Mapmaking Workshop Using R

Nov. 2023 – UB

Fuzhen Yin

1. Workshop objectives:

- Collect spatial data (.shp) from US Census Bureau
- Read spatial data (.shp) in RStudio, create maps and export
- Produce interactive maps
- Produce maps with multiple layers of information
- Produce thematic maps with socio-economic attributes

2. Data preparation

2.1. Workshop materials

Download the "Project_1" zip file from here:
https://drive.google.com/file/d/18UE3yWxXUtx2hcLMtrmMJE_l2ZuveDlW/view?usp=sharing

Remember the location where you saved the file

Right click "Project_1"to unzip

2.2. TIGER/Line spatial data from U.S. Census Bureau

Spatial (TIGER/Line) Data from the US Census Bureau (data format = shapefile, suffix = ".shp")

a) Go to website (https://www.census.gov/geographies/mapping-files/time-series/geo/tiger-line-file.2022.html#list-tab-790442341) or Google search "US census, Tiger/line Shapefiles"
b) Select 2022 and click "Web Interface"
c) Select year = "2022" and layer type = "Census tract", submit
d) Select state = "New York" and click download (! Remember file location)
e) Unzip
f) Move the unzipped folder to the "data" folder in "Project_1"

2.3. Social economic data from U.S. Census Bureau

Social economic data from US Census Bureau (data format = table, suffix = ".csv")

a) Go to website (https://data.census.gov/) or Google search "US census bureau data"
b) Click "Advanced Search"

c) Under the “Geography” category, select “Census Tract” --> “New York” --> “Erie County” --> “All Census Tracts within Erie County, New York”
d) In the search box, type “median income”
e) In the results, select “S1903 Median income in the past 12 month”
f) Download table --> select “2021” --> “Download.csv” (! Remember file location)
g) Unzip
h) Move the unzipped folder to the “data” folder in “Project_1”

2.4. Spatial data from NYS Clearinghouse

Spatial data from NYS clearing house

a) Go to website (https://data.gis.ny.gov/) or Google search “NYS Clearinghouse”
b) Search “school”
c) “Schools K-12” --> “Download” --> “Download Shapefile” --> “Save” (! Remember file location)
d) Unzip

Move the unzipped folder to the “data” folder in “Project_1”

3. Install R and R studio

R is the based app to process R programming language, and RStudio integrates with R to provide further functionality such as graphical user interface (GUI). We need to install both R and RStudio for this workshop.

Note: there are R and RStudio installed in the computer lab. Please use the PC there if you cannot install.

3.1. Install R

Website: https://cran.rstudio.com/

Windows users:

“Download and install R” → “Download R for Windows” → “install R for the first time” → “Download R-4.3.1 for Windows” → save → Go to the folder where you save the file “R-4.3.1-win.exe”, double click to run the installation, click “Next” until the end → “Finish”

Mac users:

“Download R for macOS” --> Under the “Latest release”, download the version that is suitable for your machine (“R-4.3.2-arm64.pkg” or “R-4.3.2-x86_64.pkg”) --> save --> Go to the folder where you save the file, run the installation, click “Next” until the end --> “Finish”

3.2. Step 2: Install R studio

Website: https://posit.co/download/rstudio-desktop/

Windows users:

"Download RStudio desktop for windows" → Save → Go to the folder where you download the file "RStudio-2023.09.1-494.exe" --> Run the installation, click "Next" until the end → "Finish"

Mac users:

Scroll down --> Find the row with "macOS 11+", and click "

RSTUDIO-2023.09.1-494.DMG" --> Save --> Go to the folder where you download the file --> Run the installation, click "Next" until the end --> "Finish"

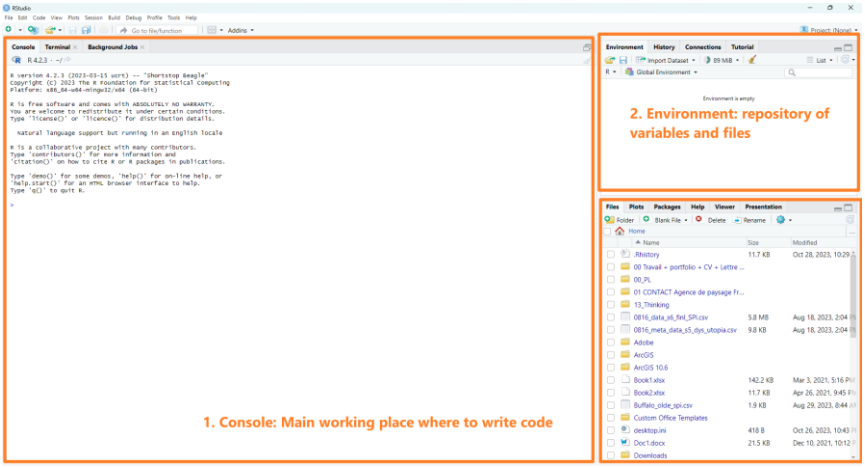
4. Brief introduction to RStudio

4.1. Run RStudio

Double click to run RStudio

(Note: if some issues occur during packages installation, use the following method to restart RStudio. Right click RStudio--> “Run as administrator” --> “Yes”)

After opening RStudio, you will see the following interface.

