Exploring the use of ggplot2 in data visualization

Introduction

If you have ever visited the current events news site fivethirtyeight you would have no doubt seen the plethora of graphs and charts available in each post. Fivethiryteight updates readers on events related to sports, economics, politics, and culture with an extensive statistical analysis to back up their facts. When I first visited the site, I remember being very impressed by the qualities atrnd diversity of the graphs displayed. Since taking this class, I have discovered that one of the main data visualization tools that fivethrirtyeight.com uses is what we have learned in class, ggplot2.

Motivation

Ever since learning ggplot2, I have become much more interested and consumed with the data visualization part of any project. The use of ggplot2 can be used everywhere from scientific research projects to business proposals. Not only is ggplot2 very visually appealing compared to the base graphing package, but it also provides a lot of features that are not possible on the base package that allow for more unique and create ways of representing data. Now more than ever, people are willing to go to extensive lengths to collect data, and with all of this data lying around, it is important for us to find the best ways to interpret and visualize it.

History

Hadley Wickham created ggplot2 back in 2005 as a data visualization package for R. His implementation of it was a result of Leland Wilkinson’s Grammer of Graphics a general scheme for visualization that breaks up graphs into semantic components such as scales and layers. There are several features which make ggplot2 unique compared to the base package in R. Ggplot2 allows the user to add, remove, or alter components in a plot at a high level of abstraction; however, this abstraction can cause ggplot2 to be slower than lattice graphics. What it lacks in speed, it makes up for in features. Unlike the base package, ggplot2 allows for a high degree of modularity- the same underlying data can be transformed by many different scales and layers.

Let’s see what ggplot2 can do!

First we will load all the packages and data sets needed in order to go through the proceeding examples.

```r
library(ggplot2)
#install.packages(c("maps", "mapdata"))
library(maps)
library(mapdata)
library(readr)     # importing data
library(dplyr)     # data wrangling
```

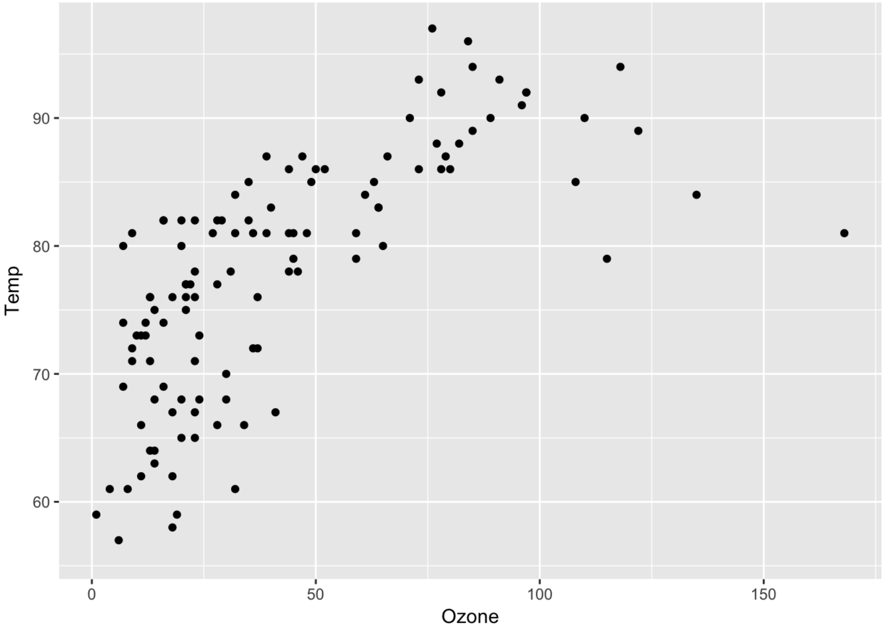
```
## Warning: package 'dplyr' was built under R version 3.4.2
```

```r
#github <- "https://github.com/ucb-stat133/stat133-fall-2017/raw/master/"
#csv <- "data/nba2017-players.csv"
#download.file(url = paste0(github, csv), destfile = 'nba2017-players.csv')
teams <- read.csv('nba2017-players.csv', stringsAsFactors = FALSE)
usa <- map_data('usa')
states <- map_data('state')
airquality <- airquality
```

Some of my favorite features of ggplot2 are how easily the code is read, the faceting aspect, the diversity in how to represent specific points, and the overall visual appearance of it. In terms of reading the code, ggplot2 constructs things based on the ‘aesthetic mapping’ as denoted by aes() as can be seen below. These aesthetic mappings describe how variables in the data are mapped to visual properties (aesthetics) of geoms. In the example below, I used a data set that gave information about the air quality in New York. By using ggplot2, it is easy to see the relationship between the temperature and the ozone levels.

