Data Visualization

Getting Fancy: Drawing Maps

Ciara Zogheib

Data Sciences Institute, University of Toronto

In today's class, we will...

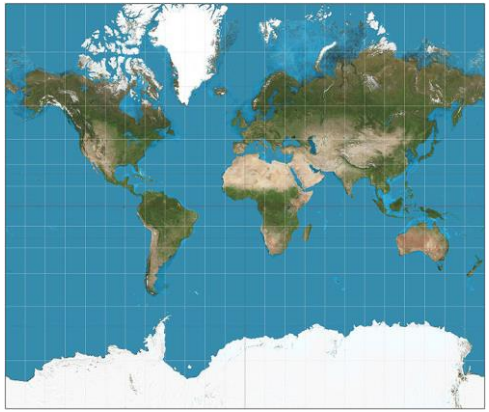
- Develop basic understanding of elements of maps
- Work through practice code from Chapter 7 (*Drawing Maps*) of Healy, K. (2018). Data Visualization: A Practical Introduction. Princeton University Press. This code will let us
    - Map choropleths with R
    - Explore alternate ways of grouping spatial data
    - Interrogate whether our data are truly spatial

Data visualization with maps: 101

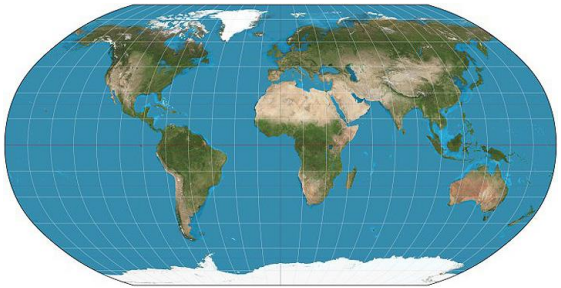
Maps and projections

- A map is “a representation usually on a flat surface of the whole or a part of an area”
- In order to display positions on the curved, 3-dimensional surface of the earth in a flat, 2-dimensional representation, our position data has to be transformed
- This transformation is called a projection
- The projection of a map impacts the map’s appearance (and therefore the conclusions that viewers will form)

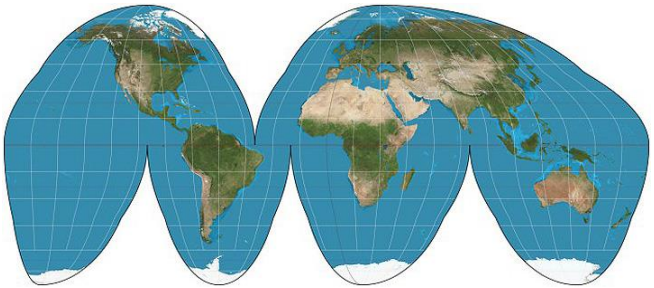
Maps and projections - Example



Mercator



Robinson



Goode’s Homolosine

Types of maps

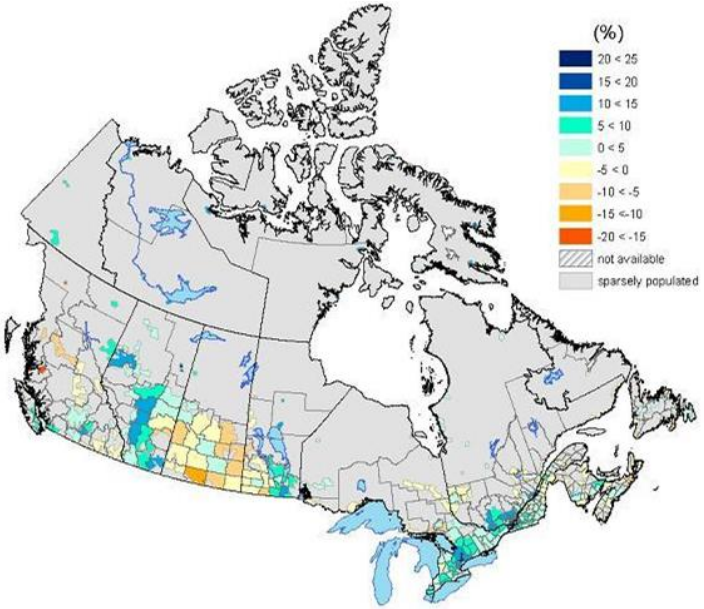
- Reference map → shows the locations, boundaries, and names of geographic areas and features
  - For example, the maps used to navigate on a road trip
- Thematic map → shows the "spatial distribution of one or more specific data themes for selected geographic areas"
  - For example, population density in each province, or percentage of college graduates in a neighbourhood
- When we visualize spatial data, we make thematic maps

(Statistics Canada, 2011)

Types of maps

Reference map showing topography of Canada

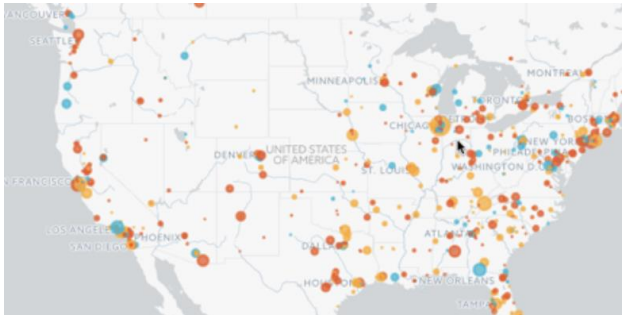
(Atlas of America, 2022)



