UC Choudhary
Final Project
Tuesday, November 29, 2022
Instructor: Dr Ryan Hassler

STAT 484 FINAL PROJECT

The purpose of this project is to explore and examine important R packages like dplyr, ggplot and usmap which help in making comprehensive maps of the world. We want to explore these packages because they are unique to R and provide a versatility and smooth integration which is not possible in any other language. The main function which I will be using in this project is usmap because it is one of the rarer and less studied packages in R. Since I use the online version of RStudio, I had to install packages from the website CRAN projects. I will use the package usmap which is specifically built for making maps of the US. Initially I thought that I would have to use the sf function but after I deeper dive, I discovered usmap. The package ggplot2 is a necessary addition to any kind of advanced plotting or mapping in R.

Let us talk about usmap first. The description of this package can be found easily using ?usmap. This is what I found

Description

It is usually difficult or inconvenient to create US maps that include both Alaska and Hawaii in a convenient spot. All map data frames produced by this package use the Albers Equal Area projection.

Map data frames

Alaska and Hawaii have been manually moved to a new location so that their new coordinates place them to the bottom-left corner of the map. These maps can be accessed by using the us_map function.

Reference: Lorenzo, Paolo Di. "Mapping the US." 2. Mapping the US, The Comprehensive R Archive Network, 12 Nov. 2022, https://cran.r-project.org/web/packages/usmap/vignettes/mapping.html.

UC Choudhary
Final Project
Tuesday, November 29, 2022
Instructor: Dr Ryan Hassler

The function provides the ability to retrieve maps with either state borders or county borders using the regions parameter for convenience.

States (or counties) can be included and excluded using the provided include and exclude parameters. These parameters can be used together with any combination of names, abbreviations, or FIPS code to create more complex maps.

FIPS lookup tools

Several functions have been included to lookup the US state or county pertaining to a FIPS code.

Likewise a reverse lookup can be done where a FIPS code can be used to retrieve the associated state(s) or county(ies). This can be useful when preparing data to be merged with the map data frame.

Plot US map data

A convenience function plot_usmap has been included which takes similar parameters to us_map and returns a ggplot2::ggplot2 object. Since the output is a ggplot object, other layers can be added such as scales, themes, and labels. Including data in the function call will color the map according to the values in the data, creating a choropleth.

As the name suggests usmap helps in mapping the US , by county, state etc.

I know that most of the data is from seven years ago but I am not aware of more recent free datasets.

Reference: Lorenzo, Paolo Di. "Mapping the US." 2. Mapping the US, The Comprehensive R Archive Network, 12 Nov. 2022, https://cran.r-project.org/web/packages/usmap/vignettes/mapping.html.

UC Choudhary
Final Project
Tuesday, November 29, 2022
Instructor: Dr Ryan Hassler

This package is specifically for mapping the US and I think it produces better maps than a few Geography textbooks. Most people have trouble understanding numbers and need pictures to understand their meaning. This package is highly useful for such people. For example when we want to study the income, the illiteracy and other demographics by state or by county.

This package produces a pictographic output with the help of ggplot2.

I did not obtain any data from the US Census Bureau or any other government agency. This data is part of the usmap package and since we are exploring the various tools in this package, I thought that since this package has this information by default, I do not need to gather any more information from other sources. For example: plot_usmap(data = statepop, values = "pop_2015", color = "red") displays the map of the United States with population data from 2015. All I had to do to access this data was import usmap package and run this command.

This is one of the massive benefits of the usmap package, because the programmer/data scientist does not need to spend hours looking for government information and all the required information is readily available at one's fingertips.