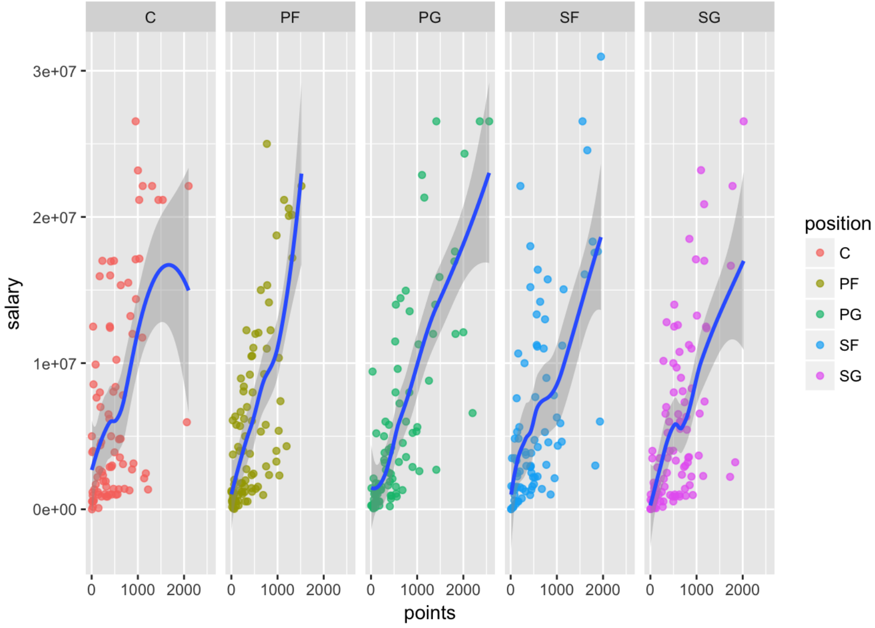
```r
# a scatterplot of Ozone and Temperature from the dataset about New York
ggplot(data = airquality, aes(x= Ozone, y = Temp)) +
  geom_point()
```

```
## Warning: Removed 37 rows containing missing values (geom_point).
```



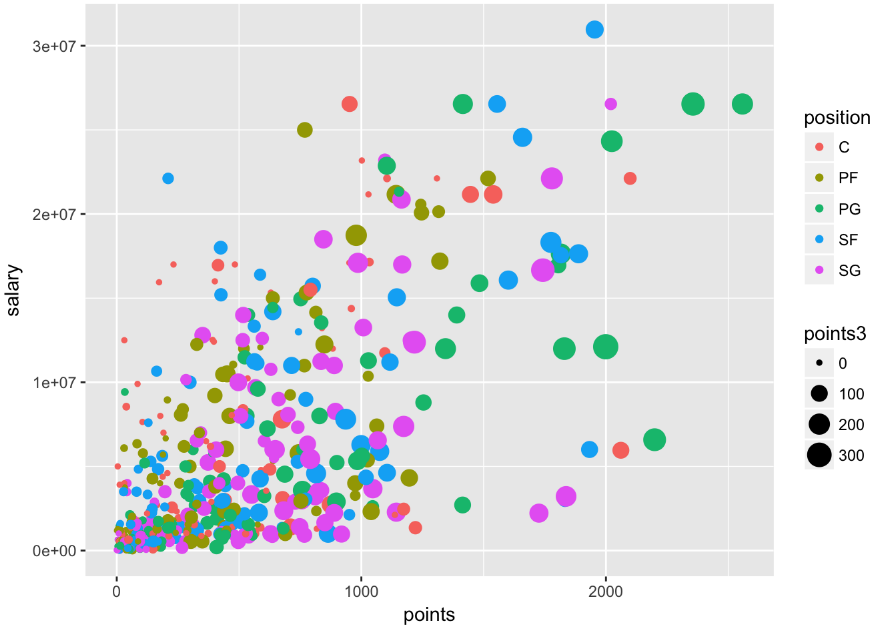
The faceting aspect of ggplot2 is something I hadn't seen before in my limited experience with Python. It was so fascinating to me that ggplot2 could just split up data by a specific grouping and then show each separately on a graph without the user having to individually write code for each grouping. Here I use the data about NBA teams to look at the association between points scored and salary earned, the faceting aspect here allows me to view these correlation by grouping the data by position of the player. I have also included a loess line to see the general correlation. You can see on the right hand side that ggplot2 automatically creates a key for the positions.

```
# a scatterplot faceted with postion
ggplot(data = teams, aes(x = points, y = salary)) +
  geom_point(aes(color = position), alpha = 0.7) +
  facet_grid(~ position) +
  geom_smooth(method = loess)
```



