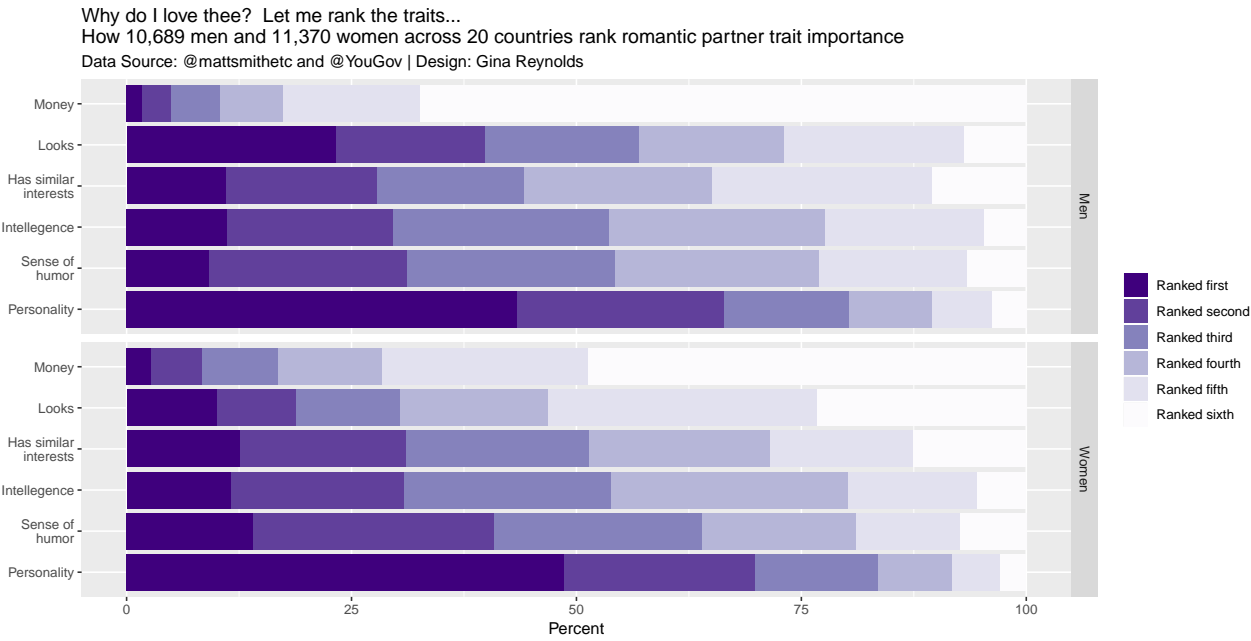
Chapter 6

Traits

A random sample from the data set:

Gender	Question_short	Rank (text)	Rank (number)	n	Percent
Men	Personality	Ranked first	1	4641.14	43.411898
Men	Looks	Ranked fourth	4	1722.09	16.110862
Men	Personality	Ranked fourth	4	989.05	9.251291
Women	Intellegence	Ranked second	2	2185.55	19.238336
Men	Money	Ranked second	2	339.41	3.180516

```
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 = guide_legend(reverse = TRUE)  
  ) +  
  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

Salaries of Trump and Obama White House Employees