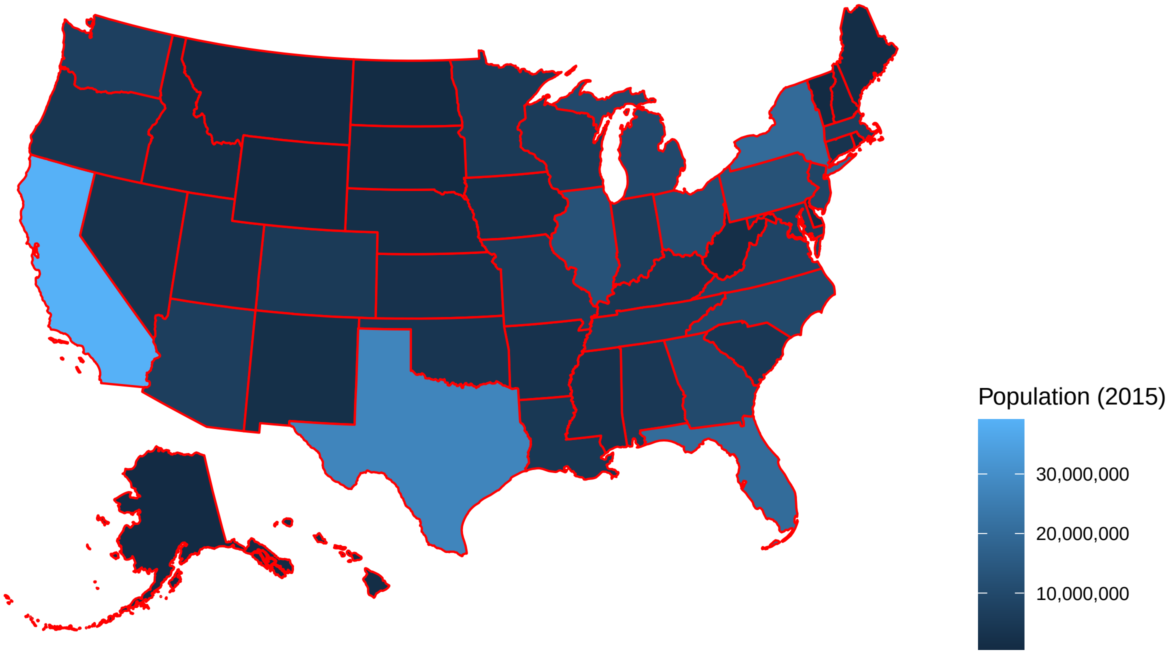
Reference: Lorenzo, Paolo Di. "Mapping the US." 2. Mapping the US, The Comprehensive R Archive Network, 12 Nov. 2022, https://cran.r-project.org/web/packages/usmap/vignettes/mapping.html.

