Plotting Data with `ggplot2`

Carlos L. Yordán

Associate Professor of International Relations

Drew University

June 3, 2022

Overview:

ggplot2 is a plotting package developed for **R** by Hadley Wickham, based on Leland Wilkinson's book: *The Grammar of Graphics* (1999).

The package is part of the **tidyverse**, which includes **dplyr**, **tidyr**, **stringr**, etc. To install this package, you can either:

```
install.packages (tidyverse)

Install.packages (ggplot2)
```

For more information, visit the package's website: https://ggplot2.tidyverse.org.

Why use `R` and `ggplot` for Plotting Rather than Excel or Google Sheets?

`R` was built as a statistical program that allows users to reproduce their workflows.

Consistency matters! Once an user develops a code that produces a particular visualization, he or she can just keep pipping new data objects and the output's "look" will be the same.

Plotting in `R` is not easy at first, but it is worth investing the time and effort to learn its unlimited potential.

A Quick Example

Question & A Glimpse of the Data:

Let’s use my project’s dataset.

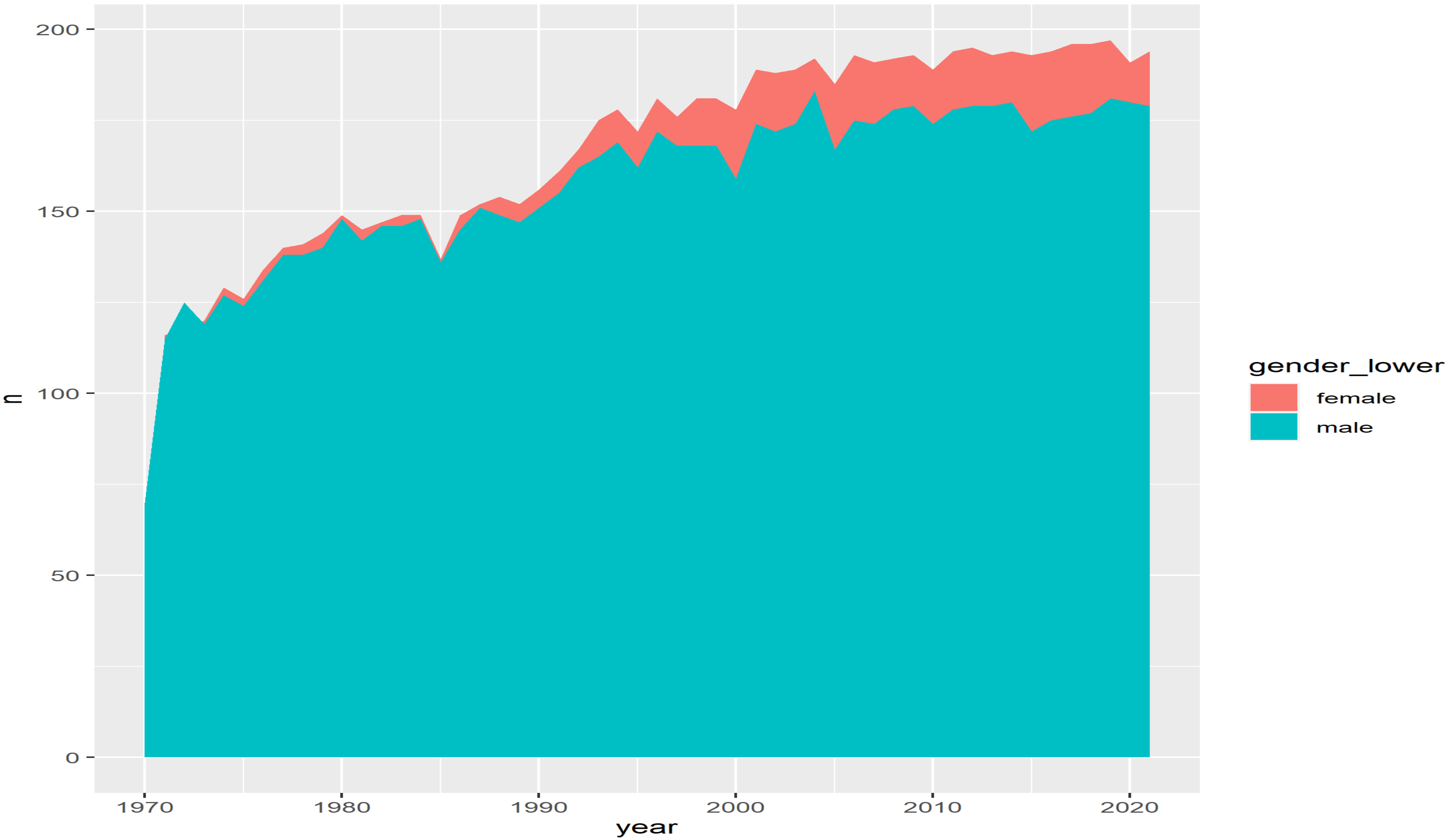
It describes who speaks at the United Nations General Debate, an annual event.