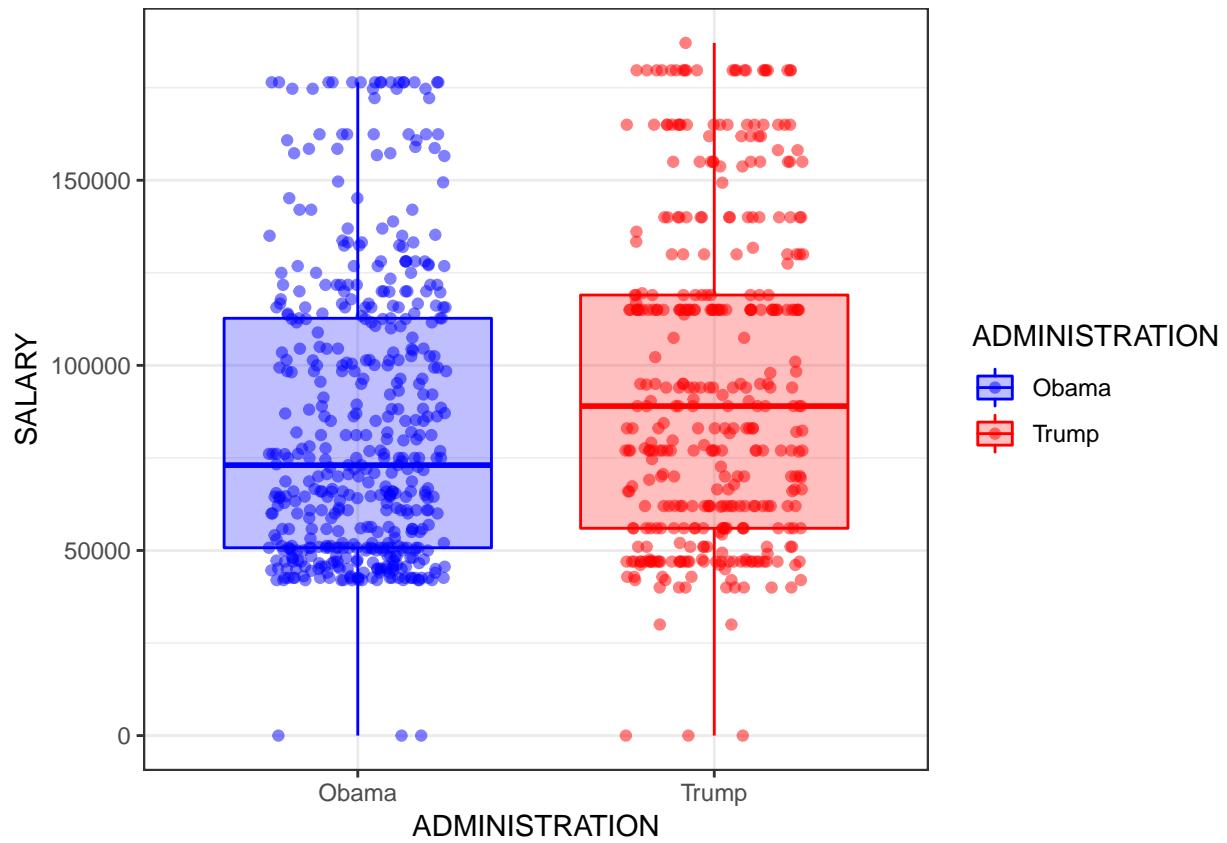
The data set, originally reported on in an NPR article, shows the difference in the distribution of salaries for the Obama and early Trump White House.

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	Smart, Kelly J.	Employee	55000	Per Annum	ASSOCIATE DIGITAL PRODUC
Obama	Lezotte, Darin R.	Employee	80073	Per Annum	SUPERVISOR OF SEARCH AND
Trump	Hoelscher, Douglas L.	Employee	115000	Per Annum	SPECIAL ASSISTANT TO THE I
Obama	Lechtenberg, Tyler A.	Employee	87103	Per Annum	SENIOR PRESIDENTIAL SPEEC
Trump	Tripepi, Timothy G.	Employee	115000	Per Annum	SPECIAL ASSISTANT TO THE I