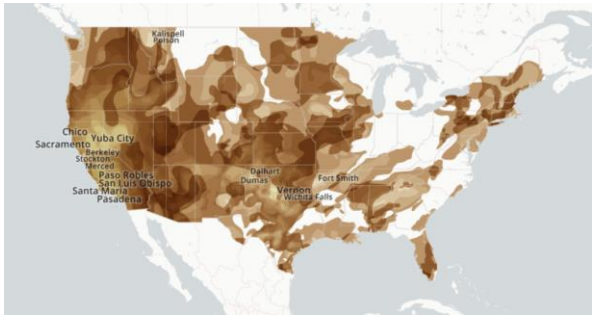
Thematic map showing population growth in Canada

(Statistics Canada, 2006)

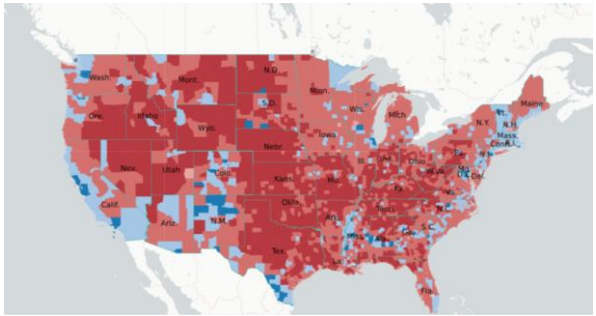
Thematic maps



Point map



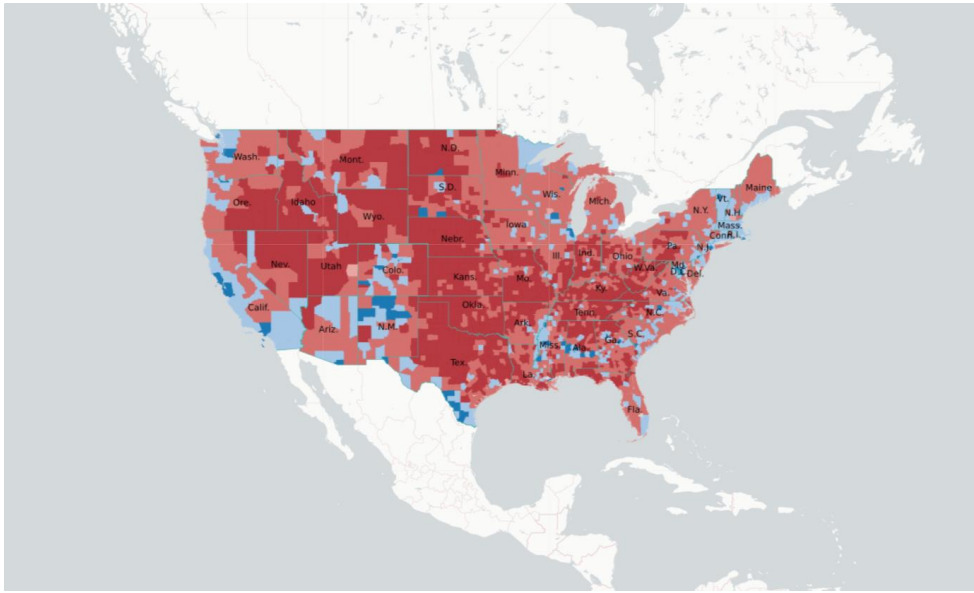
Heat map



Choropleth

Choropleths

- “Show geographical regions colored, shaded, or graded according to some variable”
- Often made using Geographical Information System (GIS) tools, but can be made using ggplot in R



(Healy, 2018)

Map state- and province-level data

Thematic map of U.S. election data

- To explore making choropleth maps in ggplot, we will again use our 2016 `election` dataset from the `socviz` library
- We will start by viewing random rows from a subset of our dataset:

```
election |> select(state, total_vote,
                   r_points, pct_trump, party, census) |>
  sample_n(5)
```

(Healy, 2018)

Thematic map of U.S. election data

- **Recall**: we have been working with spatial data throughout Module the class - it can be represented non-spatially!
- We can create dotplots of the point margins of our election data, faceted by census region, using methods we have learned so far.

(Healy, 2018)

Activity

- The next slide contains code (also available in Healy (2018)) for creating a faceted dotplot of 2016 U.S. election data by census region
- Recall: we discussed how commenting code helps to ensure reproducibility and understanding of what choices were made
- **Comment the following code so that a new viewer could understand what is being accomplished with each step**
- Hint: it can be helpful to run the code piecewise (one step at a time) to see what each step adds/changes!