Another thing that I found useful and unique in terms of visualizing data was the ggplot2 aspect that allowed you or color or change the size of each data point based on certain specifications. This contributes to the overall appearance of the graph and is something that the base package cannot do as easily. Here it is easy to code and also the result of it allows for users to easily draw conclusions just by looking at the automatically provided key. Here, also using the NBA data, I was looking at the scatterplot of salary and points but also made the distinction by position of the player by the color of the dots and the number of 3 pointers scored as differentiated by the size of the points.

```
# scatterplot with positions and 3 pointers differentiated
ggplot(data = teams, aes(x = points, y = salary)) +
  geom_point(aes(color = position, size = points3))
```

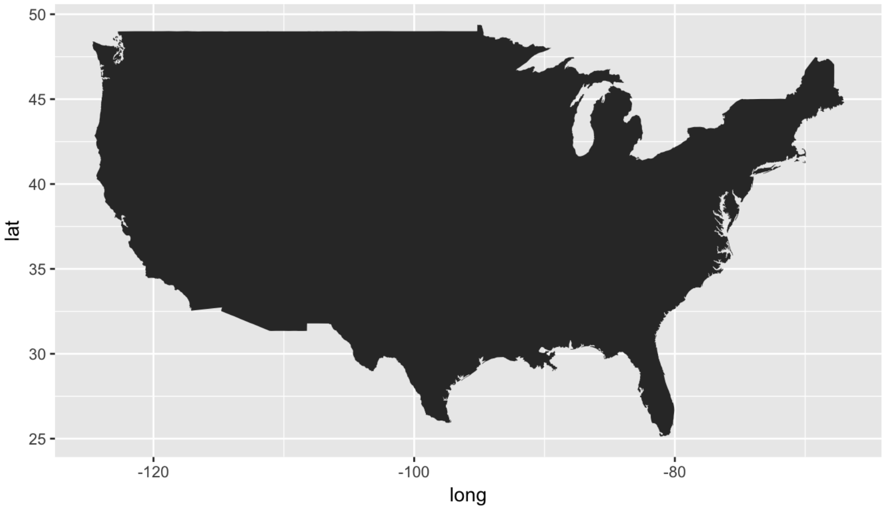


Some of the features of ggplot2 that we had not explored in class that I found during my research of this topic is the ability to create maps by using +geom_map, and its various possibilities with coordinate systems. I learned in multivariable calculus that points can be represented in different coordinate systems and in this class I see that ggplot2 can easily switch been said systems. For example, you can do +coord_cartesian to get a set of points in Cartesian coordinates and +coord_polar to get it in polar coordinates. Here I used ggplot to plot some maps of the US and states.

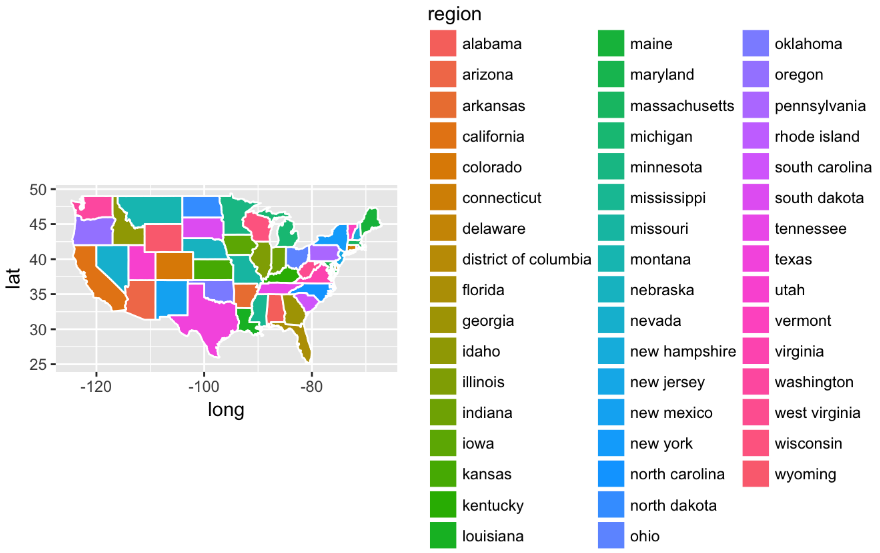
```
# let's look at some maps using ggplot2

usamap <- ggplot() + geom_polygon(data = usa, aes(x=long, y = lat, group = group)) +
  coord_fixed(1.3)
statesmap <- ggplot(data = states) +
  geom_polygon(aes(x = long, y = lat, fill = region, group = group), color = "white") +
  coord_fixed(1.3)
```

```
# our map of the US
usamap
```