The dataset includes information on world leaders’ gender, their last name, the position in their government and the type of government they represent and so forth.

For this example, we want to measure how many women have delivered their countries’ speeches at this event.

year	gender_lower	n
1970	male	70
1971	female	1
1971	male	115
1972	male	125
1973	female	1
1973	male	119
1974	female	2
1974	male	127

The Basic Geom_area Plot:

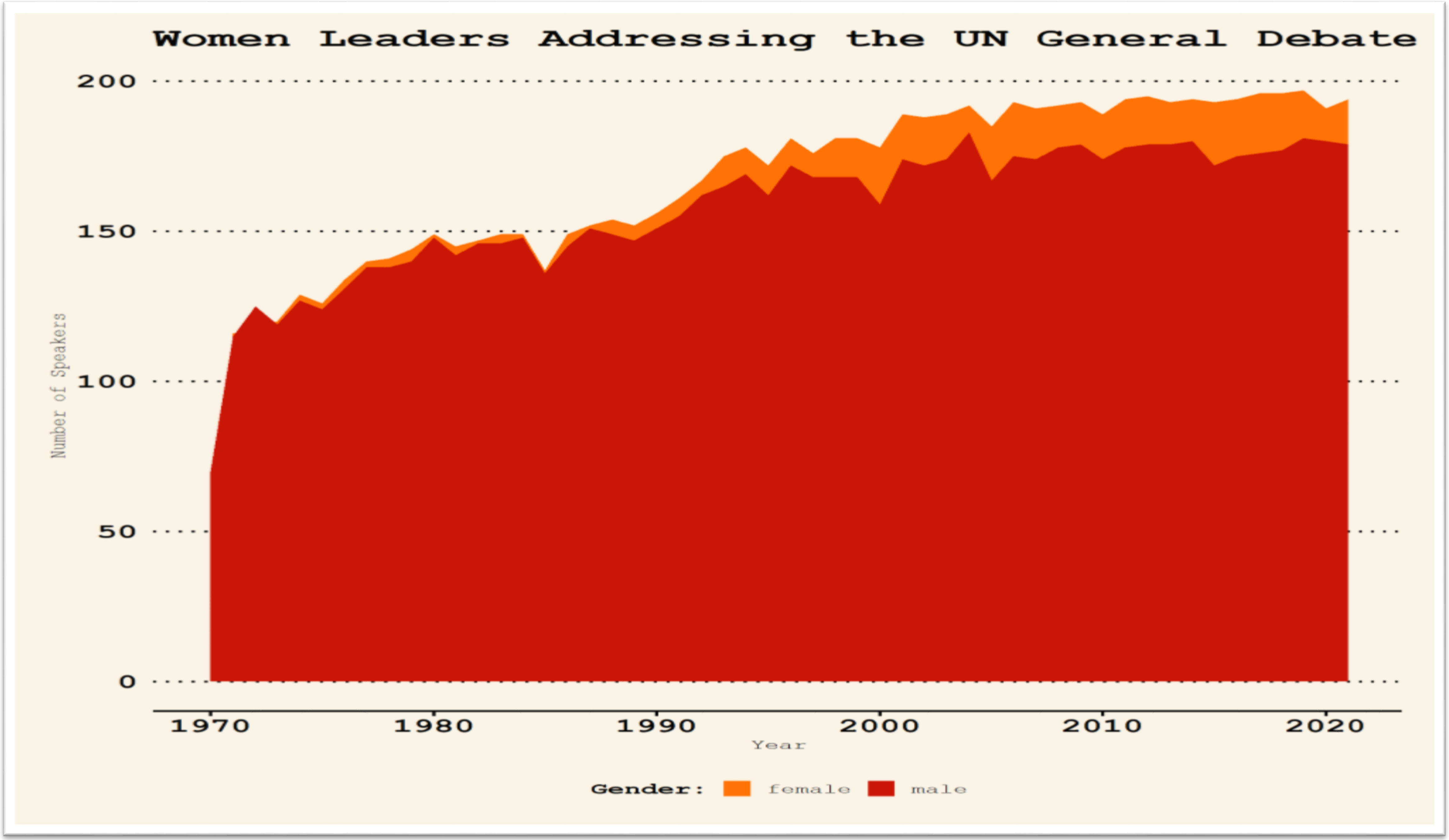


The Code for a Basic Plot

```
ggplot (gender_count, aes(x=year, y = n, fill=gender_lower)) +
      geom_area (stat = "identity", position= "stack")
```

Here is the Code for the Final Geom_area Plot:

```
ggplot (gender_count, aes(x=year, y = n, fill=gender_lower)) +
  geom_area (stat = "identity", position= "stack") +
  theme_wsj () +
  scale_fill_futurama () +
  labs (fill= "Gender:")+
  theme (text = element_text(family= "mono"))+
  theme (legend.position = "bottom",
         legend.title = element_text (size=10, face ="bold"),
         legend.text = element_text(size = 8),
         legend.key.size = unit(.75,"line"))+
  theme (plot.title = element_text(size=16, face="bold"),
         axis.title = element_text (size=8))+
  labs (title= "Women Leaders Addressing the UN General Debate",
        x = "Year",
        y = "Number of Speakers")
```


Women Leaders Addressing the UN General Debate
200
150
100
50
0
Number of Speakers
1970
1980
1990
2000
2010
2020
Year
Gender:
female
male

The Basic Code

```
ggplot (gender_count, aes(x=year, y = n,
fill=gender_lower)) +
   geom_area (stat = "identity", position=
"stack")
```

The Final Code

```
ggplot (gender_count, aes(x=year, y = n,
fill=gender_lower)) +
  geom_area (stat = "identity", position=
"stack") +
  theme_wsj () +
  scale_fill_futurama () +
  labs (fill= "Gender:")+
  theme (text = element_text(family= "mono"))+
  theme (legend.position = "bottom",
         legend.title = element_text (size=10,
face ="bold"),
         legend.text = element_text(size = 8),
         legend.key.size = unit(.75,"line"))+
  theme (plot.title = element_text(size=16,
face="bold"),
         axis.title = element_text (size=8))+
  labs (title= "Women Leaders Addressing the UN
General Debate",
        x = "Year",
        y = "Number of Speakers")
```