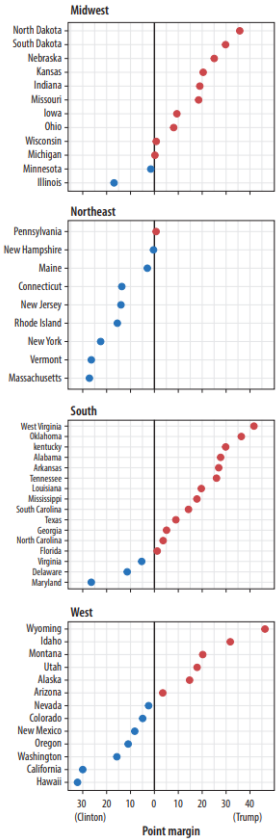
(Healy, 2018)

Activity

```
party_colors <- c("#2E74C0", "#CB454A")

p0 <- ggplot(data = subset(election, st %nin% "DC"), mapping = aes(x =
r_points, y = reorder(state, r_points), color = party))

p1 <- p0 + geom_vline(xintercept = 0, color = "gray30") +

  geom_point(size = 2)

p2 <- p1 + scale_color_manual(values = party_colors)

p3 <- p2 + scale_x_continuous(breaks = c(-30, -20, -10, 0, 10, 20, 30,
40), labels = c("30\n (Clinton)", "20", "10", "0", "10", "20", "30",
"40\n(Trump)"))

p3 + facet_wrap(~ census, ncol=1, scales="free_y") +

  guides(color=FALSE) + labs(x = "Point Margin", y = "") +

  theme(axis.text=element_text(size=8))
```

(Healy, 2018)

Activity - Result




```
party_colors <- c("#2E74C0", "#CB454A")

p0 <- ggplot(data = subset(election, st %nin% "DC"), mapping
= aes(x = r_points, y = reorder(state, r_points), color =
party))

p1 <- p0 + geom_vline(xintercept = 0, color = "gray30") +

  geom_point(size = 2)

p2 <- p1 + scale_color_manual(values = party_colors)

p3 <- p2 + scale_x_continuous(breaks = c(-30, -20, -10, 0,
10, 20, 30, 40), labels = c("30\n (Clinton)", "20", "10",
"0", "10", "20", "30", "40\n(Trump)"))

p3 + facet_wrap(~ census, ncol=1, scales="free_y") +

  guides(color=FALSE) + labs(x = "Point Margin", y = "") +

  theme(axis.text=element_text(size=8))
```

(Healy, 2018)

Thematic map of U.S. election data

- Now we want to create the map on which we will plot our data
- As with any other ggplot, we need a dataframe with our information inside - in this case, that information will give R instructions for making a map
- The `maps` package contains predrawn map data

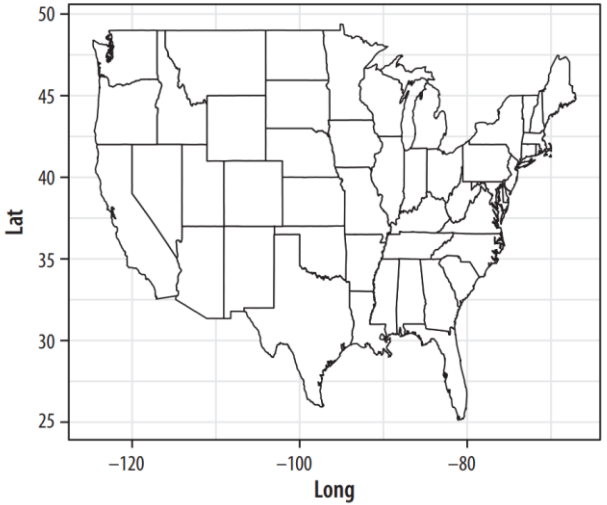
```
install.packages("maps")
library(maps)
us_states <- map_data("state")
head(us_states)
```

(Healy, 2018)

Thematic map of U.S. election data

- Our `us_states` dataframe gives R the instructions for drawing lines in the shape of a map of the United States using `geom_polygon()`

```
p <- ggplot(data =
us_states, mapping = aes(x =
long, y = lat, group =
group))

p + geom_polygon(fill =
"white", color = "black")
```

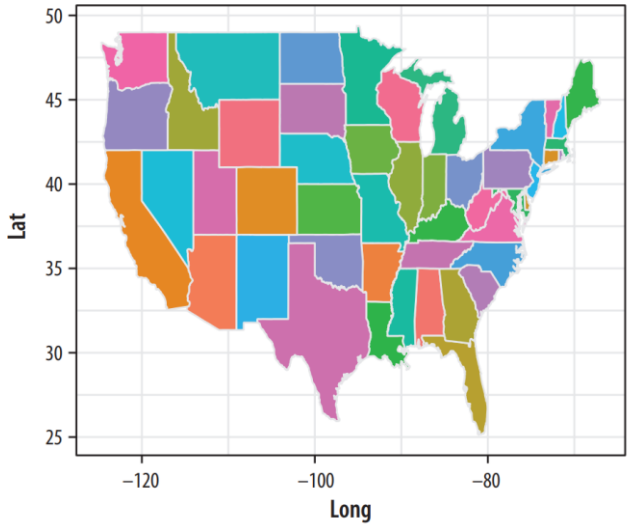


(Healy, 2018)

Thematic map of U.S. election data

- By mapping the fill aesthetic to region and changing the colour of our lines, we can adjust the appearance of our blank map

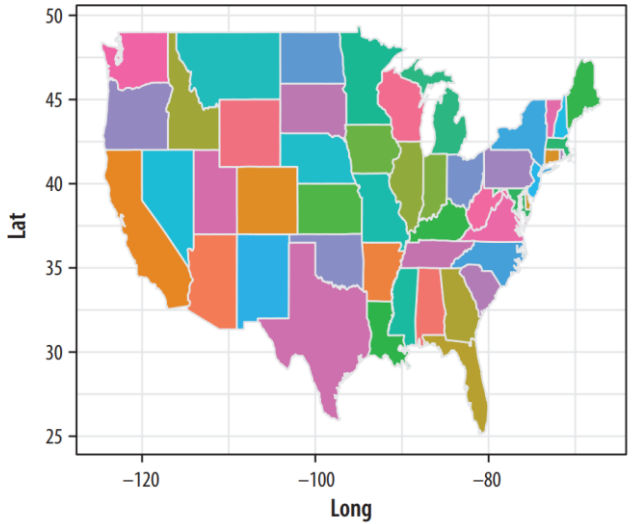
```
p <- ggplot(data = us_states,
aes(x = long, y = lat, group =
group, fill = region))

p + geom_polygon(color =
"gray90", linewidth = 0.1) +
guides(fill = FALSE)
```