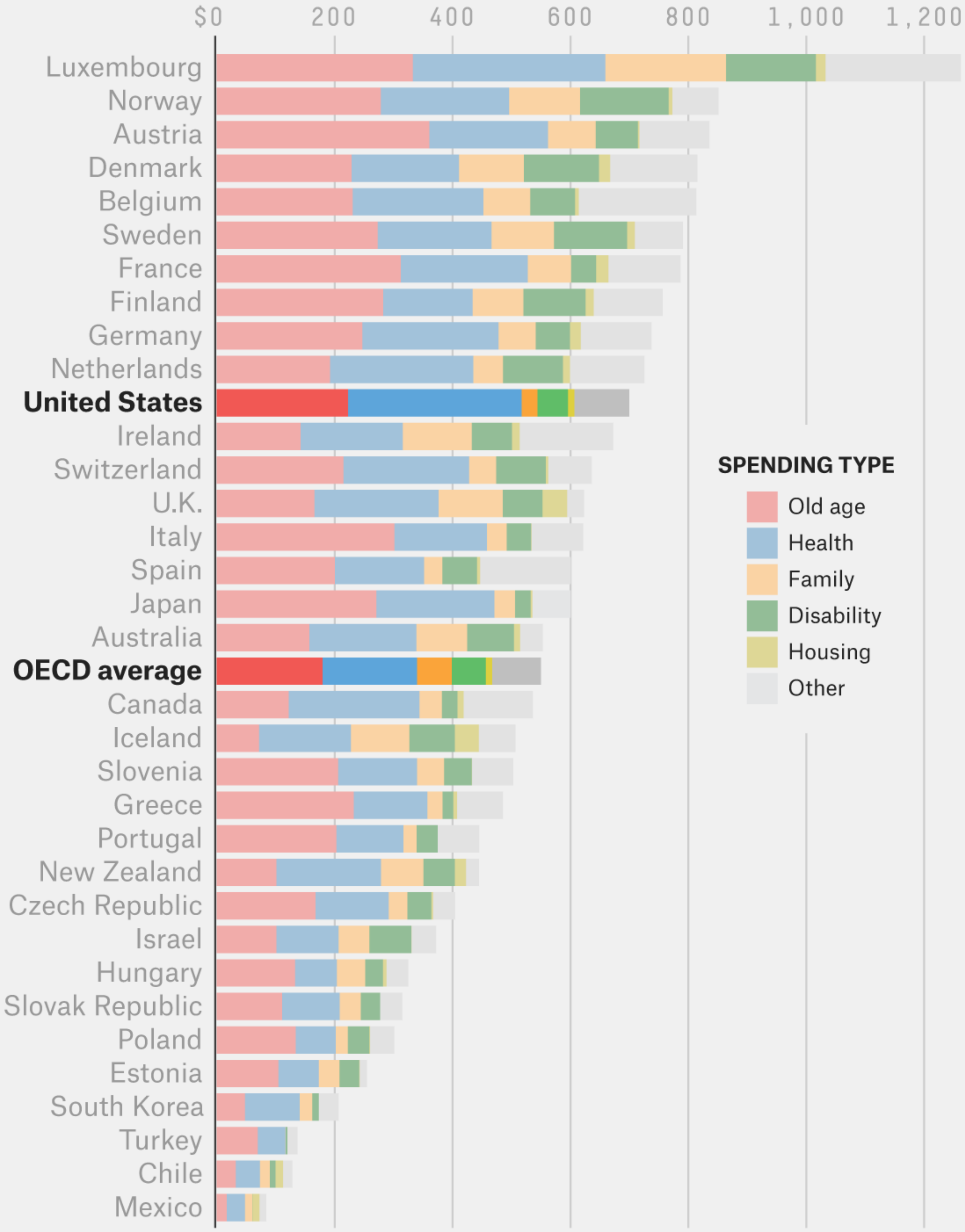
```
#our map of the states
statesmap
```



Where I have seen ggplot2 most is, as said above, on fivethirtyeight.com. Here are some great examples that really show the potential of ggplot2. It should be noted that along with ggplot2, fivethirtyeight also uses Illustrator to supplement the graphs.