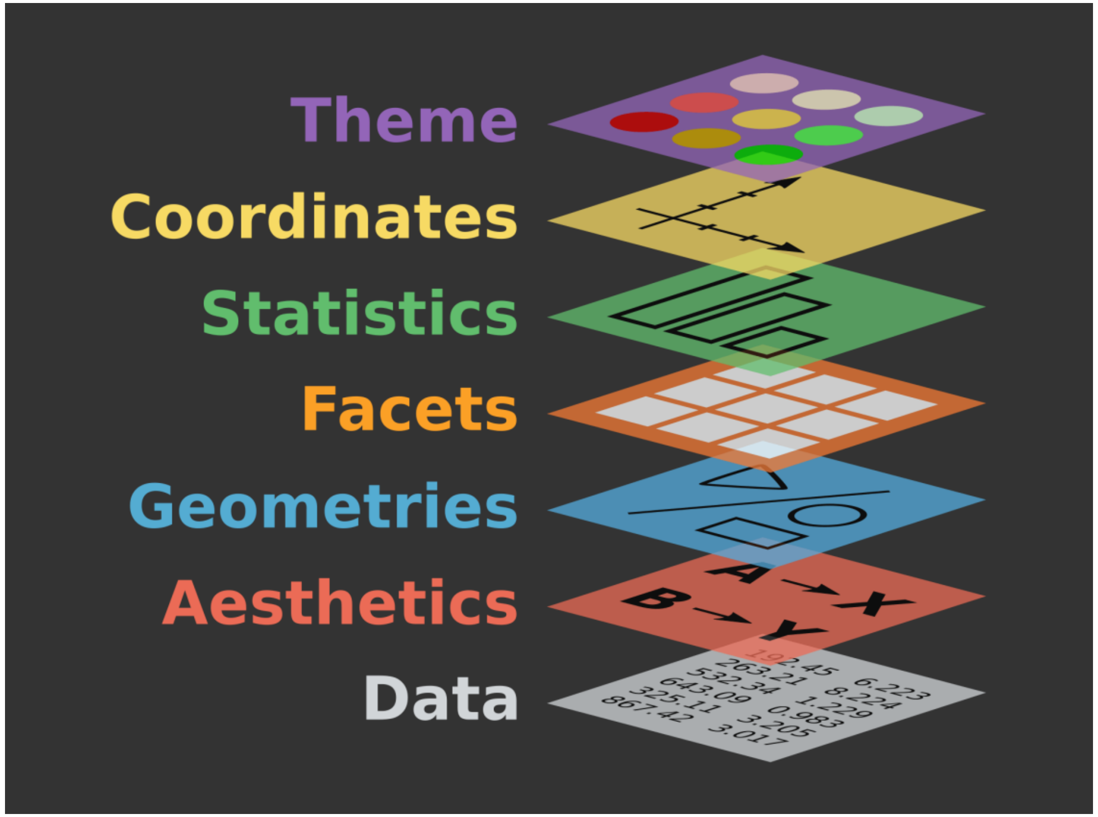
A System Based on Layers:

At first glance, `ggplot2`'s code seems to be counter-intuitive.

But, this is not the case!

`ggplot2` allows us to build a plot step-by-step.

We first start with data and end with the theme.



Source: Quebec Centre for Biodiversity Science (2022), Chapter 5 in *Workshop 3: Introduction to Data Visualization*.

A Summary of `ggplot2`'s Most Common Layers

Data	Aesthetics (aes)
The data should be structured in a tidy format, where each column is a variable and each row and observation. Using the tidyverse’s collection of packages is the best way to prepare the data for visualization	**x**, **y**, where x is the dependent variable and y the independent variable. **color**: specifies the color used by the geom **fill**: used to color the inside of a geom **shape:** the figure used to plot a point **linetype:** the type of line used in geom_line (e.g. solid, dashed, etc…) **size:** used to determine the scale of a dimension **alpha:** sets the transparency of the geom

Geometric Objects	Facets	Statistics
geom_point (): scatterplot **geom_line ():** lines connecting points by increasing the value of x **geom_path ():** lines connecting points in a particular sequence **geom_boxplot ():** box and whisker plots used for categorical variables **geom_bar:** bar charts **geom_area:** area charts **geom_histogram:** histogram for a continuous x axis **geom_smooth:** function based on a scatterplot to run a smooth regression line	**facet_wrap ()** **facet_grid ()**	Like **geoms**, but **`ggplot2`** can compute different statistical functions, such as counts, percentages, means and so forth.

Coordinates:

coord_cartesian: sets limits within the graph's area.

coord_polar: used to develop cicular plots.

coord_map: used to create geographical maps.

Themes:

These set of instructions allows the user to add fonts, colors, shapes and outlines that will enhance the plot's final result.

It is worth noting that ggplot2 comes with a number of pre-formatted themes. Other packages such as: **`ggthemes`** add many more pre-formatted themes.

Users can also develop their own custom theme.

Adding these layers together:

```
ggplot (data = dataset, aes (x = variable_name, y = variable.name, fill =
variable name) +
     Geom_* () +
     Coord_flip () +
     Theme ()
```