(Healy, 2018)

Changing our map projection

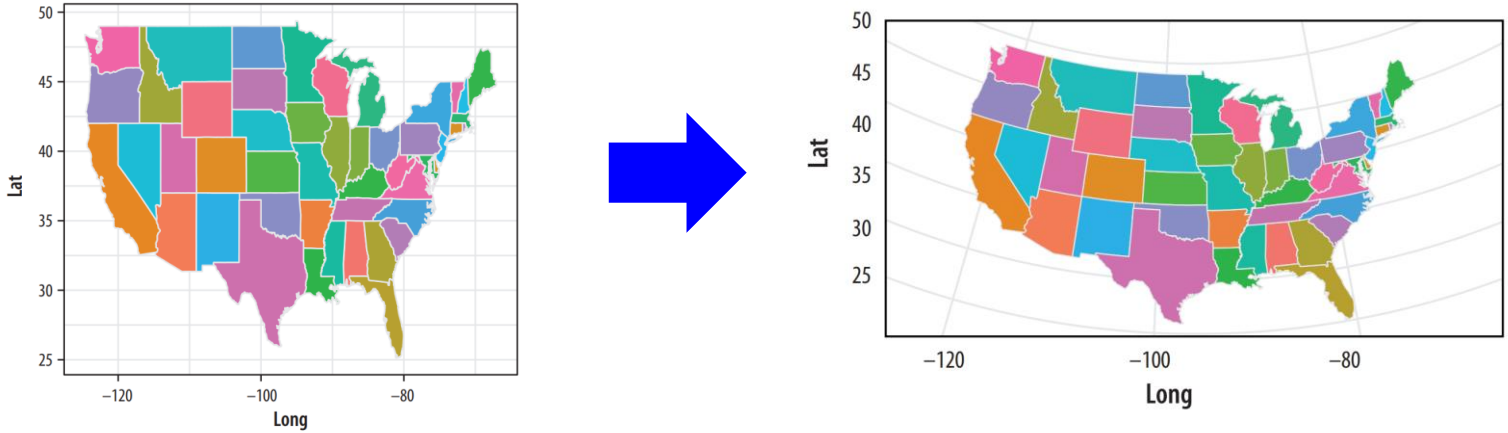
- By default, our map is plotted using the **Mercator projection**
- If we want our visualization to look more like typical maps of the United States, we can add two latitude parameters to transform our data to an **Albers projection**



(Healy, 2018)

Changing our map projection

```
p <- ggplot(data = us_states, mapping = aes(x = long, y = lat, group
= group, fill = region))

p + geom_polygon(color = "gray90", size = 0.1) +
coord_map(projection = "albers", lat0 = 39, lat1 = 45) + guides(fill
= FALSE)
```



(Healy, 2018)

Adding data to our map

- The next step is to add our election data to our map visual
- To do this, we need to merge our map dataframe and our election dataframe
- **It is very important to ensure that the variable we use to merge our datasets matches (that is, same case and no missing values), otherwise the polygons of our map visual will not join properly**

```
election$region <- tolower(election$state)

us_states_elec <- left_join(us_states, election, by = 'region')
```

(Healy, 2018)

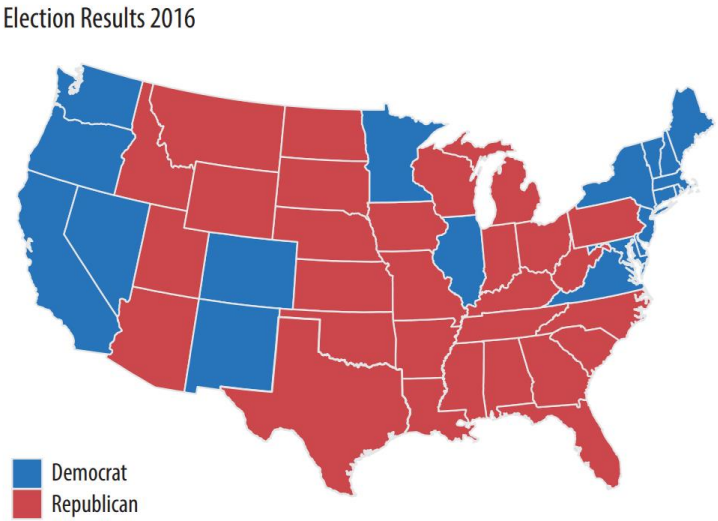
Adding data to our map

- Now that all of our data are in a single dataframe (`us_states_elec`), we can plot statewise election data on a map

```
p0 <- ggplot(data = us_states_elec, mapping = aes(x = long, y = lat,
group = group, fill = party))

p1 <- p0 + geom_polygon(color = "gray90", size = 0.1) +

  coord_map(projection = "albers", lat0 = 39, lat1 = 45)

p2 <- p1 + scale_fill_manual(values = party_colors) +

  labs(title = "Election Results 2016", fill = NULL)

p2 + theme_map()
```