```
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()
```



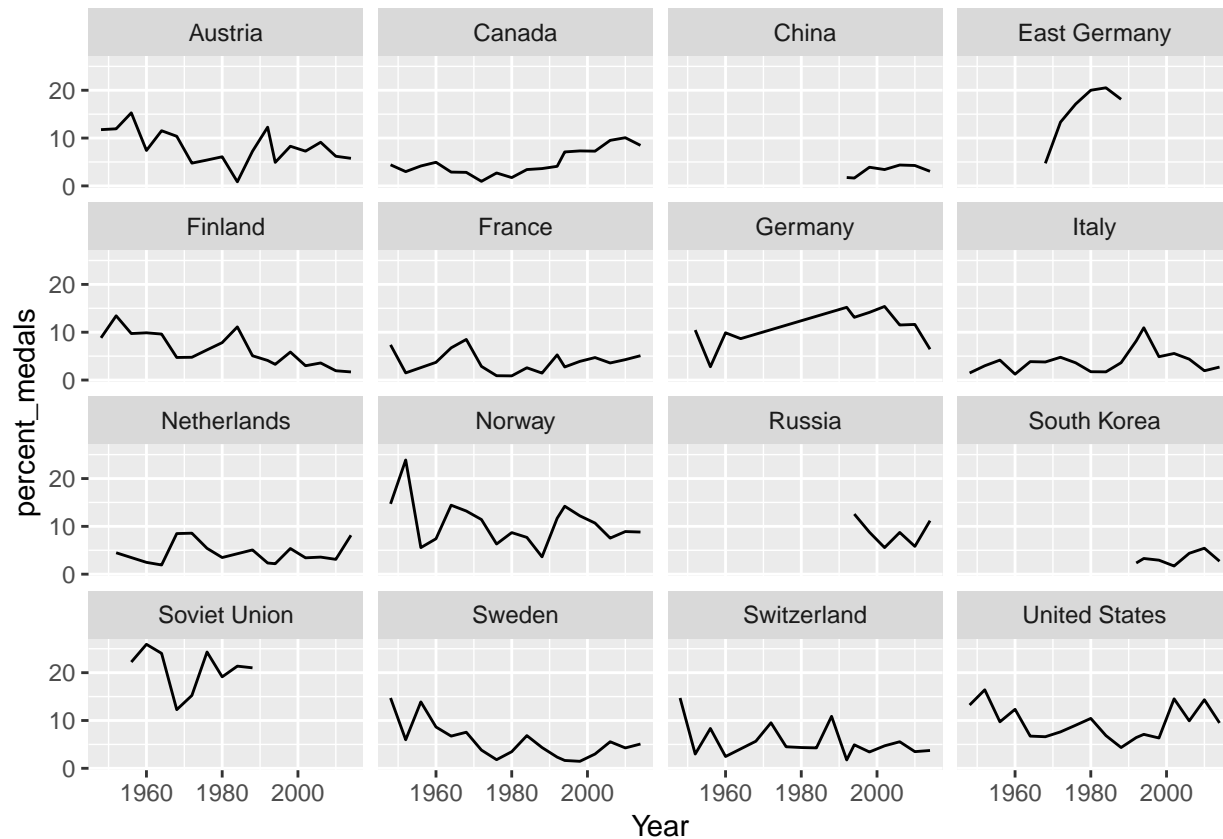
Chapter 8

Winter Games

A random sample from the data set:

Year	Sport	Event	Country	Gender	Medal Rank	Medal	Name of Athlete or
1984	Figure Skating	Women's Singles	East Germany	Women	1	gold	Katarina Witt
2002	Alpine Skiing	Women's Combined	Austria	Women	2	silver	Renate GÄttschl
1976	Nordic Combined	Men's Individual	West Germany	Men	2	silver	Urban Hettich
1998	Alpine Skiing	Men's Slalom	Norway	Men	1	gold	Hans Petter Buraas
1984	Figure Skating	Mixed Pairs	Soviet Union	Mixed	3	bronze	Soviet Union-2

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



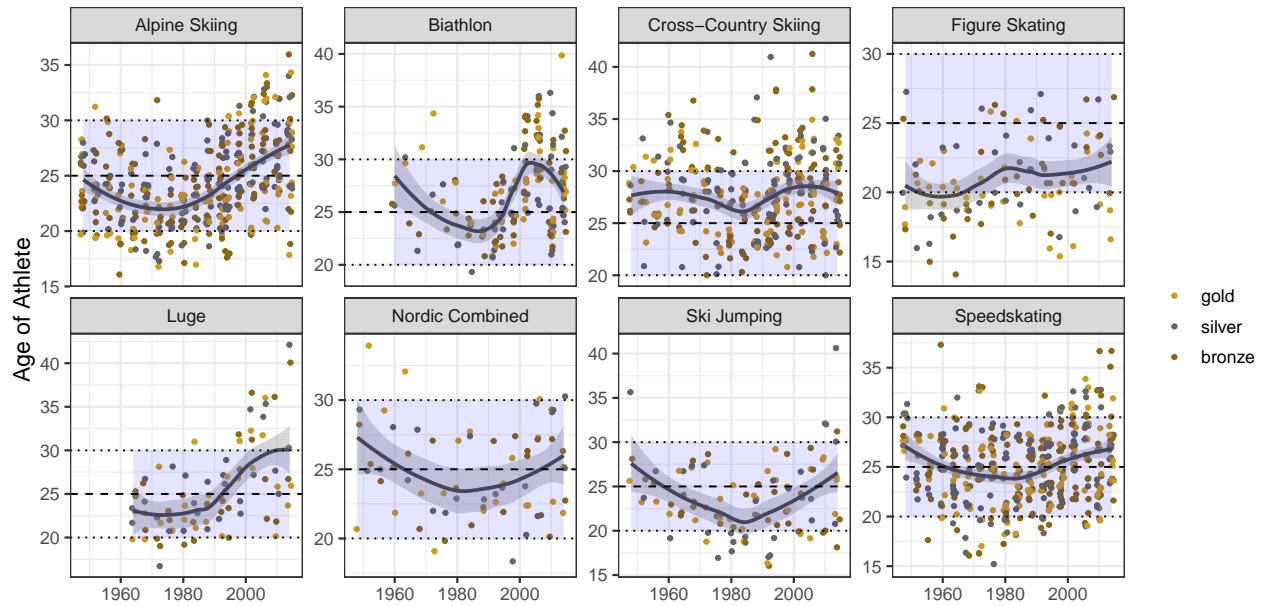
A random sample from the data set:

Year	Sport	Event	Country	Gender	Medal Rank	Medal	Name
1960	Speedskating	Women's 1,500 Meters	Soviet Union	Women	1	gold	Lidiya
1980	Speedskating	Women's 3,000 Meters	United States	Women	3	bronze	Beth L
2002	Biathlon	Women's 10-Kilometer pursuit	Russia	Women	1	gold	Olga L
1976	Luge	Men's Singles	East Germany	Men	3	bronze	Hans
1988	Cross-Country Skiing	Women's 20 Kilometers	Soviet Union	Women	2	silver	Anfisa