data

aesthetics

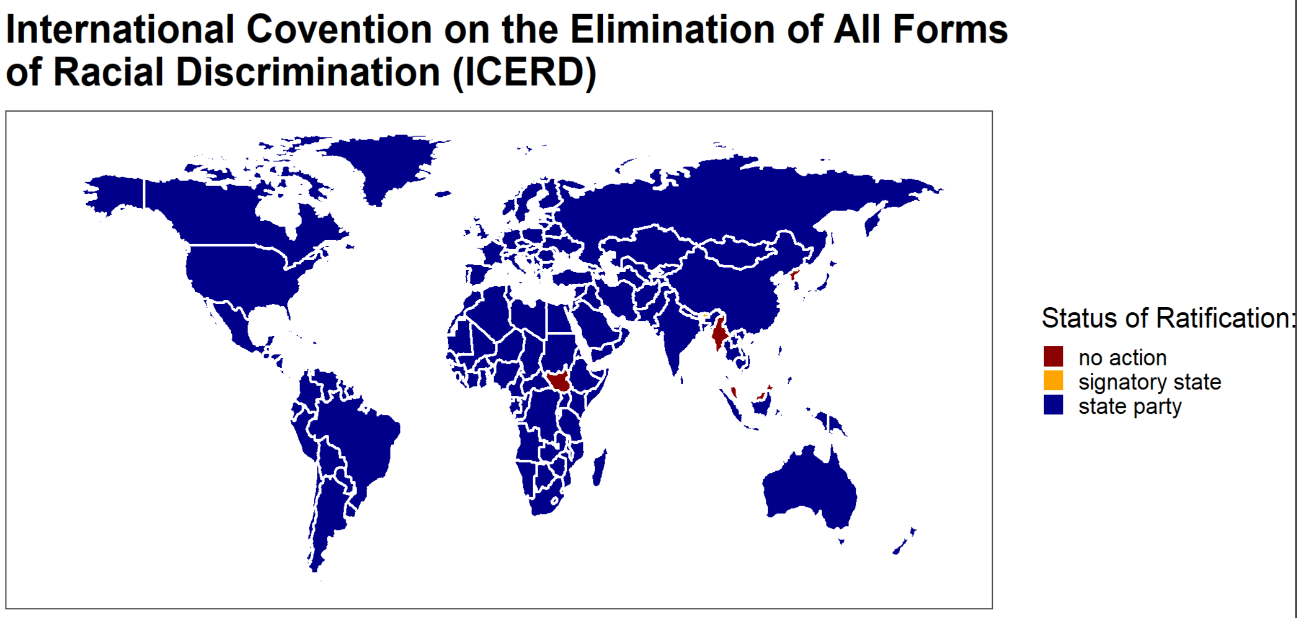
+ signs connects the layers

```
ggplot (gender_count, aes(x=year, y = n, fill=gender_lower)) +
  geom_area (stat = "identity", position= "stack") +
  theme_wsj () +
  scale_fill_futurama () +
  labs (fill= "Gender:")+
  theme (text = element_text(family= "mono"))+
  theme (legend.position = "bottom",
         legend.title = element_text (size=10, face ="bold"),
         legend.text = element_text(size = 8),
         legend.key.size = unit(.75,"line"))+
  theme (plot.title = element_text(size=16, face="bold"),
         axis.title = element_text (size=8))+
  labs (title= "Women Leaders Addressing the UN General
Debate",
        x = "Year",
        y = "Number of Speakers")
```

geom_*

ggthemes theme

theme


International Covention on the Elimination of All Forms
of Racial Discrimination (ICERD)
Status of Ratification:
🟥 no action
🟧 signatory state
🟦 state party

Code:

```
ggplot(wmap_hr, aes(long, lat, group = group)) +
  coord_fixed(1.3)+
  geom_polygon(aes(fill = status))+
  geom_polygon(data =wmap_hr, colour = "white", fill =
NA) +
  ggtitle("International Covention on the Elimination of
All Forms\nof Racial Discrimination (ICERD)") +
  scale_y_continuous()+
  scale_fill_manual (values = c("darkred", "orange",
"darkblue"))+
  theme_few ()+
  theme(axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        axis.ticks = element_blank(),
        rect = element_blank())+
  theme(plot.title = element_text(size = 14))+
    theme(plot.title = element_text(face = "bold"))+
    theme (legend.position = "right",
             legend.title = element_text(color = "black",
size=10),
             legend.text = element_text (color = "black",
size =8),
             legend.key.size = unit(0.3, 'cm'),
        legend.key.height = unit(0.3, 'cm'),
        legend.key.width = unit(0.3, 'cm'),
             )+
  labs (fill = "Status of Ratification:",
        x="",
        y= "")
```

DO YOU HAVE ANY QUESTIONS BEFORE, WE JUMP INTO R?