(Healy, 2018)

Adding data to our map - Result




```
p0 <- ggplot(data = us_states_elec, mapping = aes(x = long, y = lat, group = group, fill = party))

p1 <- p0 + geom_polygon(color = "gray90", size = 0.1) + coord_map(projection = "albers", lat0 = 39, lat1 = 45)

p2 <- p1 + scale_fill_manual(values = party_colors) + labs(title = "Election Results 2016", fill = NULL)

p2 + theme_map()
```

(Healy, 2018)

Questioning our choropleths

- Our choropleth map shows voting results spatially (by state), but is it the best way to communicate our data?
- Could other factors be influencing the trends we see on our map?

(Healy, 2018)

Alternate choropleths

Population size and demographics

- State-level maps like the one that we have created are useful and easy to read, but they can insinuate a geographical explanation where none exists
- Voting data (like our example) are heavily influenced by population density

(Healy, 2018)

Plotting population density

- We will plot U.S. population density at the county level, using a premade county map dataframe and county information dataframe, both from the `socviz` library

```
county_map |>
 sample_n(5)

county_data |>
  select(id, name, state, pop_dens) |>
  sample_n(5)
```

(Healy, 2018)

Plotting population density

- Now we can merge our `county_map` and `county_data` dataframes on the shared `id` column

```
county_full <- left_join(county_map, county_data, by = "id")
```

- Our new dataframe contains (among other variables) longitude and latitude data for our map, population density in people per square mile, and county information

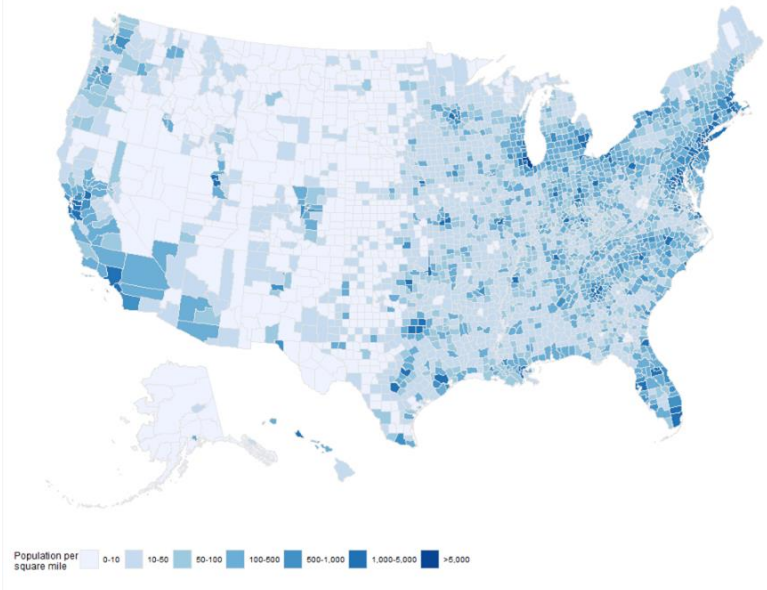
(Healy, 2018)

Plotting population density

```
p <- ggplot(data = county_full,
            mapping = aes(x = long, y = lat, fill = pop_dens, group =
group))
p1 <- p + geom_polygon(color = "gray90", size = 0.05) + coord_equal()
p2 <- p1 + scale_fill_brewer(palette="Blues",
                             labels = c("0-10", "10-50", "50-100", "100-
500", "500-1,000", "1,000-5,000", ">5,000"))
p2 + labs(fill = "Population per\nsquare mile") +
  theme_map() +
  guides(fill = guide_legend(nrow = 1)) +
  theme(legend.position = "bottom")
```

(Healy, 2018)