```
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 years") +
  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 reveal overlap") +
  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.

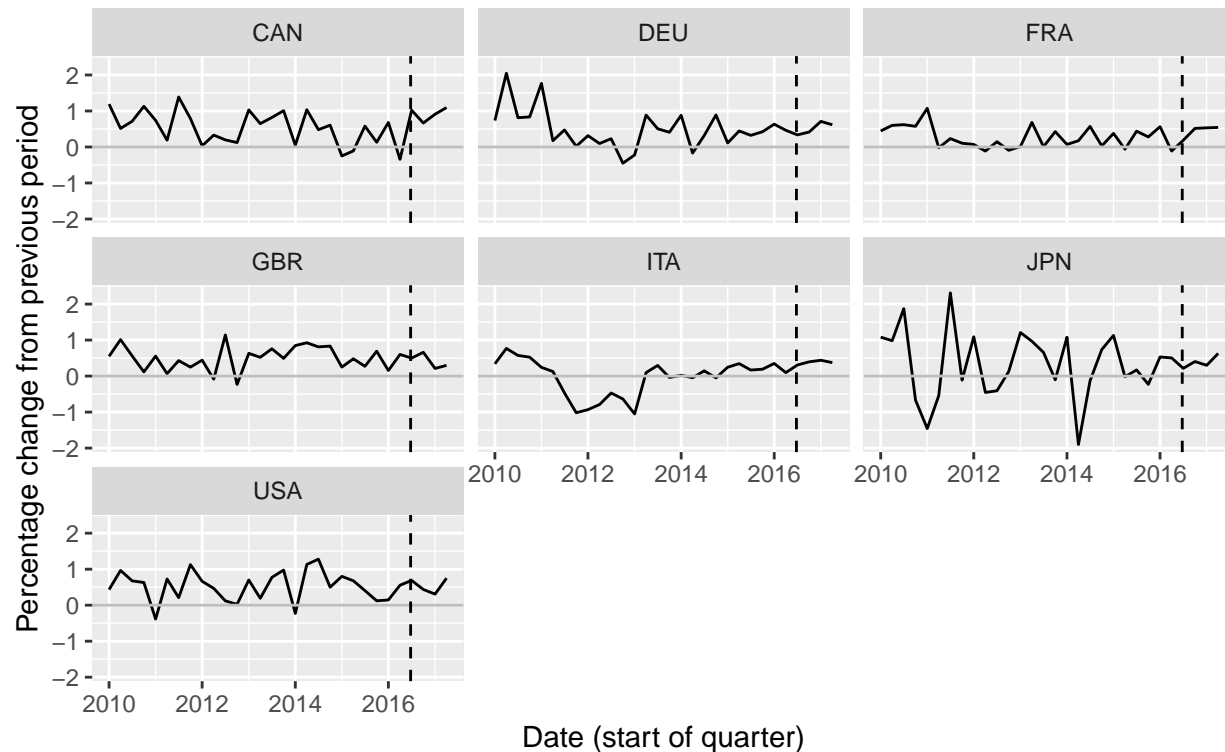
A random sample from the data set:

Country	Year	Quarter	Date (start of quarter)	Percentage change from previous period	Date (start o quarter)
JPN	2015	2	2015-04-01	-0.019508	2015-04-01
CAN	2012	1	2012-01-01	0.031895	2012-01-01
CAN	2011	3	2011-07-01	1.387897	2011-07-01
GBR	2016	2	2016-04-01	0.601298	2016-04-01
CAN	2010	1	2010-01-01	1.191352	2010-01-01

```
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")
```

Quarterly GDP Growth in Relation to Brexit Vote

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	2017	1	2017-01-01	0.307393	2017-01-01
JPN	2017	2	2017-04-01	0.629149	2017-04-01
DEU	2013	2	2013-04-01	0.886494	2013-04-01
JPN	2016	4	2016-10-01	0.403650	2016-10-01
DEU	2013	4	2013-10-01	0.408634	2013-10-01

