Lab 4 - Making Static Maps

Bailey McNichol

2021 November 11

Lab 04: Making static maps

Read the instructions COMPLETELY before starting the lab

This lab builds on many of the discussions and exercises from class, including the "frankenmap" exercise from class.

Formatting your submission

This lab must be placed into a public repository on GitHub (www.github.com). Before the due date, submit **on Canvas** a link to the repository. I will then download your repositories and run your code. The code must be contained in either a .R script or a .Rmd markdown document. As I need to run your code, any data you use in the lab must be referenced using **relative path names**. Finally, answers to questions I pose in this document must also be in the repository at the time you submit your link to Canvas. They can be in a separate text file, or if you decide to use an RMarkdown document, you can answer them directly in the doc.

Introduction

This lab is much more free-form than previous assignments. You will be completing your own version of the in-class "frankenmap" using proper cartographic principles. I encourage you to use whatever resources you find useful, including the relevant sections of the Lovelace chapter and online resources such as: https://mgimond.github.io/Spatial/good-map-making-tips.html

Load in packages and data

```
library(tidyverse)
library(GISTools)
library(sf)
library(tmap)
library(raster)
library(grid)

# Nebraska Counties
counties <- sf::read_sf("../data/County_Boundaries-_Census.shp") %>% sf::st_make_valid()
# lancaster = CRS arguments: +proj=longlat +datum=WGS84 +no_defs
# Municipal Boundaries
mb <- sf::read_sf("../data/Municipal_Boundaries.shp") %>% sf::st_make_valid()
# State Parks
parks <- sf::read_sf("../data/State_Park_Locations.shp") %>% sf::st_make_valid()
```

```r
# Streams
streams <- sf::read_sf("../data/Streams_303_d_.shp") %>% sf::st_make_valid()
# Lancaster County DEM
LC_dem <- raster::raster("../data/lancaster_dem/lc_dem.tif")
# Re-project DEM to match coordinate reference system of other layers
crs(LC_dem) <- "+proj=longlat +zone=14 +datum=WGS84 +units=m +no_defs"
```

Your tasks

1. Using the same descriptions as the in-class frankenmap, create a map that follows "good" cartographic principles. I have included the instructions below for reference.

Original description:

- State data frame:
  - Nebraska counties, symbolized (filled) by some variable of interest
  - Borders between counties symbolized using something other than the defaults
  - A scale bar

See the step-by-step annotated code below to create the map of Nebraska counties, color-coded by median age.

```r
# Subset Nebraska data to only include Lancaster County
lancaster <- counties %>%
           dplyr::filter(NAME10 == "Lancaster")

# Add Nebraska counties shapefile
State <- tm_shape(counties) +
  # Color code the counties by their median age, modifying legend title to be more legible
  tm_fill("MedianAge", title = "Median Age") +
  tm_shape(counties) +
  # Change the borders to dashed black lines
  tm_borders(col = "black", lty = "dashed") +
  # Adjust the legend to avoid overlap with the map
  tm_layout(legend.stack = "vertical", legend.title.size = 1,
            legend.bg.color = "white",
            legend.bg.alpha = 0.5,
            inner.margins=c(0.05,0.01,0.01,0.01),
            legend.position = c("left", "bottom")) +
  # Add Lancaster County
  tm_shape(lancaster) +
  # Highlight county boundary with a thick red line for inset
  tm_borders(lwd = 3, col = "red")
```

- County data frame:
  - Municipal boundaries within Lancaster County, with labels for names
  - State parks in Lancaster County, symbolized using a non-default symbol
  - 303d streams in Lancaster County, symbolized differently by "impairment"
  - Use the DEM to plot elevation behind a semi-transparent Lancaster County (I have placed a new DEM in the repo's data folder)

See the step-by-step annotated code below to create a map of Lancaster County, showing the elevations via the DEM, the municipal boundaries, streams color-coded based on impairment, and state parks, also including a north arrow, scale bar, and title.

```
# Intersect Lancaster County and Statewide Municipal Boundaries
mb.lan <- st_intersection(mb, lancaster)
```

```
## Warning: attribute variables are assumed to be spatially constant throughout all
## geometries
```

```
# Intersect Lancaster County and Streams
stream.lan <- st_intersection(streams, lancaster)
```

```
## Warning: attribute variables are assumed to be spatially constant throughout all
## geometries
```

```
# Determine the bounding of Lancaster County with st_bbox()
st_bbox(lancaster)
```

```
##      xmin      ymin      xmax      ymax
## -96.91394  40.52302 -96.46363  41.04612
```