Plotting population density

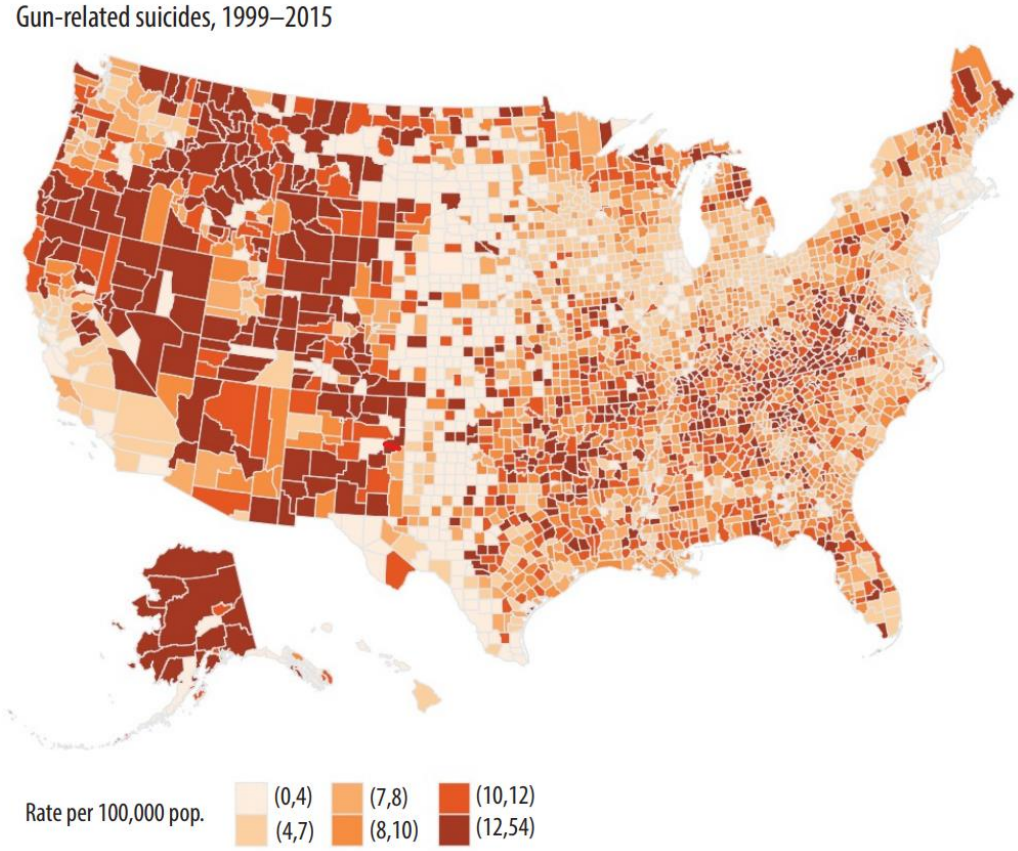



```
p <- ggplot(data = county_full, mapping = aes(x = long, y = lat, fill = pop_dens, group = group))

p1 <- p + geom_polygon(color = "gray90", size = 0.05) + coord_equal()

p2 <- p1 + scale_fill_brewer(palette="Blues",labels = c("0-10", "10-50", "50-100", "100-500", "500-1,000", "1,000-5,000", ">5,000"))

p2 + labs(fill = "Population per\nsquare mile") + theme_map() + guides(fill = guide_legend(nrow = 1)) + theme(legend.position = "bottom")
```

(Healy, 2018)

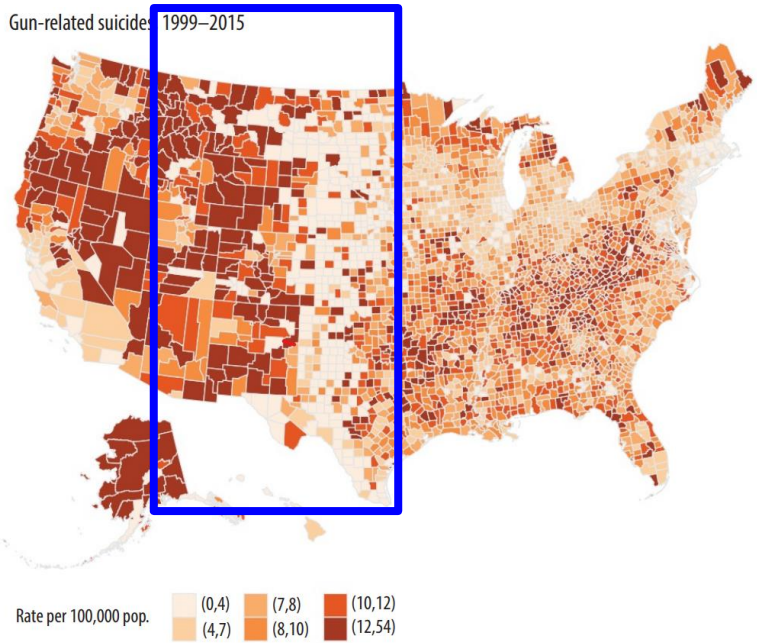
Why is it important to consider things like population density when we visualize data with thematic maps?

Map - Gun-related suicides in the United States

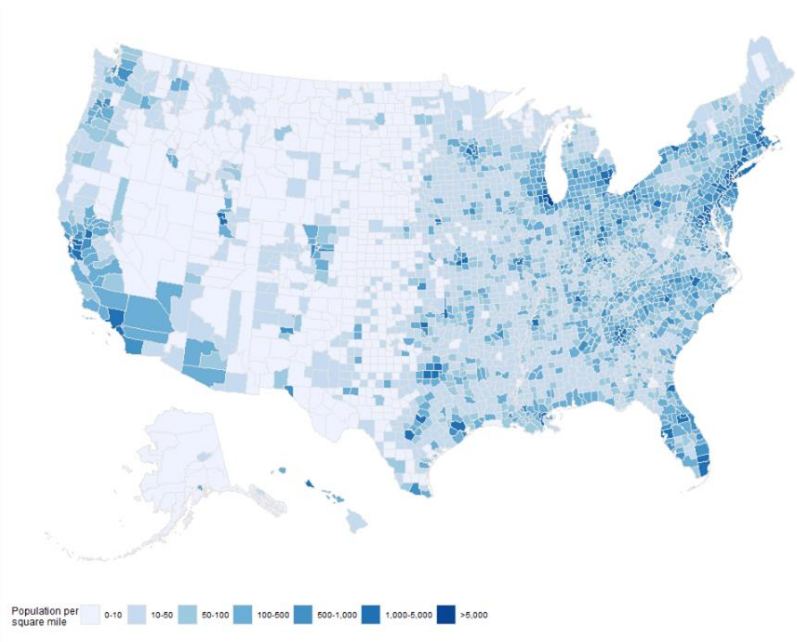


(Healy, 2018)