What governments spend on social assistance

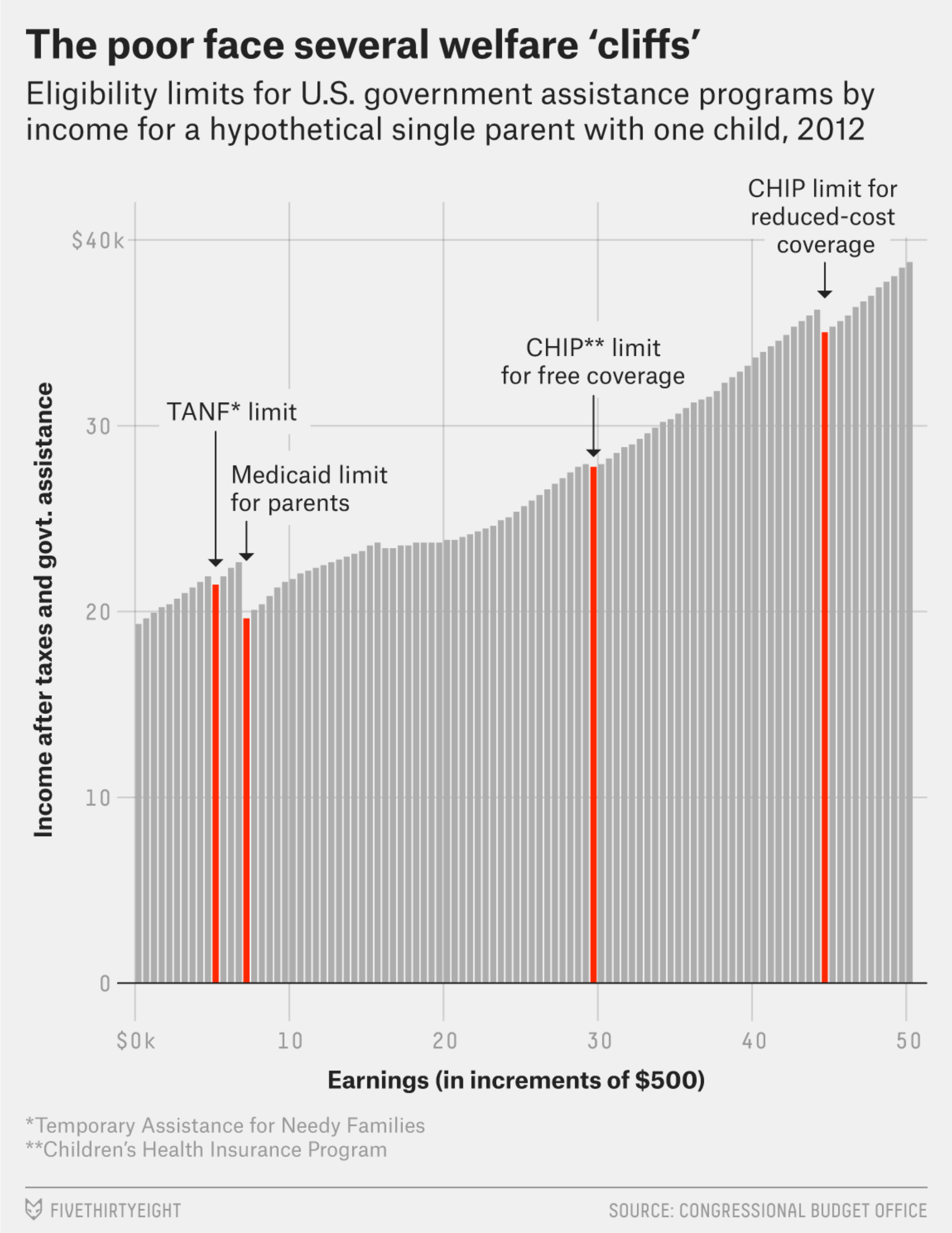
Per capita monthly public spending on social programs for OECD countries as of 2011



At constant prices and constant purchasing power parity, in 2005 U.S. dollars

FIVETHIRTYEIGHT

SOURCE: OECD



The poor face several welfare 'cliffs'

Eligibility limits for U.S. government assistance programs by income for a hypothetical single parent with one child, 2012

*Temporary Assistance for Needy Families
**Children's Health Insurance Program

FIVETHIRTYEIGHT

SOURCE: CONGRESSIONAL BUDGET OFFICE

Conclusion

Since ggplot2 is a data visualization tool, it application possibilities are endless. From science projects and school projects to business proposals and data exploration, ggplot2 can make the graphics for all these topics visually appealing, easy to understand, and relatively easy to make. It' is my favorite data visualization tool so far and I look forward to learning more about it and applying it to the new datasets we encounter. It's ability to scale and layer multiple data sets is what makes it unique and powerful.

References

- history about ggplot2
- images from fivethrityeight
- ggplot2 vs base
- pros of ggplot2
- what it means to layer data
- more about the aes
- a youtube video