

L05 Maps

Data Visualization (STAT 302)

Anna Wagman

Contents

Overview	1
Datasets	1
Exercises	1
Challenge(s)	7

Overview

The goal of this lab is to explore various ways of building maps with `ggplot2`.

Challenges are not mandatory for students to complete. We highly recommend students attempt them though. *We would expect graduate students to attempt the challenges.*

Datasets

We'll be using the `US_income.rda` dataset which is already in the `/data` subdirectory in our `data_vis_labs` project. You'll also be downloading your own data to build maps.

```
# Load package(s)
library(tidyverse)
library(skimr)
library(maps)

# Load dataset(s)
load(file = "data/US_income.rda")
```

Exercises

Exercise 1

Plot 1 Make a county map of a US state using `geom_polygon()`. Maybe use your home state (mine is below) or a favorite state. Please do not use the state in example provided in the book.

Optional: Consider adding major cities (or your home town).

Hints:

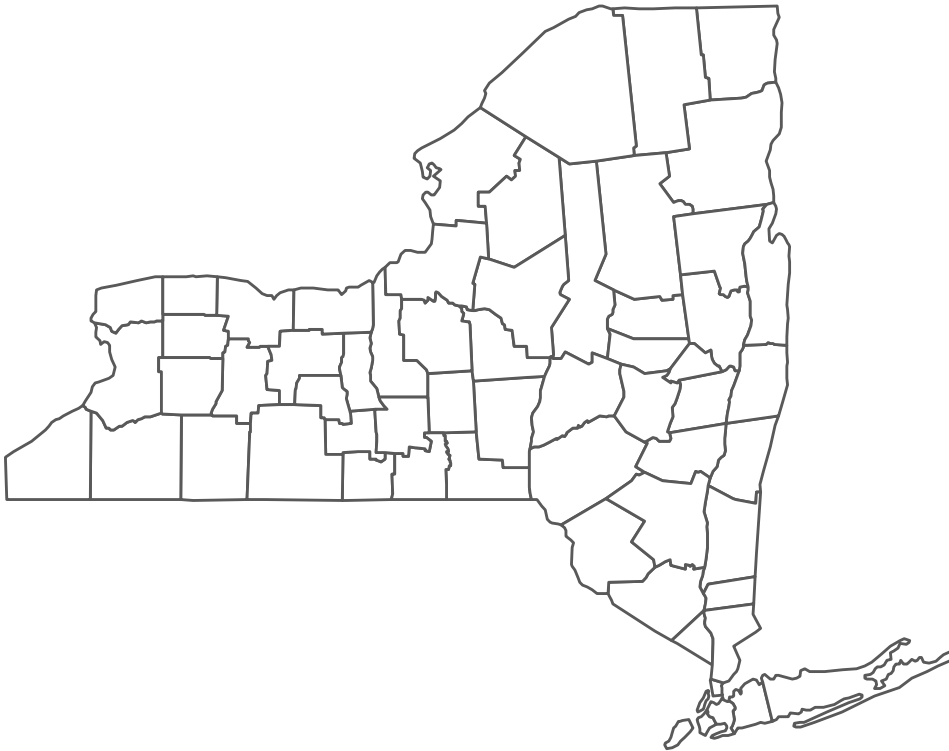
- See section 6.1 in our book.
- Void theme

```
#select my home state
ny <- map_data("county", "new york") %>%
  select(long, lat, group, id = subregion)

#plot state, divide by counties
```

```
ggplot(ny, aes(x = long, y = lat)) +
  #title
  ggtitle("New York") +
  #geompoly method
  geom_polygon(aes(group = group), fill = "white", color = "grey35") +
  #size
  coord_quickmap() +
  #theme
  theme_void()
```