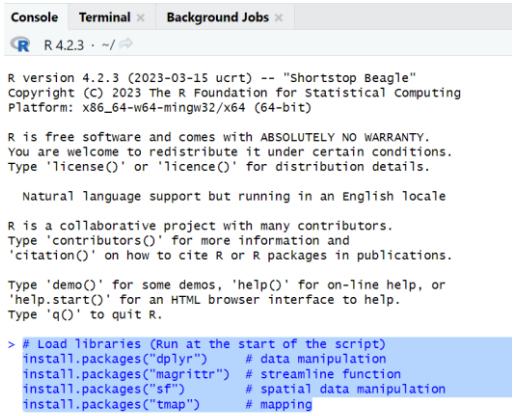


4.2. Step 2: Install packages

Packages are groups of functions that have been developed by researchers from a variety of fields (e.g., statistics, computer science, geography). There are tons of packages. For this workshop, we only need to install a few that are related to map making.

Copy and paste the command below into the console, then hit “Enter”. Wait a few seconds for the package to install on your machine.

```
# Load libraries (Run at the start of the script)
install.packages("dplyr")      # data manipulation
install.packages("magrittr")   # streamline function
install.packages("sf")         # spatial data manipulation
install.packages("tmap")       # mapping
```



paste --> select all rows --> hit "ENTER"

Hint: if you would like to know more about a particular package, google search "R package [PACKAGE NAME]", you can find tons of tutorials and explanations on that package. For example, by searching "R package tmap", can you see a search result "tmap: get started!"? It will surprise you!

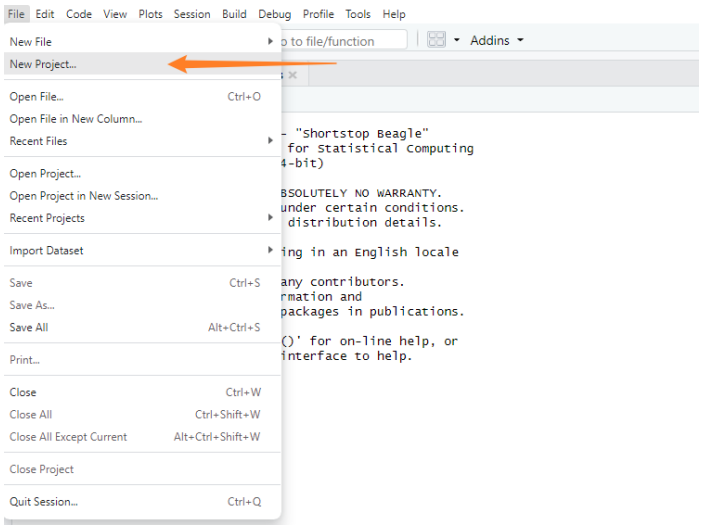
4.3. Basic coding concepts in R

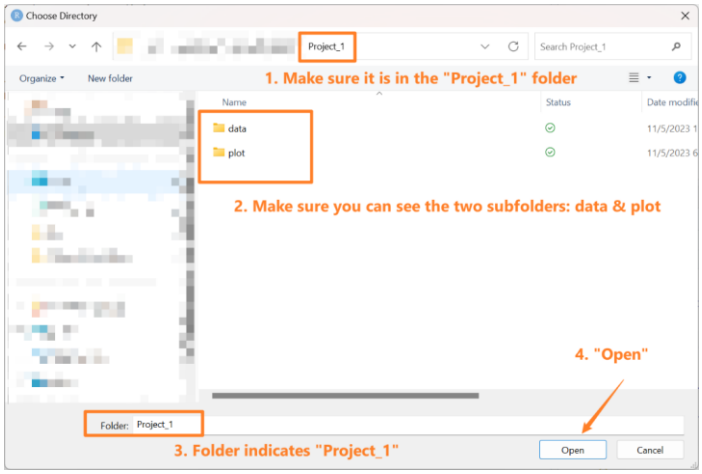
a) **Project**: enables your work to be bundled in a portable, self-contained folder. Put all the relevant scripts, datafiles, figures in subfolders.
b) **Script**: a text file containing a set of commands and comments.
c) **Command**: a specific action assigned to a program to perform a specific task (case sensitive!!).
d) **Comment**: starts with "#" and is added to make the code easier for humans to understand. It will not be read by the machine. Deleting comments will not affect the workflow of your functions. But it's always good practice to comment on your code.