Final-Project.R

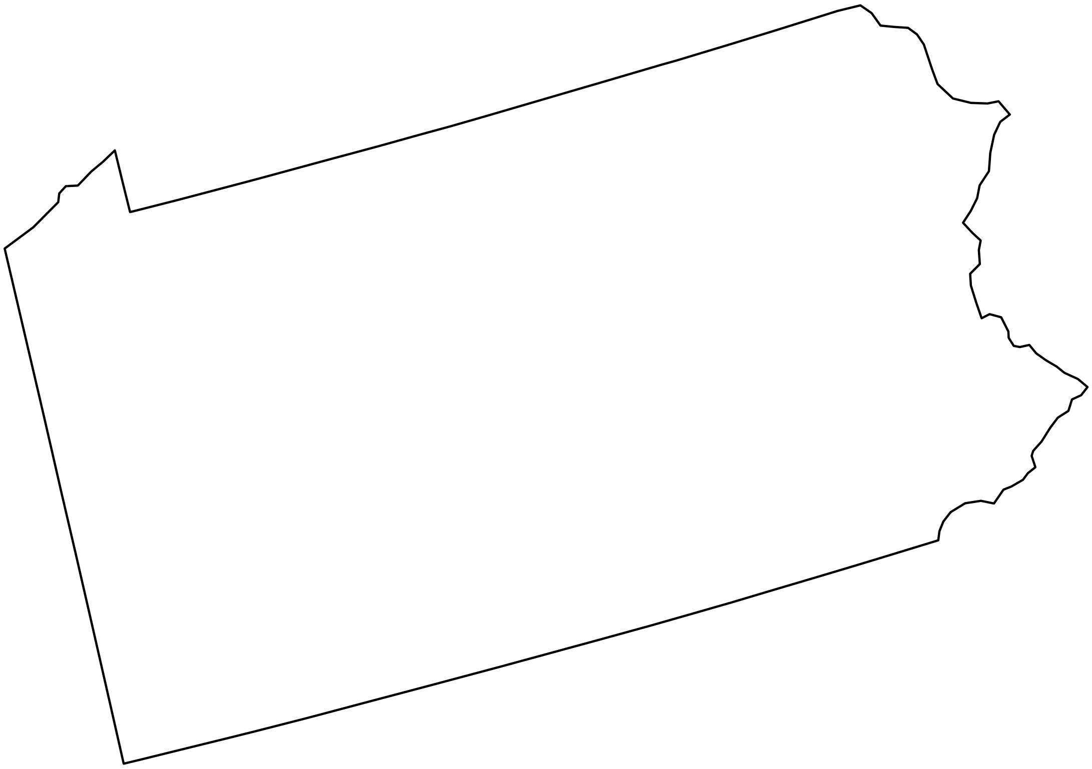
r1586649

2022-11-30

```
library(usmap)
library(ggplot2)
library(stringr)
#This is an example of how powerful usmap package is.
#It has rendered even pwerful functions like sf obsolete.
#Most of this rapid development in R is thanks to the community who want to
#make R a mainstream language for data scientists.
#The lab function is a part of ggplot2 which helps in creating legend
# naming axes and creating headings
#Let us try to make a map of the US States based on population(2015)
plot_usmap(data = statepop, values = "pop_2015", color = "red") +scale_fill_continuous(name = "Populati
  theme(legend.position = "right")
```



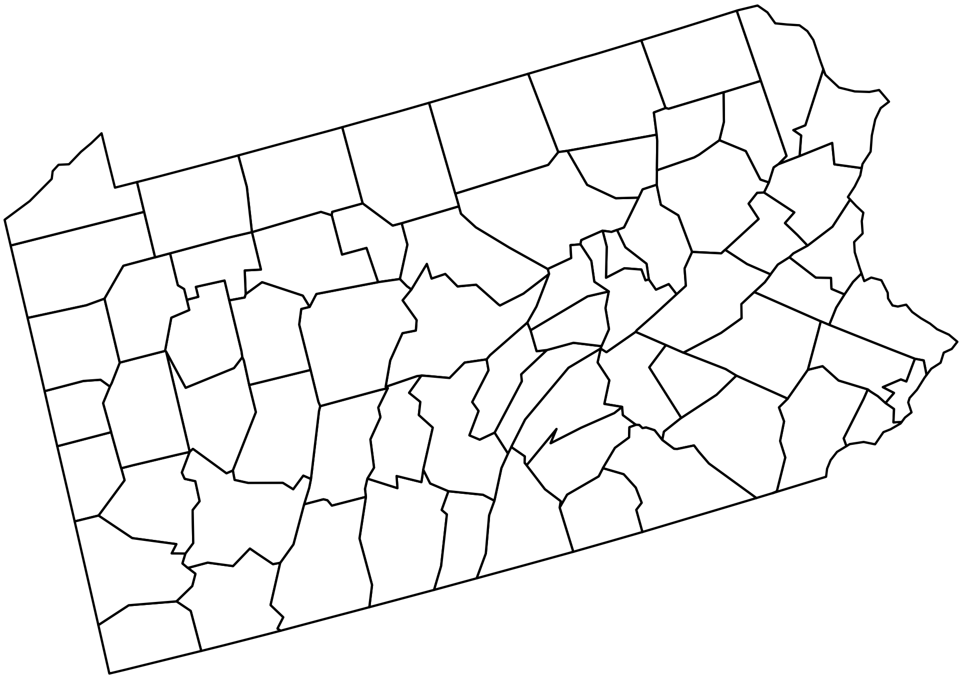
```
#The above is a a code snippet which I found online
# A basic map of PA can be made using the code below
plot_usmap(include = c("PA"))
```



```r
# I will make two maps of PA in this demo
# I will map PA by county
# I will map the East Coast states by population
plot_usmap("counties", include = c("PA")) +
  labs(title = "PA Counties", subtitle = "This
       is a blank map of the counties of PA.")
```

PA Counties

This
    is a blank map of the counties of PA.