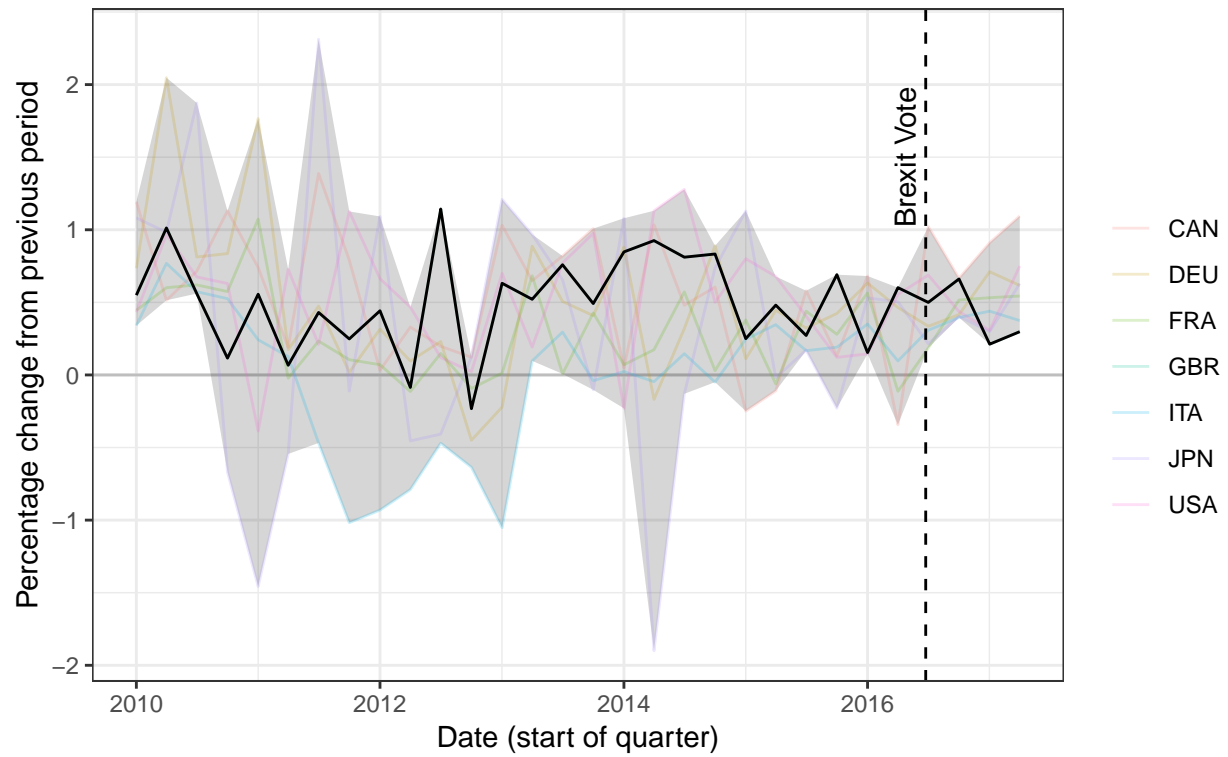
```
ggplot(data = data) +
  aes(x = `Date (start of quarter)`) +
  aes(y = `Percentage change from previous period`) +
  aes(ymin = min_) +
  aes(ymax = max_) +
  geom_hline(yintercept = 0, col = "grey") +
  geom_ribbon(alpha = .2) +
  geom_line(aes(col = Country), alpha = .2) +
  geom_line(data = data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



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 afternoon d
The White Hart	Aylesbury	51.81533	-0.8097584	8.75	7.40	5.
Sandford House	Huntingdon	52.33042	-0.1871106	8.75	7.59	6.
The Pear Tree	Birmingham	52.43198	-1.8928248	8.75	5.40	5.
Woodrow Wilson	Carlisle	54.89075	-2.9307650	8.75	7.19	5.
The Hornet	Birmingham	52.49268	-1.8194877	8.75	5.80	5.

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

A random sample from the data set:

	long	lat	group	order	region	subregion
31356	131.06856	42.902245	430	31356	China	NA
48165	107.59160	-2.976563	755	48165	Indonesia	Belitung
3372	-99.43433	-72.406639	72	3372	Antarctica	Thurston Island
1289	-58.08242	-26.138575	20	1289	Argentina	NA
19133	-103.47412	68.115036	345	19133	Canada	NA

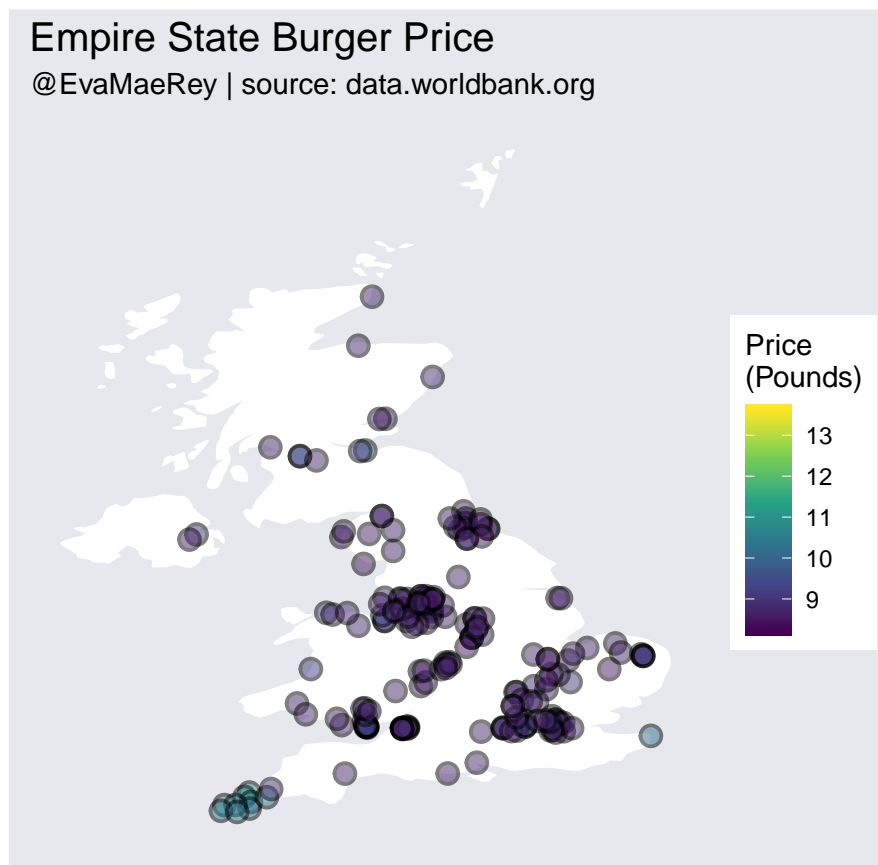
```
# create a blank ggplot theme
theme_opts <- theme(
  panel.grid.minor = element_blank(),
  panel.grid.major = element_blank(),
  panel.background = element_blank(),
  plot.background = element_rect(fill = "#e6e8ed"),
  panel.border = element_blank(),
  axis.line = element_blank(),
  axis.text.x =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