5. Mapmaking

5.1. Create a new project

In RStudio menu → File → New Project → Existing Directory → Browse → Navigate to the "Project_1" folder → **Double clicks** to go inside of the folder "Project_1" → Open → Create Project

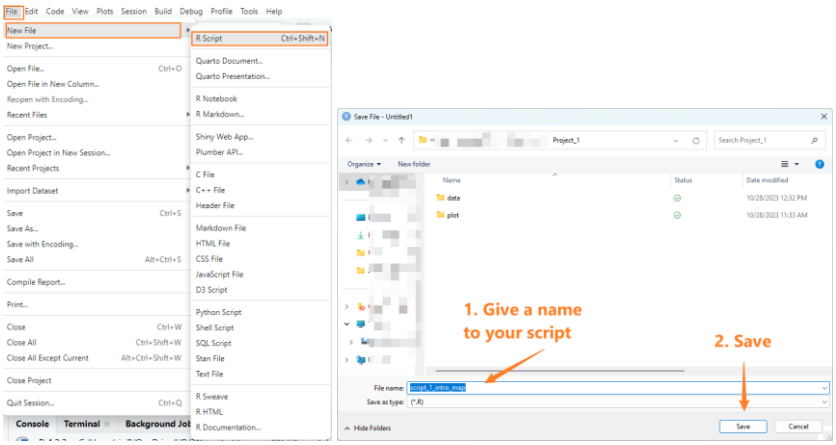




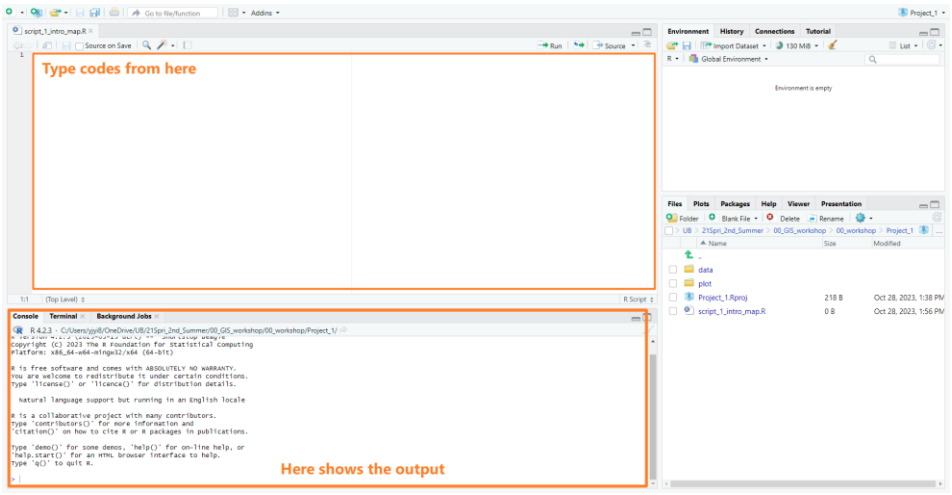
5.2. Create an R Script and save

File → New File → R Script

File → Save → Specify File Name → Save



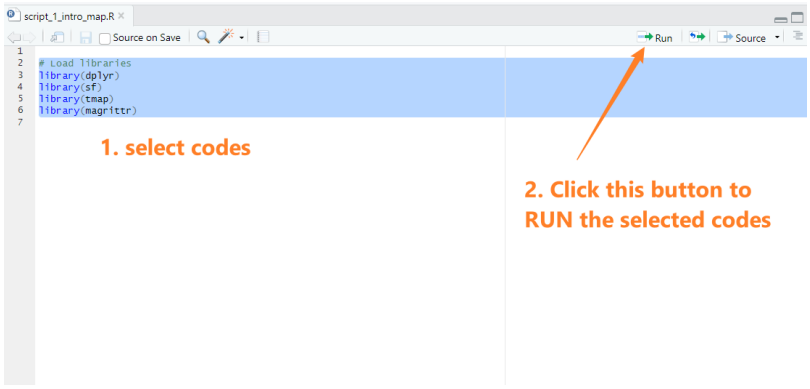
After creating the script, the interface should be like this:



5.3. Load in packages we have installed before.

Copy pastes the chunk of codes below → Select those lines → Run

```
# ====== Library =======
# Load libraries (Run at the start of the script)
library(dplyr)      # data manipulation
library(magrittr)   # streamline function
library(sf)         # spatial data manipulation
library(tmap)       # mappings
```



5.4. Step Making Maps

- Black represents commands.
- New added codes are in **bold**.
- Green represents comment lines.