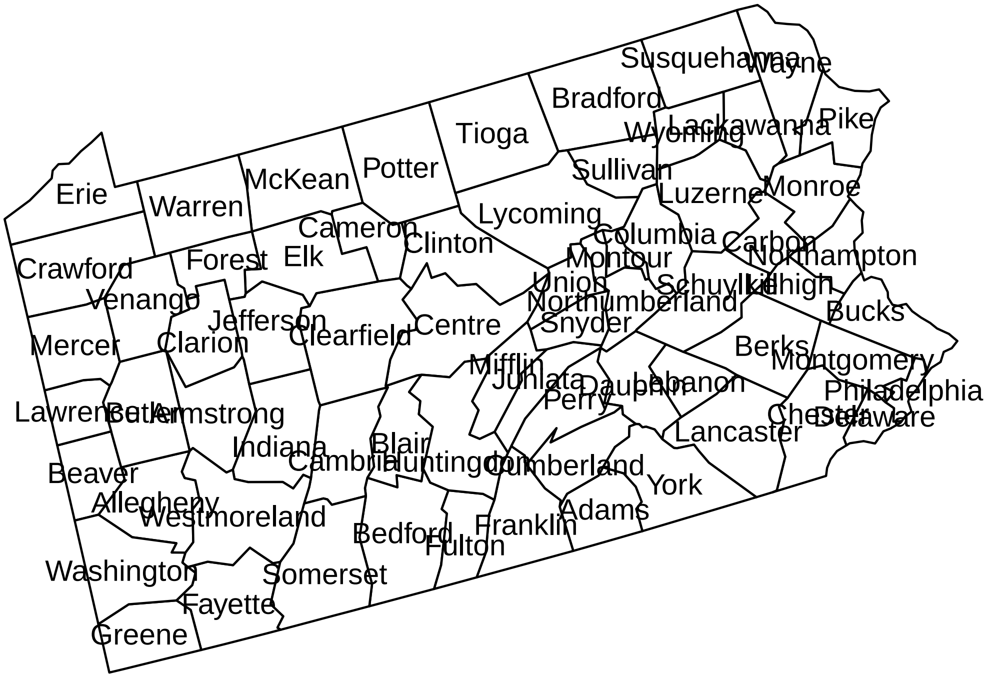


```
plot_usmap("counties", include = c("PA"),labels = TRUE) +
  labs(title = "PA Counties", subtitle = "This
      is a blank map of the counties of PA.")
```

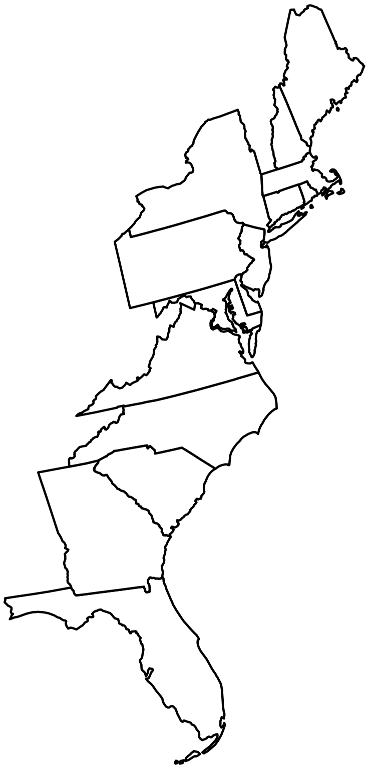
PA Counties

This

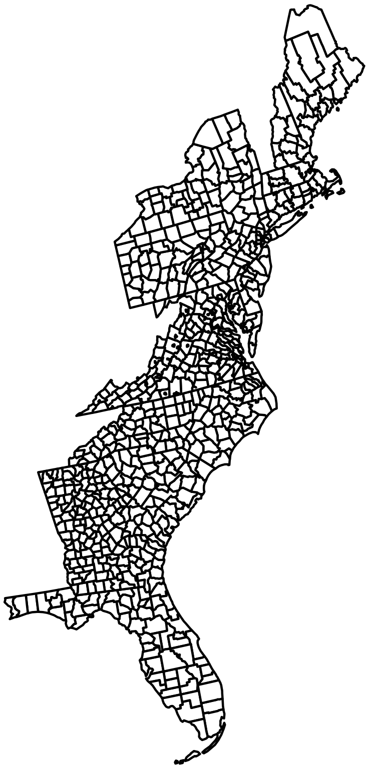
is a blank map of the counties of PA.