New York



Plot 2 Now use `geom_sf()` instead. You'll need to download data for this. You can use either the `tigris` (github page) or `raster` packages. Either `tigris`' `counties()` with `cb = TRUE` or `raster`'s `getData()` could be useful.

```
library(raster)

## Loading required package: sp
##
## Attaching package: 'raster'
## The following object is masked from 'package:skimr':
##
##   bind
## The following object is masked from 'package:dplyr':
##
##   select
## The following object is masked from 'package:tidyr':
```

```
##
##   extract
library(tigris)

## To enable
## caching of data, set `options(tigris_use_cache = TRUE)` in your R script or .Rprofile.
library(sf)

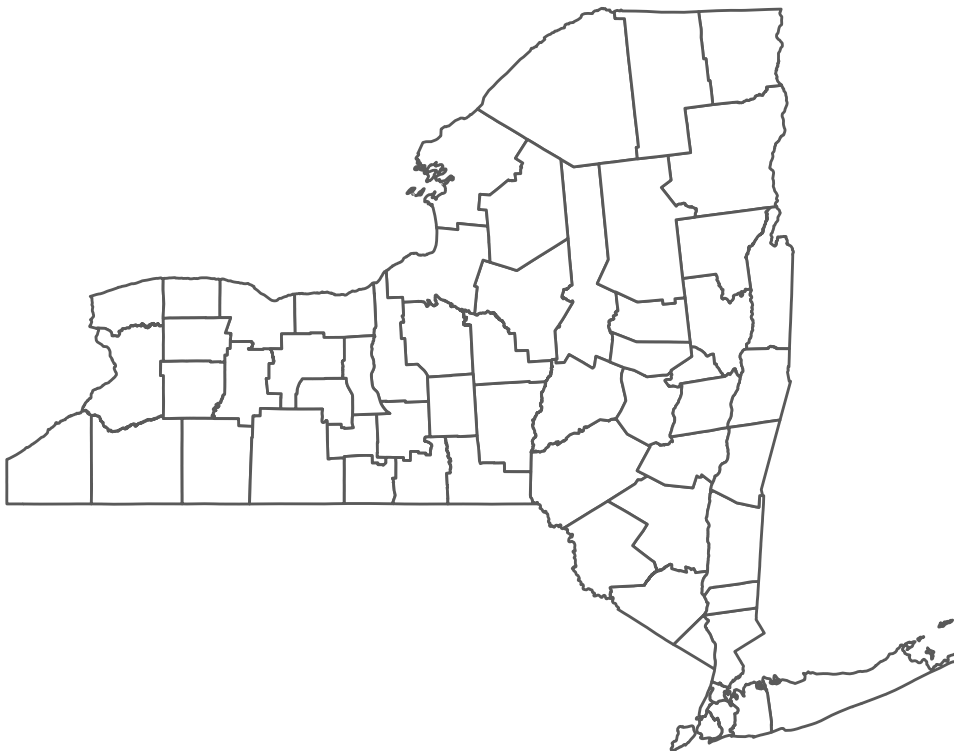
## Linking to GEOS 3.8.1, GDAL 3.2.1, PROJ 7.2.1
library(ggplot2)
library(dplyr)

##create same plot using geom_sf

##use tigris (cb = TRUE) and class = sf
#isolate homestate new york
nyny <- counties(state = "NY", cb = TRUE, class = "sf")

## |
##plot new york using geom_sf
ggplot(data = nyny, aes(geometry = geometry)) +
  geom_sf(fill = "white") +
  coord_sf() +
  theme_void() +
  ggtitle("New York")
```

New York



Exercise 2

Using US_income dataset, recreate the following graphics as precisely as possible.

```
# Setting income levels
US_income <- mutate(
  US_income,
  income_bins = cut(
    ifelse(is.na(median_income), 25000, median_income),
    breaks = c(0, 40000, 50000, 60000, 70000, 80000),
    labels = c("< $40k", "$40k to $50k",
               "$50k to $60k", "$60k to $70k", "> $70k"),
    right = FALSE
  )
)
```