1) Map of NYS places

- Read in the spatial data of NYS places (shapefile .shp) and create a simple map.
- Copy pastes the chunk of codes below → Select & Run
- Notes: function `st_read("PATH")` is used to read shapefiles (with the suffix ".shp")

```
# ====== 1. Map NYS places (cities + villages) =======
# read spatial data from a shapefile (with a suffix .shp)
nys_place <- st_read("data/tl_2022_36_place/tl_2022_36_place.shp")
# map
tmap_mode("plot")
tm_shape(nys_place) + tm_polygons()
```

Commented [fyin1]: nys_place is a name you give to the file.

Commented [fyin2]: The relative path where the shapefile is stored.

2) Map NYS places and export

```
# ====== 2. Map NYS places and save as .pdf =======
# read spatial data from a shapefile (.shp)
nys_place <- st_read("data/tl_2022_36_place/tl_2022_36_place.shp")

# export start. Specify file name
pdf("plot/map1_nys_places.pdf")
    # Set mode to "plot" to export
tmap_mode("plot")
tm_shape(nys_place) + tm_polygons()
dev.off()
```

Commented [fyin3]: Save the plot in the folder"plot" and name it as : "map1_nys_places.pdf".

Feel free to use a different name.

3) Create an interactive map of NYS places

```
# ====== 3. Interactive map NYS places (cities + villages) =======
# read spatial data from a shapefile (.shp)
nys_place <- st_read("data/tl_2022_36_place/tl_2022_36_place.shp")

# Interactive map, set the plot mode to "view"
tmap_mode("view")
tm_shape(nys_place) + tm_polygons()
```

Commented [fyin4]: Set the tmap_mode to "view" for an interactive map.

4) Re-design the map of NYS places

```
# ====== 4. Adjust the design of NYS places map =======
# read spatial data from a shapefile (.shp)
nys_place <- st_read("data/tl_2022_36_place/tl_2022_36_place.shp")

# Interactive map, set the plot mode to "view"
tmap_mode("view")
tm_shape(nys_place) + tm_polygons(alpha=0.5, col="orange", border.col = "black", lwd = 1)
```

Commented [fyin5]: "alpha" specifies the transparency (0=transparent).

"col" = map fill color. Try "blue"

"border.col" = border color

"lwd" = line width

5) Filter out a particular city from NYS places

```
# ====== 5. Extract a particular city/village =======
# read spatial data from a shapefile (.shp)
nys_place <- st_read("data/tl_2022_36_place/tl_2022_36_place.shp")
place_1 <- nys_place %>% filter(NAME == "Niagara Falls")    # Try "Buffalo" or "Tonawanda"

# Interactive map, set the plot mode to "view"
tmap_mode("view")
tm_shape(place_1) + tm_polygons(alpha=0.5, col="orange", border.col = "black", lwd = 2)
```

Commented [fyin6]: Based on the variable "NAME"'s value to filter out a particular city. Try to replace "Niagara Falls" with other value for example "Buffalo".

6) Overlapping the map of NYS places with the map of a particular city

```
# ====== 6. Extract a particular city/village and export map =======
# read spatial data from a shapefile (.shp)
nys_place <- st_read("data/tl_2022_36_place/tl_2022_36_place.shp")
place_1 <- nys_place %>% filter(NAME == "Niagara Falls")   # Try "Buffalo" or "Tonawanda"

# export start. Specify file name
pdf("plot/map2_NYS_places_niagara.pdf")
    # Set mode to "plot" to export
tmap_mode("plot")
tm_shape(nys_place) + tm_polygons() +
  tm_shape(place_1) + tm_polygons(alpha=0.5, col="white", border.col = "red", lwd = 2)
dev.off()
```

Commented [fyin7]: Add another layer of map.

7) Map of NYS health facilities (points)

```
# ====== 7. Map of health facilities =======
# read spatial points for health facilities in NYS
nys_health <- st_read("data/nys_health_facility/nys_health_facility.shp")

# Interactive map, set the plot mode to "view"
tmap_mode("view")
tm_shape(nys_health) + tm_dots(col="pink", alpha=1)
```

Commented [fyin8]: File location. If you want to load other spatial data, change the file location here.

Commented [fyin9]: The data of health facilities is point data. This time, we use "tm_dots" instead of "tm_polygons".

"col" = color of dots
"alpha" = transparency level. 1 = not transparent.