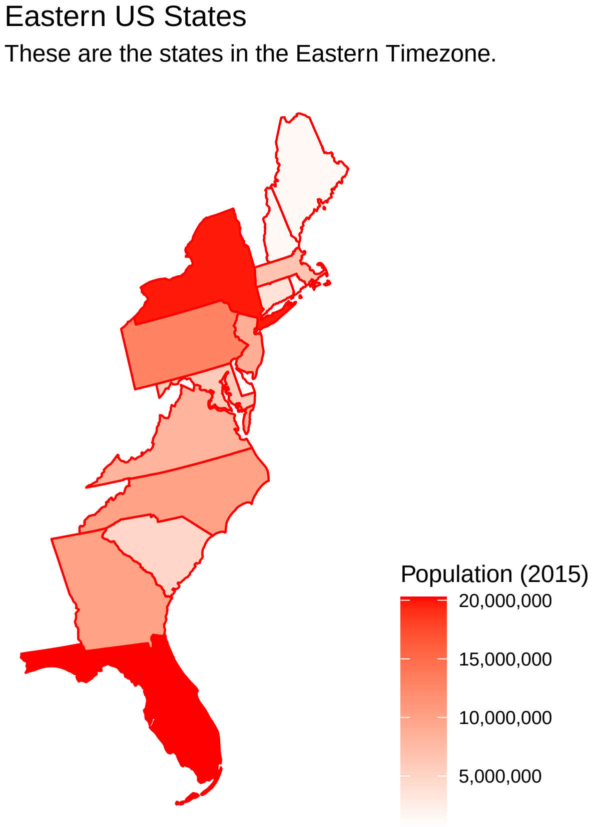
```r
#The above are the counties of Pennsylvania
#Let us make a blank map of the East Coast States
plot_usmap(include = c("PA", "ME", "NH", "MA", "RI", "CT", "NY","NJ","DE",
                       "MD","VA", "NC","SC","GA","DC","FL"))
```



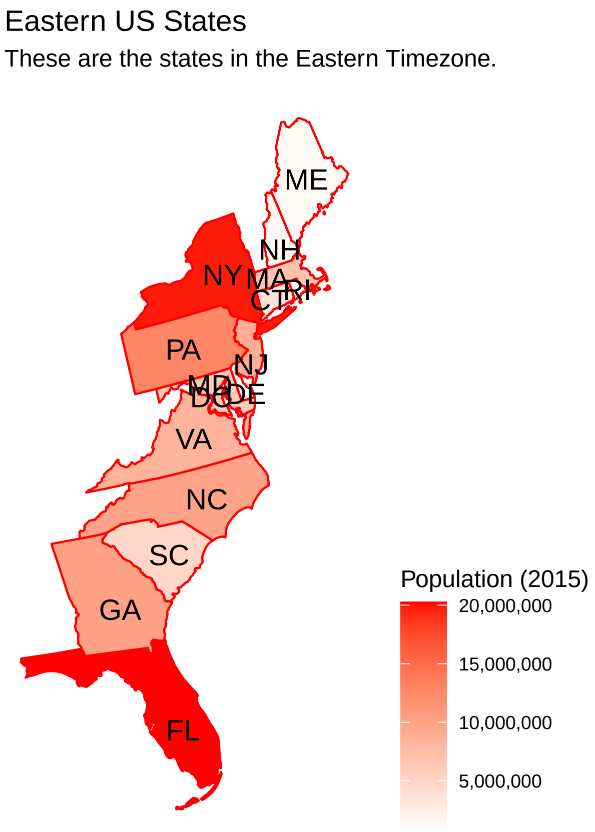
```
#I believe these are the East Coast States
#Let us map them by county
plot_usmap("counties", include = c("PA", "ME", "NH", "MA", "RI", "CT", "NY","NJ","DE",
                                    "MD","VA", "NC","SC","GA","DC","FL"))
```



```
#Now let us map them by population
plot_usmap(data = statepop, values = "pop_2015", include = c("PA", "ME", "NH", "MA", "RI", "CT", "NY","
                                                             "MD","VA", "NC","SC","GA","DC","FL"), colo
  scale_fill_continuous(
    low = "white", high = "red", name = "Population (2015)", label = scales::comma
  ) +
  labs(title = "Eastern US States", subtitle = "These are the states in the Eastern Timezone.") +
  theme(legend.position = "right")
```