Comparing maps



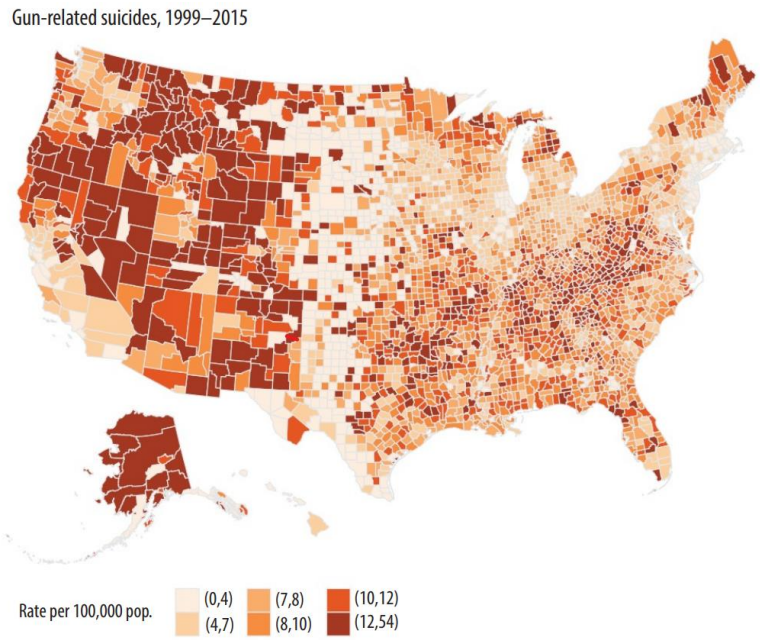
Gun-related suicides



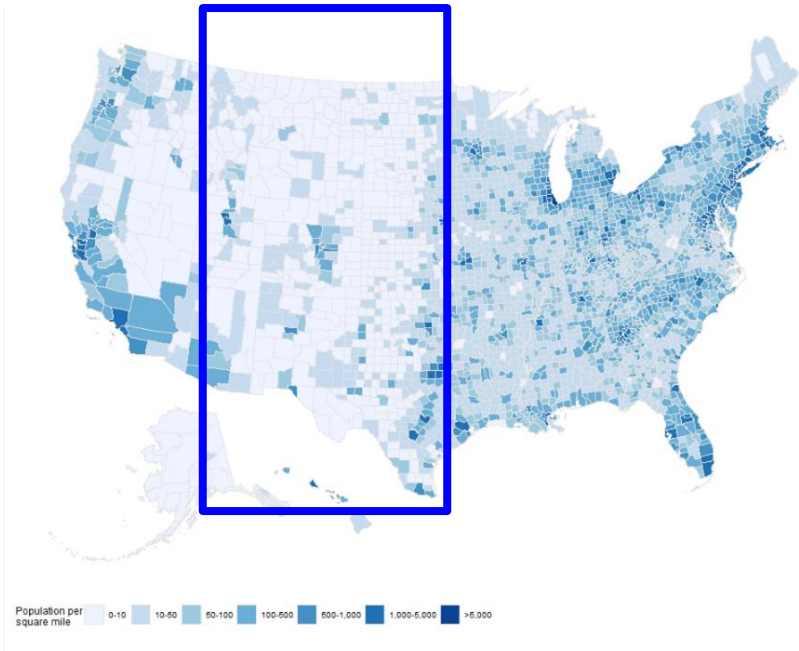
Population density

(Healy, 2018)

Comparing maps



Gun-related suicides



Population density

(Healy, 2018)

Maps need context

- If there are fewer than ten firearm-related suicides in a county per year, the CDC will report them as ‘suppressed’ so that individuals cannot be identified
- Imagine a situation with 10 such deaths in a county with only 100,000 inhabitants. We know that the number is between 0 and 10, so we could put it in our 0-10 (lowest suicide rate) bin on our map
- **But** 12 such deaths in the same county would put it in the highest suicide rate bin

(Healy, 2018)

