Redesigining a Visualization

Cody Hilyard

August 2021

Introduction

As an undergraduate student studying Statistics at Utah State University, I completed two visualization courses centered on using data analysis to "make critical discoveries, and communicate them clearly" (see https://math.usu.edu/~symanzik/teaching/2020_stat5560/stat5560_001_overview.pdf). One assignment required transforming a "bad" graphic into a "good" graphic, according to principles discussed from *The American Statistician* (see https://www.jstor.org/stable/2683253?seq=1#metadata_info_tab_contents). The graphic that I selected illustrates the total number of deaths caused by shark attacks in various locations worldwide. According to the author's words, "In the continental U.S., Florida has the highest number of shark attacks for world locations with the highest shark attack activity". While the visualization illustrating this claim accurately shows it is true, the visualization is difficult to read with precision when looking at locations other than Florida:

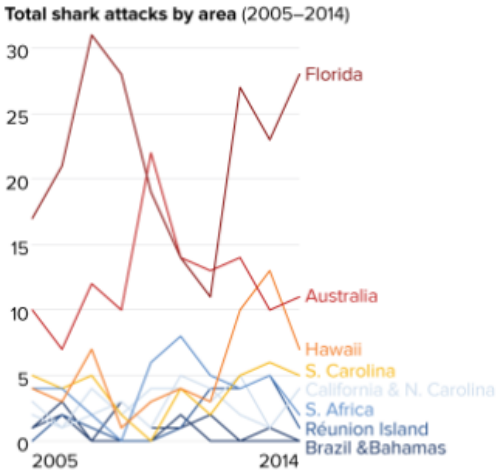


Figure 1: Source: https://abcnews.go.com/US/shark-attacks-told-graphs/story?id=31779076

Using the principles discussed by *The American Stastician*, the following reasons for re-designing the graphic were determined:

- *Rule 2: Hide What Data You Do Show.* The data is disorganized, specifically when there are 5 or less shark attacks. It is difficult to accurately determine the number of attacks per location per year. If the graph was printed in black and white ink, the graph would be comletely illegible. Also, speaking more specifically regarding issues with certain locations, it is unclear which lines individually belong to California and North Carolina. The lines for both locations are the same color but do not have identical data points. For other locations, there are different shades of the same colors which makes them difficult to differentiate.
- *Rule 10: Label (a) Illegibly, (b) Incompletely, (c) Incorrectly, and (d) Ambiguously.* The minimalistic approach for labeling the x-axis is problematic. Determining the number of attacks for years 2006 to 2013 is extremely difficult, unless you meticulously count the points individually as you follow the lines. This is only if you are able to accurately do so.
- *Rule 11: More is murkier.* See answer for *Rule 2: Hide What Data You Do Show*.

- *Rule 12: If It Has Been Done Well in the Past, Think of Another Way to Do It.* There are several graphing methods that would have been immensely easier to understand, one of which will be used in this document.

Procedure

First, data needs to be acquired. Since the graphic was found in an online news article, there isn't an accessible data set. Using the method of carefully following each point as described in *Rule 10* above, estimated values can be collected.

Step One: load required packages and create lists containing years, location names, and point estimates per location (lists are being used to easily categorize data for each location):

```
library(dplyr)         # simplifies data manipulation techniques
library(ggplot2)       # creates visualizations
library(kableExtra)    # creates tables
library(RColorBrewer) # enhances color selections

year <- rep(list(2005:2014), 10)

location <- list(rep("Australia", 10),
                 rep("Bahamas", 10),
                 rep("Brazil", 10),
                 rep("California", 10),
                 rep("Florida", 10),
                 rep("Hawaii", 10),
                 rep("North Carolina", 10),
                 rep("Reunion Island", 10),
                 rep("South Africa", 10),
                 rep("South Carolina", 10))

attack <- list(c(10, 7, 12, 10, 22, 14, 13, 14, 10, 11),
               c(1, 2, 0, 0, 0, 2, 0, 0, 0, 0),
               c(1, 3, 0, 3, 1, 1, 4, 2, 1, 0),
               c(3, 1, 4, 2, 4, 4, 3, 5, 1, 4),
               c(17, 21, 31, 28, 19, 14, 11, 27, 23, 28),
               c(4, 3, 7, 1, 3, 4, 3, 10, 13, 7),
               c(2, 1, 2, 3, 1, 5, 4, 2, 1, 4),
               c(0, 2, 1, 0, 0, 1, 4, 4, 5, 1),
               c(4, 4, 2, 0, 6, 8, 5, 4, 5, 2),
               c(5, 4, 5, 2, 0, 4, 2, 5, 6, 5))
```

Explanation of Variables:

- $year$ contains the years included in the original graph, repeated 10 times since there are 10 locations with shark attack counts for each year.
- $location$ contains the name of each location, repeated 10 times since there are 10 shark attack counts per location.
- $attack$ contains the estimated shark attack counts for each location and year. The values from left to right represent the estimated count for each year in chronological order (2005, 2006, ..., 2013, 2014), while the values from top to bottom represent the count for each location in alphabetical order (Australia, Bahamas, ..., South Africa, South Carolina).