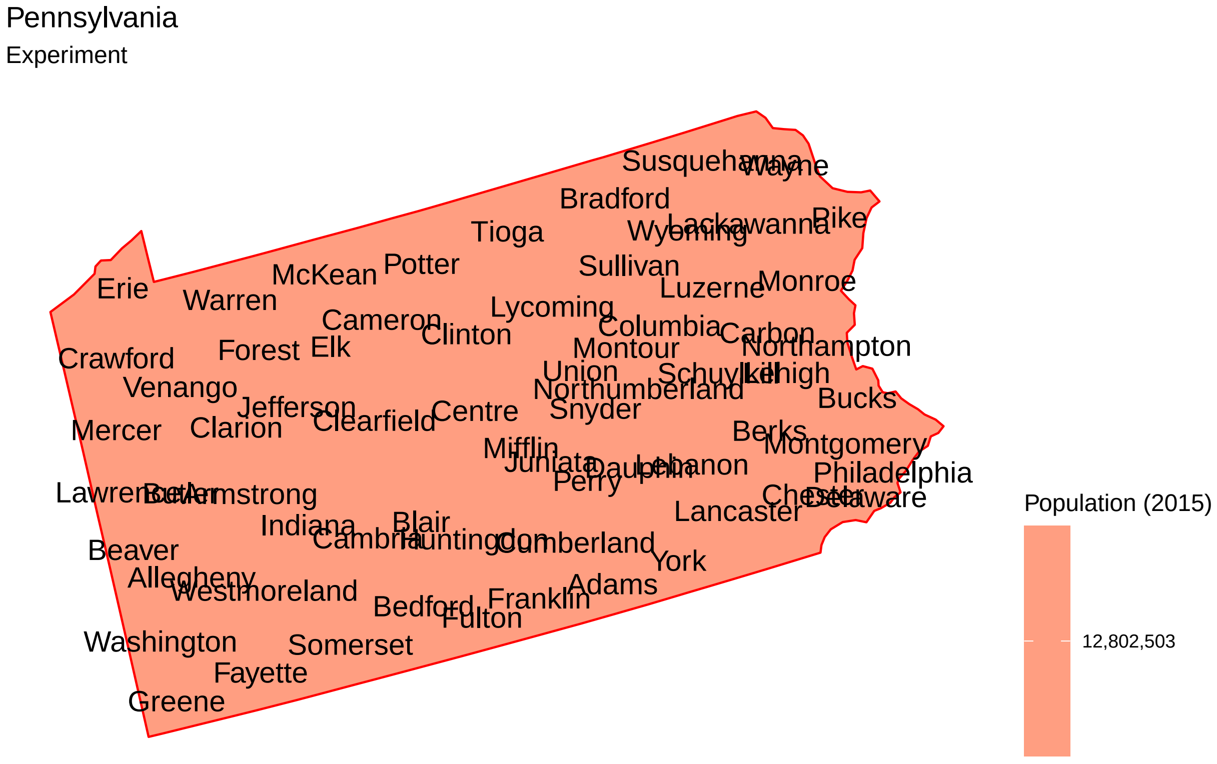
```
#Now with labels
plot_usmap(data = statepop, values = "pop_2015", include = c("PA", "ME", "NH", "MA", "RI", "CT", "NY","
                                                             "MD","VA", "NC","SC","GA","DC","FL"), colo
  scale_fill_continuous(
    low = "white", high = "red", name = "Population (2015)", label = scales::comma
  ) +
  labs(title = "Eastern US States", subtitle = "These are the states in the Eastern Timezone.") +
  theme(legend.position = "right")
```



Eastern US States

These are the states in the Eastern Timezone.

```
#I know that New England looks a bit cluttered but we can change the scale easily.
#Names of the counties and display of the total population a failed experiment
plot_usmap("counties", data = statepop, values = "pop_2015", include = c("PA"), color="red",labels = TRU
  scale_fill_continuous(low = "white", high = "red", name = "Population (2015)", label = scales::comma)
  labs(title = "Pennsylvania", subtitle = "Experiment") +
  theme(legend.position = "right")
```



```
# I don't know all the data sets that could make this possible
#This project would have been impossible cran.r-project.org: massive shout-out
```