ggplot(data = world_map_df %>% filter(region == "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 Burger`,
                           colour = "black", shape = 21, stroke = 1, alpha = .5, size = 3)
  ) +
  labs(fill = "Price\n(Pounds)") +
  labs(title = "Empire State Burger Price") +
  labs(subtitle = "@EvaMaeRey | source: data.worldbank.org") +
  theme_opts

```



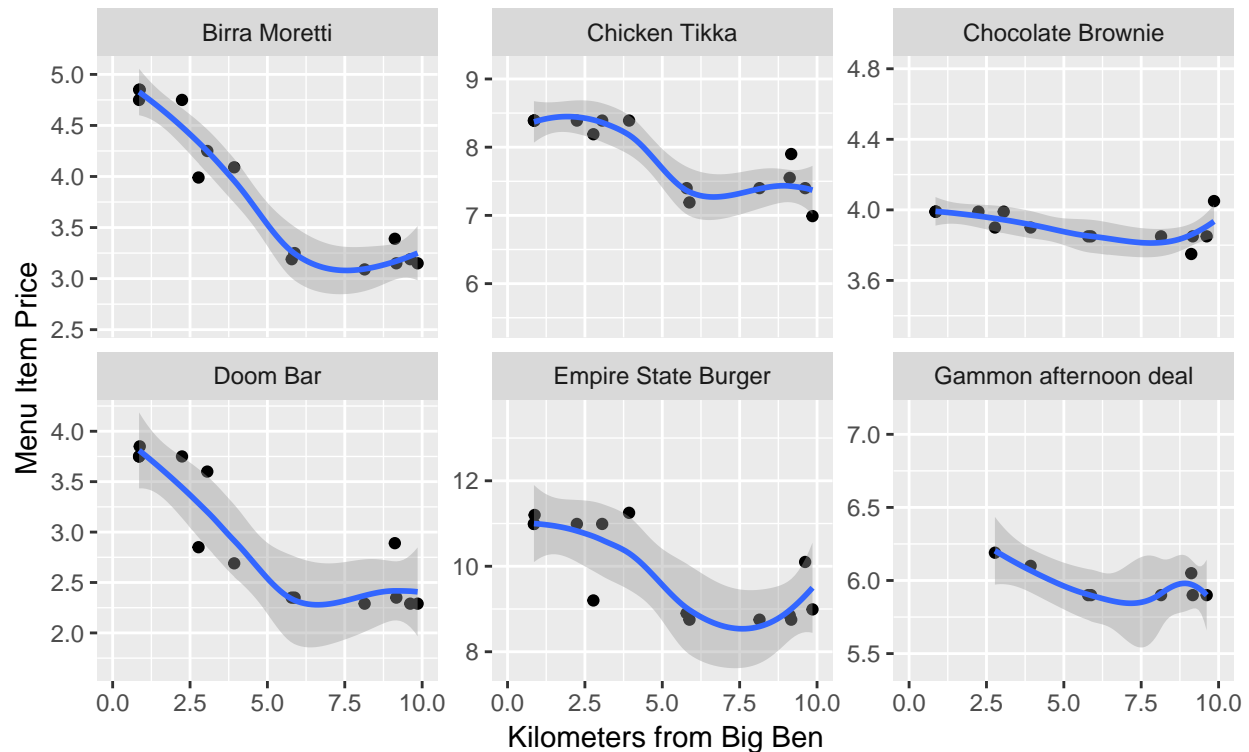
A random sample from the data set:

Name	Location	Latitude	Longitude	Notes	Moretti as a % of a tikka	Moretti as % of
JJ Moon's	Tooting, London	51.42787	-0.1684140	NA	0.4506438	0.3
The Hornet	Birmingham	52.49268	-1.8194877	NA	0.4741379	0.3
The Asparagus	Battersea, London	51.47000	-0.1713948	NA	0.4520167	0.3
The JP Joule	Sale	53.42406	-2.3175328	NA	0.4310811	0.3
The Old Swanne Inne	Great Malvern	52.13869	-2.3000250	NA	0.4297636	0.3