Plot 1 *Hints:*

- geom_sf() — boundary color is "grey80" and size is 0.2
- viridis package (discrete = TRUE in scale_* function)
- Void theme

```
library(viridis)

## Loading required package: viridisLite

##
## Attaching package: 'viridis'

## The following object is masked from 'package:maps':
##
##      unemp

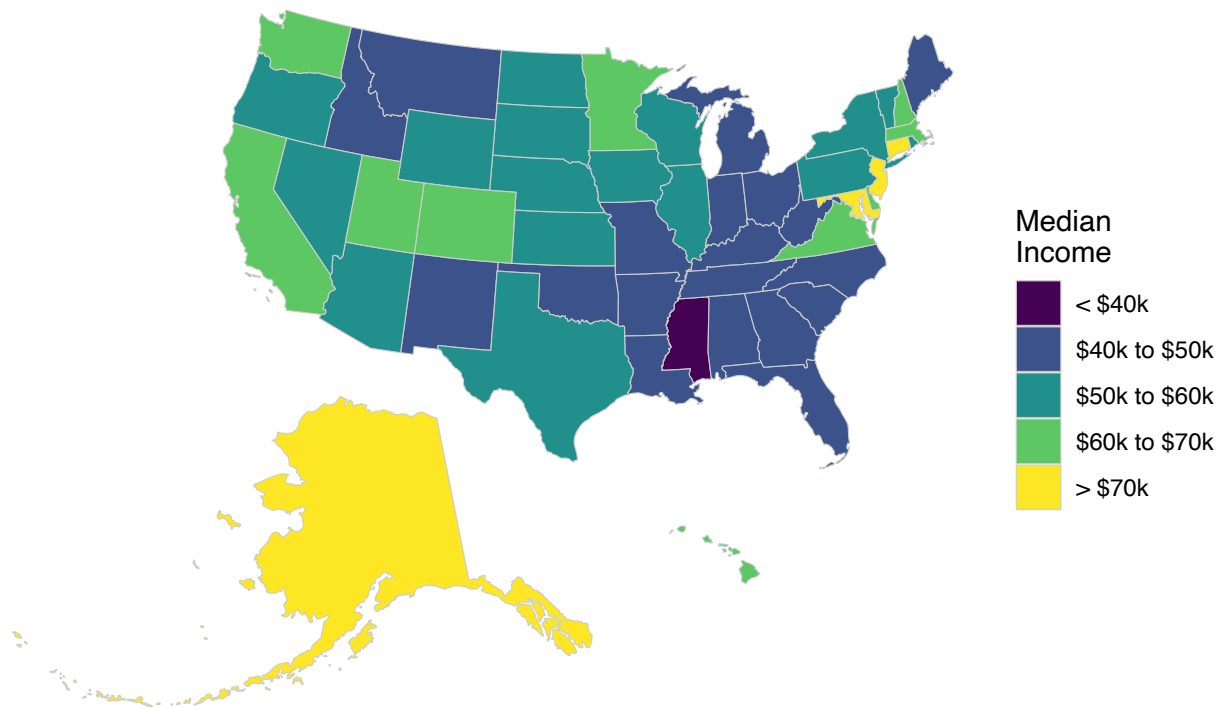
ggplot(data = US_income) +
  #use geom_sf
  geom_sf(aes(geometry = geometry, fill = income_bins)
    , color = "grey80"
    , size = 0.2
    , size = 0.2) +

  #use viridis package
  viridis::scale_fill_viridis(discrete = TRUE, name = "Median\nIncome") +
  #size the map
  coord_sf() +
  #set the theme
  theme_void()

## old-style crs object detected; please recreate object with a recent sf::st_crs()

## old-style crs object detected; please recreate object with a recent sf::st_crs()

## old-style crs object detected; please recreate object with a recent sf::st_crs()
```



Plot 2 *Hints:*

- `statebins::geom_statebins()`
- `viridis` package (`discrete = TRUE` in `scale_*` function)
- `Statebins` theme

```
library(statebins)

ggplot(US_income, aes(state = name, fill = income_bins)) +
  # use geom_statebins
  geom_statebins() +
  #use viridis color scheme
  viridis::scale_fill_viridis(discrete = TRUE, name = "Median\nIncome") +
  theme_statebins()
```



Median Income

 < \$40k	 \$40k to \$50k	 \$50k to \$60k	 \$60k to \$70k	 > \$70k
---	--	--	--	---

Exercise 3

Pick any city or foreign country to build a map for. You can dress it up or make it as basic as you want. Also welcome to try building a graphic like that depicted at the end of section 6.5 — use a different region though.

```
library(ggplot2)
library(dplyr)
require(maps)
require(viridis)

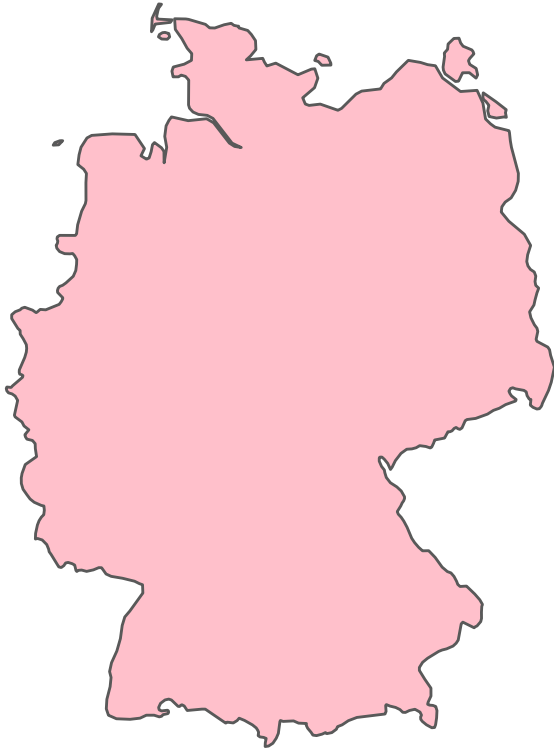
world_map <- map_data("world")

# select a foreign country
Germany_map <- c("Germany")

# Retrieve data about
Germany_map <- map_data("world", region = Germany_map)

#plot state, divide by counties
ggplot(Germany_map, aes(x = long, y = lat)) +
  #title
  ggtitle("Germany") +
  #geompoly method
  geom_polygon(aes(group = group), fill = "pink", color = "grey35") +
  #size
  coord_quickmap() +
  #theme
  theme_void()
```

Germany



Challenge(s)

Using the `tidycensus` package and few others, try to create a map like below using these directions. Try using a different geographical area and a different variable from the ACS.