8) Map of k-12 schools (points)

```
# ====== 8. Map of k-12 schools =======
# read spatial points for k-12 in NYS
nys_school <- st_read("data/nys_k_12_schools/Schools_K-12.shp")

# Interactive map, set the plot mode to "view"
tmap_mode("view")
tm_shape(nys_school) + tm_dots(col="purple",  alpha=0.5)
```

9) Overlapping NYS places, the city of Buffalo, with health facilities and k-12 schools

```
# ====== 9. Map and export health facilities, schools in Buffalo =======
# read spatial data of NYS places and extract the record: NAME== "Buffalo"
nys_place <- st_read("data/tl_2022_36_place/tl_2022_36_place.shp")
place_buff <- nys_place %>% filter(NAME == "Buffalo")
# read spatial points for health facilities & schools in NYS
nys_health <- st_read("data/nys_health_facility/nys_health_facility.shp")
nys_school <- st_read("data/nys_k_12_schools/Schools_K-12.shp")

# export start
pdf("plot/map3_buff_places_school_health.pdf")
# Static map for export, set the plot mode to "plot"
tmap_mode("plot")
tm_shape(place_buff) + tm_polygons(alpha=0, border.col = "black", lwd = 2) +
  tm_shape(nys_health) + tm_dots(alpha=0.5, col="pink", size=0.2) +
  tm_shape(nys_school) + tm_dots(alpha=0.5, col="purple", size= 0.2)
dev.off()
```

10) Map census tracts

```
# ====== 10. Map NYS Census Tract and export (Add compass and legend bar) =======
nys_tract <- st_read("data/tl_2021_36_tract/tl_2021_36_tract.shp")

# export start
pdf("plot/map4_nys_tract.pdf")
    # map
tmap_mode("plot")
tm_shape(nys_tract) + tm_polygons(alpha=0.5, col="azure", border.col = "black", lwd = 1) +
  tm_compass(north = 0, position = c("left", "bottom")) +
  tm_scale_bar(position = c("left", "bottom"), width = 0.15)
dev.off()
# export end
```

Commented [fyin10]: Add compass and scale bar.

Change the position to the top and right:
position = c("right", "top").

11) Extract all census tracts located in Erie County

Federal Information Processing System (FIPS) Codes for States and Counties:
https://transition.fcc.gov/oet/info/maps/census/fips/fips.txt

NYS Erie county FIPS is "36029". First two digits are state code and the last three digits are county code.

```
# ====== 11. Extract Erie County (NYS) Census Tract and export =======
# read all census tract data boundaries in NYS
nys_tract <- st_read("data/tl_2021_36_tract/tl_2021_36_tract.shp")
# extract only census tracts in Erie county based on COUNTYFP
erie_tract <- nys_tract %>% filter(COUNTYFP=="029")

# export
pdf("plot/map5_erie_tract.pdf")
    # map
tmap_mode("plot")
tm_shape(nys_tract) + tm_polygons(alpha=0.5, col="azure", border.col = "black", lwd = 1) +
  tm_shape(erie_tract) + tm_polygons(col="coral", border.col = "black", lwd = 1) +
  tm_compass(north = 0, position = c("left", "bottom")) +
  tm_scale_bar(position = c("left", "bottom"), width = 0.15)

dev.off()
```

Commented [fyin11]: Extract all census tracts within a particular county by filtering based on the "COUNTYFP" value.

 # Niagara County COUNTYFP=="063"
Try to replace "029" using "063".

12) Map of water body in Erie County

```
# ====== 12. Map water body in Erie County (NYS) =======
# read spatial data (.shp) for water body in Erie county
erie_water <- st_read("data/tl_2022_36029_areawater/tl_2022_36029_areawater.shp")

# export
pdf("plot/map6_erie_water.pdf")
    # map
tmap_mode("plot")
tm_shape(erie_water) + tm_polygons(alpha=0.5, col="cyan", border.col = "blue", lwd = 0.5) +
  tm_compass(north = 0, position = c("left", "bottom")) +
  tm_scale_bar(position = c("left", "bottom"), width = 0.15)
dev.off()
```

Commented [fyin12]: File location.

Commented [fyin13]: "alpha" = transparency
"col" = filling color
"border.col" = border color
"lwd" = line width.