```
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")
```

Wetherspoon Pubs' Menu Item Prices v. Distance from Big Ben

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 equivalence (a 45° angle) is used as a reference and shows the result that you would see if there were 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:

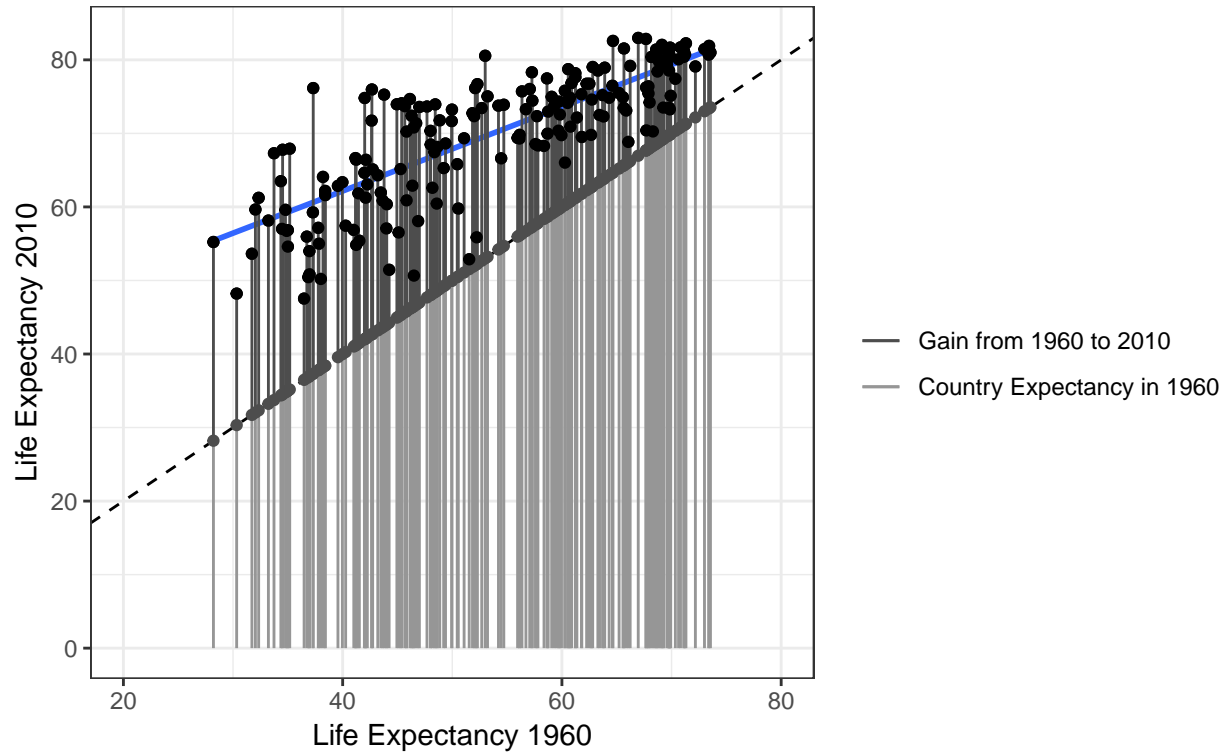
Life Expectancy 1960	Country Code	Country Name	Region	Income Group	Year
72.17659	DNK	Denmark	Europe & Central Asia	High income	2010
48.01371	EGY	Egypt, Arab Rep.	Middle East & North Africa	Lower middle income	2010
45.38441	TUR	Turkey	Europe & Central Asia	Upper middle income	2010
36.87241	CIV	Cote d'Ivoire	Sub-Saharan Africa	Lower middle income	2010
46.99883	NIC	Nicaragua	Latin America & Caribbean	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) +
  # coord_fixed() +
  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("grey59", "grey30", "grey30")
  ) +
  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



Chapter 12

Myers Briggs

This data looks at the relationship between 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.

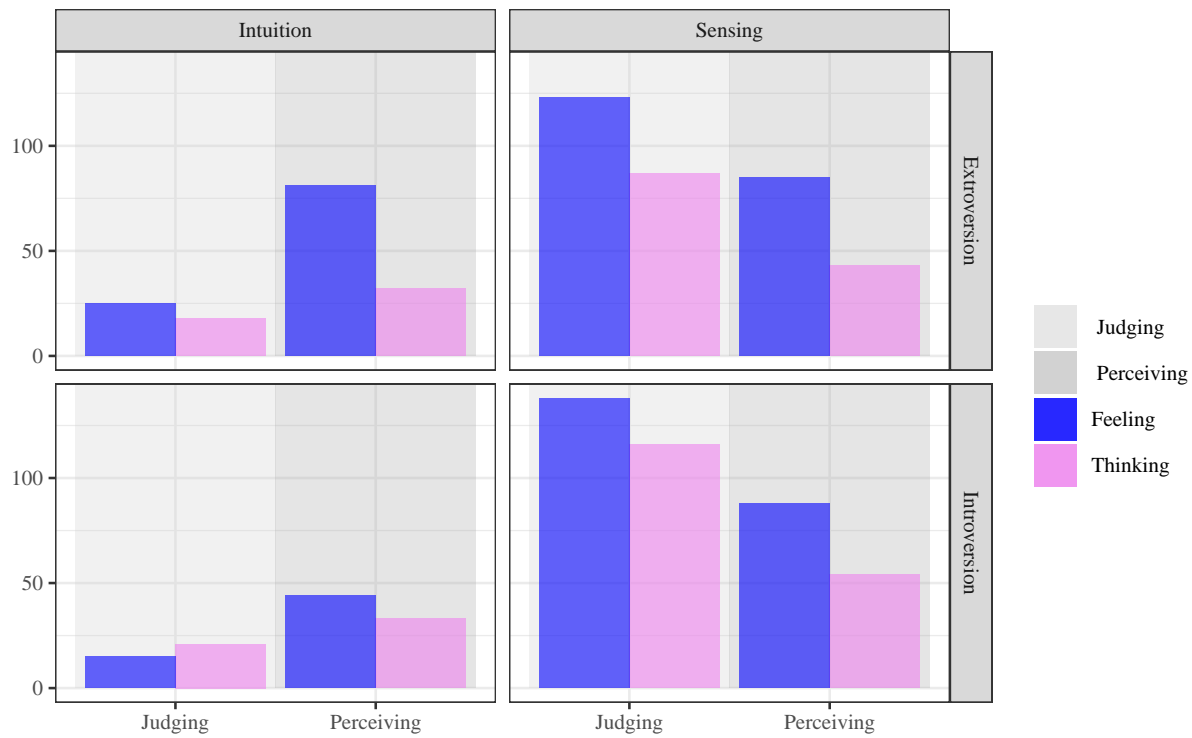
A random sample from the data set:

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