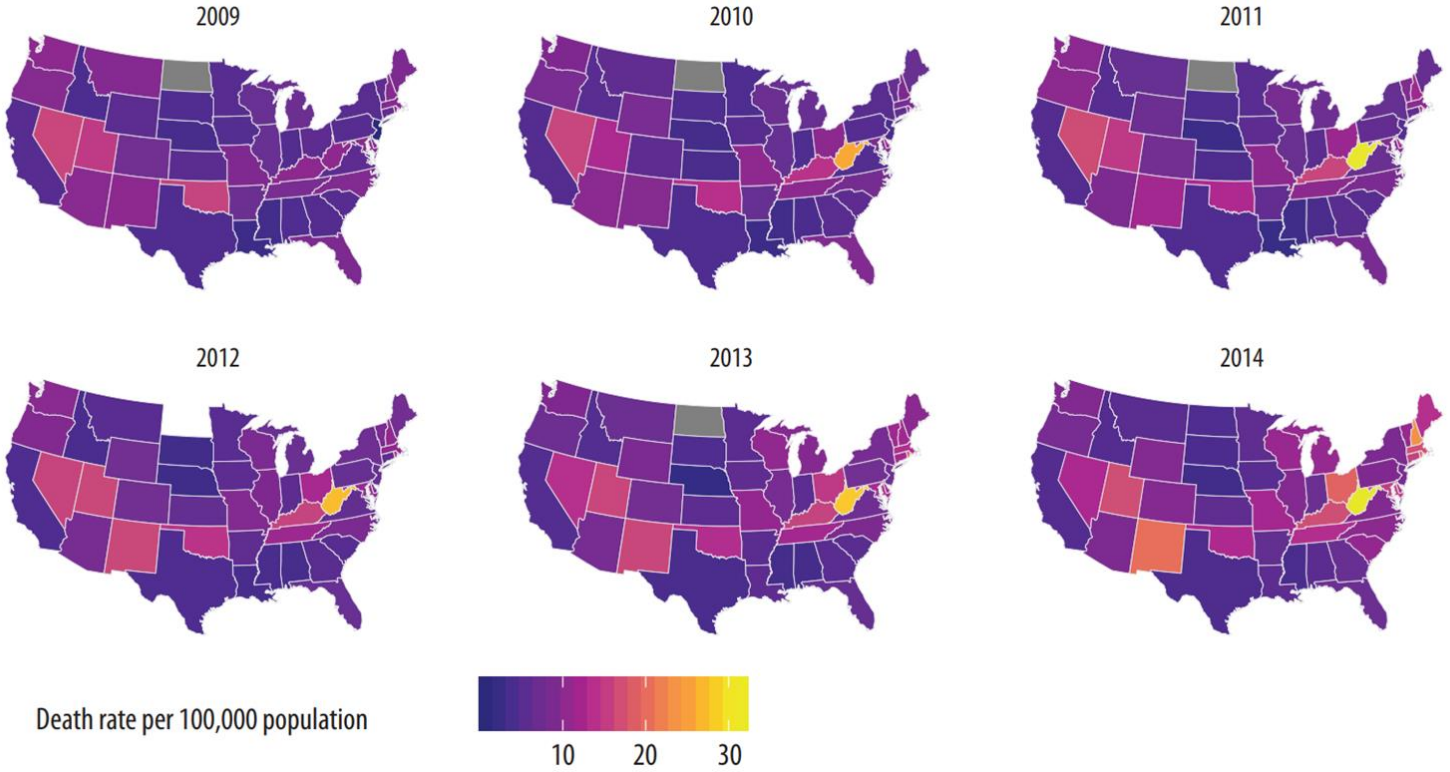
Maps need context

- **Differences in reporting and choices made while binning our data can produce misleading and mistaken spatial data visualizations**
- It is worth exploring underlying population and demographic data before relying on choropleths to find trends

(Healy, 2018)

Are our data really spatial?

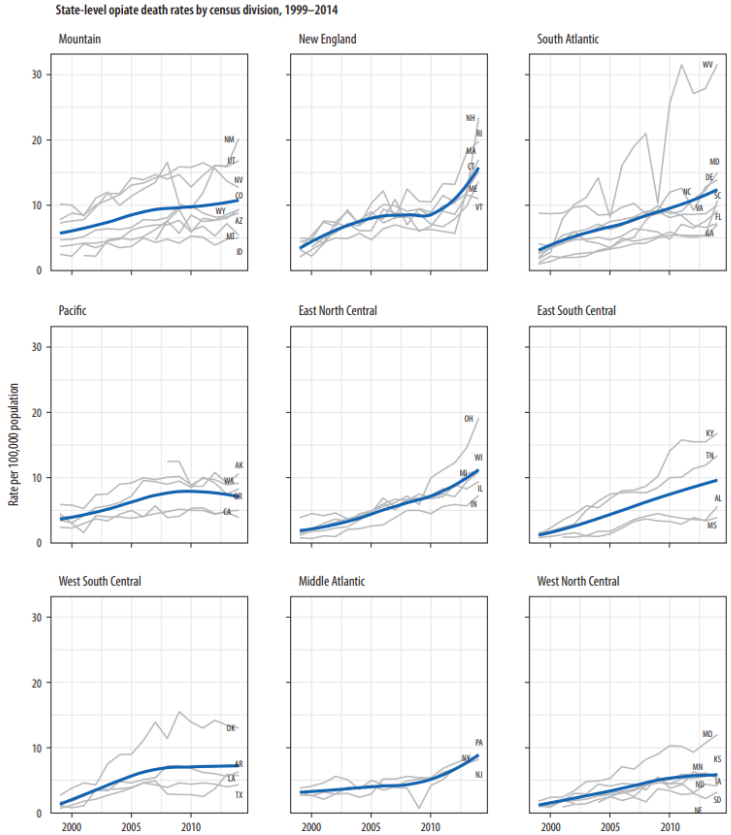
When are maps helpful?



(Healy, 2018)

When are maps helpful?

- The same data as the previous slide, visualized as a faceted time series by census region
- This version of the data lets us see change over time, and emphasizes differing trajectories of states within regions



(Healy, 2018)

Spatial for convenience

- Often (especially with public data), data are collected at the level of provinces, states, or some geographic area, or rates for these groups
  - For convenience and practicality
  - To protect individual privacy
- As data scientists, we need to “take care not to commit a kind of fallacy of misplaced concreteness that mistakes the unit of observation for the thing of real substantive or theoretical interest”
- That is, **data being plotted on a map does not necessarily mean that location is a factor affecting those data**

(Healy, 2018)


You've got to feel the flow...

Next...

- Dynamic Data Visualization
- What is the difference between static and dynamic data visualization? When should we use each?
- What are some tools for making dynamic data visualizations?