13) Overlapping Erie County census tracts with water body map

```
# ====== 13. Overlap Erie census tracts with water body =======
# read spatial data
erie_water <- st_read("data/tl_2022_36029_areawater/tl_2022_36029_areawater.shp")
# read all census tract in NYS and extract a county based on COUNTYFP
nys_tract <- st_read("data/tl_2021_36_tract/tl_2021_36_tract.shp")
erie_tract <- nys_tract %>% filter(COUNTYFP=="029")

# export
pdf("plot/map7_erie_tract_water.pdf")
    # map
tmap_mode("plot")
tm_shape(erie_tract) + tm_polygons(alpha=0.1, col = "grey", border.col = "black", lwd = 0.5) +
  tm_shape(erie_water) + tm_polygons(alpha=0.5, col="cyan", border.col = "blue", lwd = 0.5) +
  tm_compass(north = 0, position = c("left", "bottom")) +
  tm_scale_bar(position = c("left", "bottom"), width = 0.15)
dev.off()
```

14) Add NYS places boundaries to the map.

```
# ====== 14. Overlap census tracts, water body and places =======
# read spatial boundaries for NYS places and water body
nys_place <- st_read("data/tl_2022_36_place/tl_2022_36_place.shp")
erie_water <- st_read("data/tl_2022_36029_areawater/tl_2022_36029_areawater.shp")

# read all census tract in NYS and extract a county based on COUNTYFP
nys_tract <- st_read("data/tl_2021_36_tract/tl_2021_36_tract.shp")
erie_tract <- nys_tract %>% filter(COUNTYFP=="029")

# export
pdf("plot/map8_erie_tract_water_place_layout.pdf")
    # map
tmap_mode("plot")
tm_shape(erie_tract) + tm_polygons(alpha=0.1, col = "grey", border.col = "black", lwd = 0.5) +
  tm_shape(erie_water) + tm_polygons(alpha=0.5, col="cyan", border.col = "blue", lwd = 0.5) +
  tm_shape(nys_place) + tm_polygons(alpha = 0, border.col = "red", lwd = 1.5, lty="dashed") +
  tm_compass(north = 0, position = c("left", "bottom")) + tm_scale_bar(position = c("left",
"bottom"), width = 0.15)
dev.off()
```

Commented [fyin14]: "lty" = line type. The default values is "solid". Other values include "dotted", "dotdash", "longdash".

15) Add additional layers: health facilities and schools

```
# ====== 15. Overlap census tracts, water body, places, health facility, schools =======
# read spatial boundaries for NYS places and water body
nys_place <- st_read("data/tl_2022_36_place/tl_2022_36_place.shp")
erie_water <- st_read("data/tl_2022_36029_areawater/tl_2022_36029_areawater.shp")

# read spatial points for health facilities & schools in NYS
nys_health <- st_read("data/nys_health_facility/nys_health_facility.shp")
nys_school <- st_read("data/nys_k_12_schools/Schools_K-12.shp")

# read all census tract in NYS and extract a county based on COUNTYFP
nys_tract <- st_read("data/tl_2021_36_tract/tl_2021_36_tract.shp")
erie_tract <- nys_tract %>% filter(COUNTYFP=="029")

# export
pdf("plot/map9_erie_tract_water_place_hospital_school.pdf")
    # map
tmap_mode("plot")
tm_shape(erie_tract) + tm_polygons(alpha=0.1, col = "grey", border.col = "black", lwd = 0.5) +
  tm_shape(erie_water) + tm_polygons(alpha=0.5, col="cyan", border.col = "blue", lwd = 0.5) +
  tm_shape(nys_place) + tm_polygons(alpha = 0, border.col = "red", lwd = 1.5, lty="dashed") +
  tm_shape(nys_health) + tm_dots(alpha=0.5, col="pink", size=0.1) +
  tm_shape(nys_school) + tm_dots(alpha=0.5, col="purple", size= 0.1) +
  tm_compass(north = 0, position = c("left", "bottom")) +
  tm_scale_bar(position = c("left", "bottom"), width = 0.15)
dev.off()
```

16) Read and clean tabular data from US Census Bureau

```
# ====== 16. Read Tabular data from US Census Bureau =======
# Census Tract level. American Community Survey 5-Year
tab_census_pop <- read.csv("data/ACSST5Y2021.S0101_2023-10-12T000852/ACSST5Y2021.S0101-
Data.csv")
tab_census_pop$GEO_ID <- gsub("1400000US","", tab_census_pop$GEO_ID)
```

Commented [fyin15]: File location.

Commented [fyin16]: Clean the variable of "GEO_ID" by removing the first several letters "**1400000US".** This is the same for all American Community Survey data.