```
ggplot(d) +
  aes(x = `(J)udging/(P)erceiving`) +
  aes(fill = `(T)hinking/(F)eeling`) +
  facet_grid(`(E)xtroversion/(I)ntroversion` ~
             `(S)ensing/(I)ntuition`) +
  geom_rect(aes(x = NULL, y = NULL,
                xmin = mins, xmax = max,
                fill = `judging perceiving`),
            ymin = 0, ymax = 700, data = background
  ) +
  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/>



Chapter 13

Wine

Wine production in Europe may have been volatile 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 volatility early in this period.

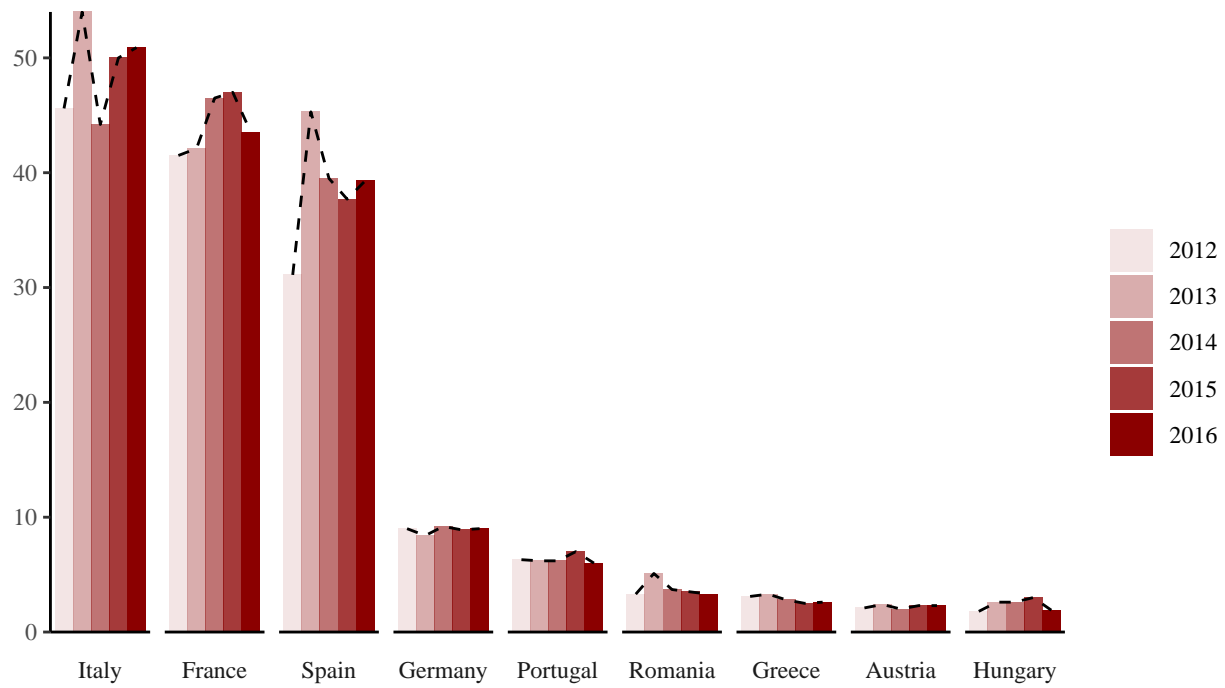
```
df <- readxl::read_xlsx("raw_data/Wine_Production_by_country.xlsx") %>%  
  filter(Country != "World total")
```

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

```
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  
theme_classic(base_family = "Times") +  
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)  
)
```

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

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 inconsistent number of days in each year. There is a measurement for every day, but leap years contain an 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	mean
2015-06-06	11.323	2015	06-06	2000-06-06	0.0314528	
1985-03-29	15.957	1985	03-29	2000-03-29	0.0443250	
1996-12-18	13.106	1996	12-18	2000-12-18	0.0364056	
2017-01-02	12.614	2017	01-02	2000-01-02	0.0350389	
2017-04-03	14.227	2017	04-03	2000-04-03	0.0395194	

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

```
# 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 = c(0, 0)
  ) +
```

```

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 calendar-year average extent of Arctic sea ice.",
  col = "white",
  size = 7
) +
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(
  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")
)

```