Step Two: use a for-loop to extract data from each list and store relevant data together:

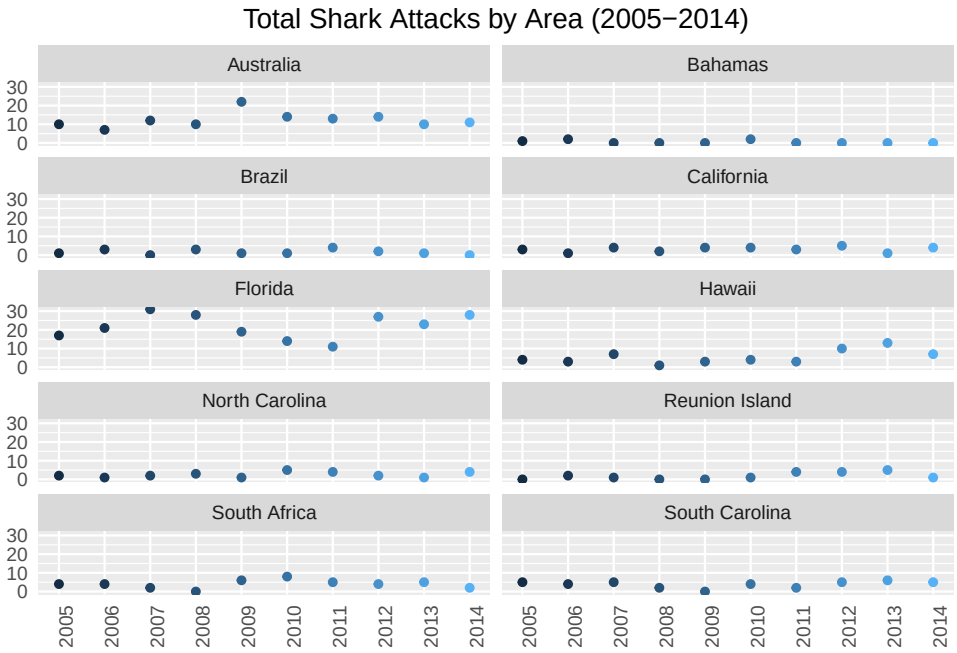
```
df <- data.frame()
for(i in 1:10) {
  attack_info <- data.frame(year[[i]], location[[i]], attack[[i]])
  df <- bind_rows(df, attack_info)
}

colnames(df) <- c("year", "location", "attack")
```

Below is sample of the data which has been created for one of the ten locations:

Year	Location	Attacks
2005	Australia	10
2006	Australia	7
2007	Australia	12
2008	Australia	10
2009	Australia	22
2010	Australia	14
2011	Australia	13
2012	Australia	14
2013	Australia	10
2014	Australia	11

Second and lastly, data needs to be visualized using a new technique:

```
ggplot(data = df, aes(x = year, y = attack, color = year)) +
  geom_point() +
  facet_wrap(~location, ncol = 2) +
  labs(title = "Total Shark Attacks by Area (2005-2014)",
       x = "", y = "") +
  scale_x_continuous(limits = c(2005, 2014),
                     breaks = seq(2005, 2014, by = 1)) +
  scale_y_continuous(limits = c(0, select(df, attack) %>% max()),
                     breaks = seq(0, select(df, attack) %>% max(), by = 10)) +
  theme(axis.text.x = element_text(angle = 90),
        axis.ticks = element_blank(),
        legend.position = "none",
        panel.grid.minor.x = element_blank(),
        plot.title = element_text(hjust = 0.5))
```



Results

Following the transformation of the original graphic, there is clearer evidence supporting the author's statement that "In the continental U.S., Florida has the highest number of shark attacks for world locations with the highest shark attack activity". While the high number of shark attacks in Florida is easily seen in both graphics, the new design allows better comparison between Florida and the other nine locations. With each set of results separated, comprehending which data points belong to which locations is no longer an issue. To ensure the visualization has optimal quality, let us revisit the principles from *The American Statistician*:

- *Rule 2: Hide What Data You Do Show.* The data is organized in a manner which allows the viewer to identify the yearly count of shark attacks for all ten locations without any challenge. Locations with identical shark attack counts for the same year no longer have the same line and color representing the same count.
- *Rule 10: Label (a) Illegibly, (b) Incompletely, (c) Incorrectly, and (d) Ambiguously.* The label for the x-axis lists each year from 2005 to 2014, enabling the viewer to accurately identify the number of shark attacks for all years, not only the first and last.
- *Rule 11: More is murkier.* Although there are 100 shark attack counts illustrated, the visualization looks 'clean' rather than 'murky'.
- *Rule 12: If It Has Been Done Well in the Past, Think of Another Way to Do It.* Separating out the data points for each location is a graphing method that aggregates data in a manner which is easy to understand.

References

- Create a more elegant-looking table (see https://cran.r-project.org/web/packages/kableExtra/vignettes/awesome_table_in_pdf.pdf).
- Place table of sample data where desired (see https://stackoverflow.com/questions/53153537/rmarkdown-setting-the-position-of-kable).
- Remove the x-axis minor gridlines (see https://stackoverflow.com/questions/56251993/remove-specific-vertical-gridline-from-ggplot).