17) Join tabular data with spatial data at census tract level

```
# ====== 17. Join Tabular data with spatial data (Census Tract) =======
# Read tabular census data - tract level. American Community Survey 5-Year
tab_census_pop <- read.csv("data/ACSST5Y2021.S0101_2023-10-12T000852/ACSST5Y2021.S0101-
Data.csv")
tab_census_pop$GEO_ID <- gsub("1400000US","", tab_census_pop$GEO_ID)

# Read spatial data of census tracts and extract those in Erie
nys_tract <- st_read("data/tl_2021_36_tract/tl_2021_36_tract.shp")
erie_tract <- nys_tract %>% filter(COUNTYFP=="029")

# Join
erie_tract_pop <- left_join(erie_tract, tab_census_pop, by=c("GEOID"="GEO_ID"))
# identify the variable to map (S0101_C01_001E: Estimate!!Total!!Total population)
erie_tract_pop$S0101_C01_001E <- as.numeric(erie_tract_pop$S0101_C01_001E)

# export
pdf("plot/map10_erie_tract_population.pdf")
    # map + scale bar and north arrow on the map
tmap_mode("plot")
tm_shape(erie_tract_pop) + tm_polygons(col = "S0101_C01_001E", border.col = "black", lwd = 0.5)
+
  tm_compass(north = 0, position = c("left", "bottom")) + tm_scale_bar(position = c("left",
"bottom"), width = 0.15)
dev.off()
```

Commented [fyin17]: "**S0101_C01_001E**" column for total population.

Check the meta data that is downloaded together with the table for more column names..

Commented [fyin18]: Color the map based on the variable of "total population". See the previous comment for more explanation on variable name.

18) Change color palette for the thematic map.

```
# ====== 18. Design thematic map =======
# Read tabular census data - tract level. American Community Survey 5-Year
tab_census_pop <- read.csv("data/ACSST5Y2021.S0101_2023-10-12T000852/ACSST5Y2021.S0101-
Data.csv")
tab_census_pop$GEO_ID <- gsub("1400000US","", tab_census_pop$GEO_ID)

# Read spatial data of census tracts and extract those in Erie
nys_tract <- st_read("data/tl_2021_36_tract/tl_2021_36_tract.shp")
erie_tract <- nys_tract %>% filter(COUNTYFP=="029")

# Join
erie_tract_pop <- left_join(erie_tract, tab_census_pop, by=c("GEOID"="GEO_ID"))
# identify the variable to map (S0101_C01_001E: Estimate!!Total!!Total population)
erie_tract_pop$S0101_C01_001E <- as.numeric(erie_tract_pop$S0101_C01_001E)

# export
pdf("plot/map11_erie_tract_population.pdf")
    # map + scale bar and north arrow on the map
tmap_mode("plot")
tm_shape(erie_tract_pop) +
  tm_polygons(col = "S0101_C01_001E", border.col = "black", lwd = 0.5, palette = "YlGn", n=10)
+
  tm_compass(north = 0, position = c("left", "bottom")) + tm_scale_bar(position = c("left",
"bottom"), width = 0.15)
dev.off()
```

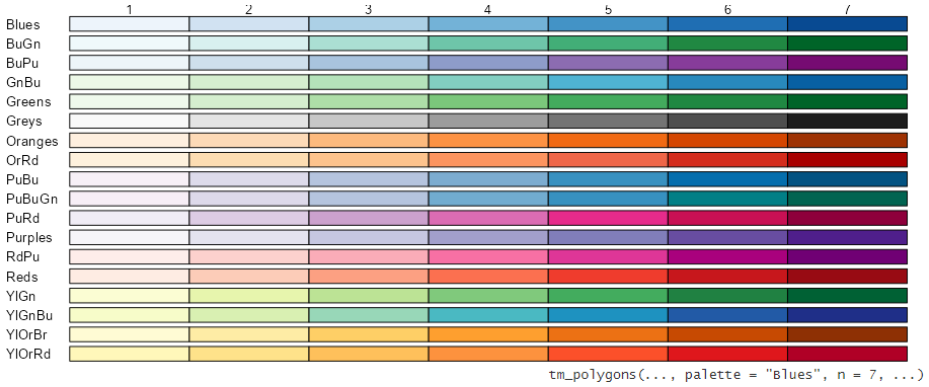
Commented [fyin19]: Use a predefined color palette called "YlGn", and break the values into 10 categories.

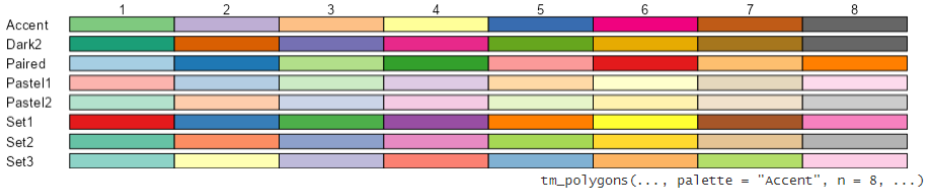
See appendix for more color palettes.

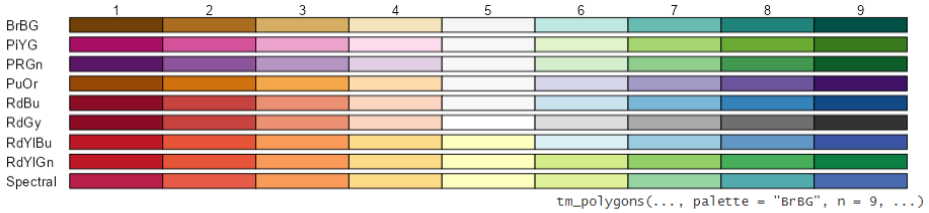
5.5. Save and close

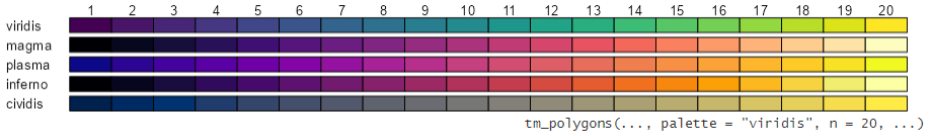
- File--> Save all
- File --> Close Project.
- To open a saved project: File --> Open project --> Navigate to and open the folder “Project_1”--> Click “Project_1.**Rproj**” --> Open

6. Appendix – Color code in R

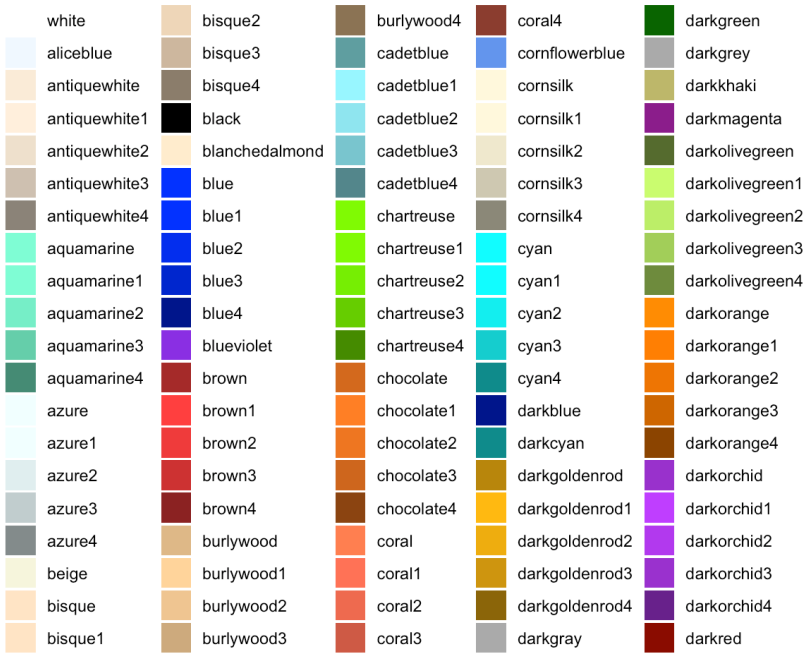








#1


white
aliceblue
antiquewhite
antiquewhite1
antiquewhite2
antiquewhite3
antiquewhite4
aquamarine
aquamarine1
aquamarine2
aquamarine3
aquamarine4
azure
azure1
azure2
azure3
azure4
beige
bisque
bisque1
bisque2
bisque3
bisque4
black
blanchedalmond
blue
blue1
blue2
blue3
blue4
blueviolet
brown
brown1
brown2
brown3
brown4
burlywood
burlywood1
burlywood2
burlywood3
burlywood4
cadetblue
cadetblue1
cadetblue2
cadetblue3
cadetblue4
chartreuse
chartreuse1
chartreuse2
chartreuse3
chartreuse4
chocolate
chocolate1
chocolate2
chocolate3
chocolate4
coral
coral1
coral2
coral3
coral4
cornflowerblue
cornsilk
cornsilk1
cornsilk2
cornsilk3
cornsilk4
cyan
cyan1
cyan2
cyan3
cyan4
darkblue
darkcyan
darkgoldenrod
darkgoldenrod1
darkgoldenrod2
darkgoldenrod3
darkgoldenrod4
darkgray
darkgreen
darkgrey
darkkhaki
darkmagenta
darkolivegreen
darkolivegreen1
darkolivegreen2
darkolivegreen3
darkolivegreen4
darkorange
darkorange1
darkorange2
darkorange3
darkorange4
darkorchid
darkorchid1
darkorchid2
darkorchid3
darkorchid4
darkred