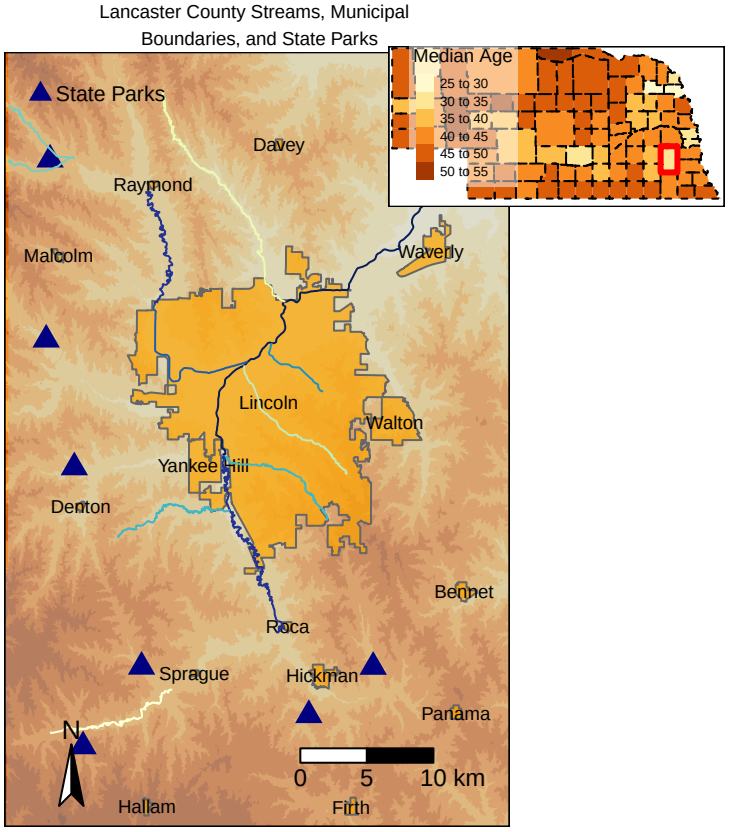
```
# Create bounding region based on eastings and northings (for creating the full map with inset)
region <- st_bbox(c(xmin = -96.91394, xmax = -96.4636,
                    ymin = 40.5230, ymax = 41.04612),
                    crs = st_crs(lancaster)) %>% st_as_sfc()

# Add Lancaster County DEM to show elevation, and place within boundary box
County <- tm_shape(LC_dem, bbox = region) +
  # Remove legend to avoid overlap
  tm_raster(legend.show = FALSE) +
  # Add Lancaster County shapefile
  tm_shape(lancaster) +
  # Fill in a transparent gray to overlay on DEM
  tm_fill(col = "gray", alpha = 0.5) +
  # Add Municipal Boundaries in Lancaster County
  tm_shape(mb.lan) +
  # Fill in semi-transparent orange
  tm_polygons(col = "orange", alpha = 0.7) +
  # Label municipalities
  tm_text("NAME", size = 0.6) +
  # Add State Parks in Lancaster County
  tm_shape(parks) +
  # Make navy blue triangles and reduce default size
  tm_symbols(shape = 17, col = "navy", size = 0.5) +
  # Add a legend for the state parks since they are not individually labelled
  tm_add_legend(type = c("symbol"), shape = c(17),
                col = c("navy"), labels = "State Parks", ) +
  # Add streams in Lancaster County
  tm_shape(stream.lan) +
  # Color-code by their impairment
  tm_lines(col = "Impairment", palette = "YlGnBu", legend.col.show = FALSE) +
```

```
  # Add an informative map title, centered and broken into two lines
  tm_layout(main.title = "Lancaster County Streams, Municipal \n Boundaries, and State Parks",
            main.title.position = "center", main.title.size = 0.6) +
  # Add a north arrow
  tm_compass(type = "arrow", position = c("left", "bottom"), size = 2) +
  # Add a scale bar
  tm_scale_bar(breaks = c(0,5,10), text.size = 0.75)
```

- Putting it together
  - Put the state map as an inset to the county map
  - A north arrow
  - A title

The north arrow and title were added above; below, I map the county boundary box ("County"), then add the state map ("State") as an inset using the viewport() function.

```
# Plot the complete Lancaster County map
County
# Add the inset Nebraska state map in the top right corner using viewport(); the first two arguments sp
print(State, vp = viewport(0.74, 0.83, width = 0.28, height = 0.28))
```



2. Make a second static map of your choosing. You may choose any spatial extent, domain, or technique. I'm looking for creativity, good coding practices (including comments), and for you to demonstrate independent thinking. There are minor restrictions you must follow:

1. It must include vector AND raster data in some manner
2. It must include spatial data relating to a social process (e.g., political boundaries) AND spatial data relating to an environmental process (e.g., water resources)
3. The map should "stand on its own" and communicate its purpose without additional text
4. That's it!

For my map, I downloaded a 1:250,000 raster image file of New York City (NYC) County in New York State. I then found shapefiles including the NYC Community Districts (political boundaries), NYC parks, and NYC Community Health Survey Data, which comes from a city-wide survey estimating occurrence of a wide variety of chronic diseases and behavioral risk factors. Below, I plot the raster, and overlay the political districts, color-coded by the percentage of people experiencing depression in each, and the locations of parks, which are color-coded based on size.

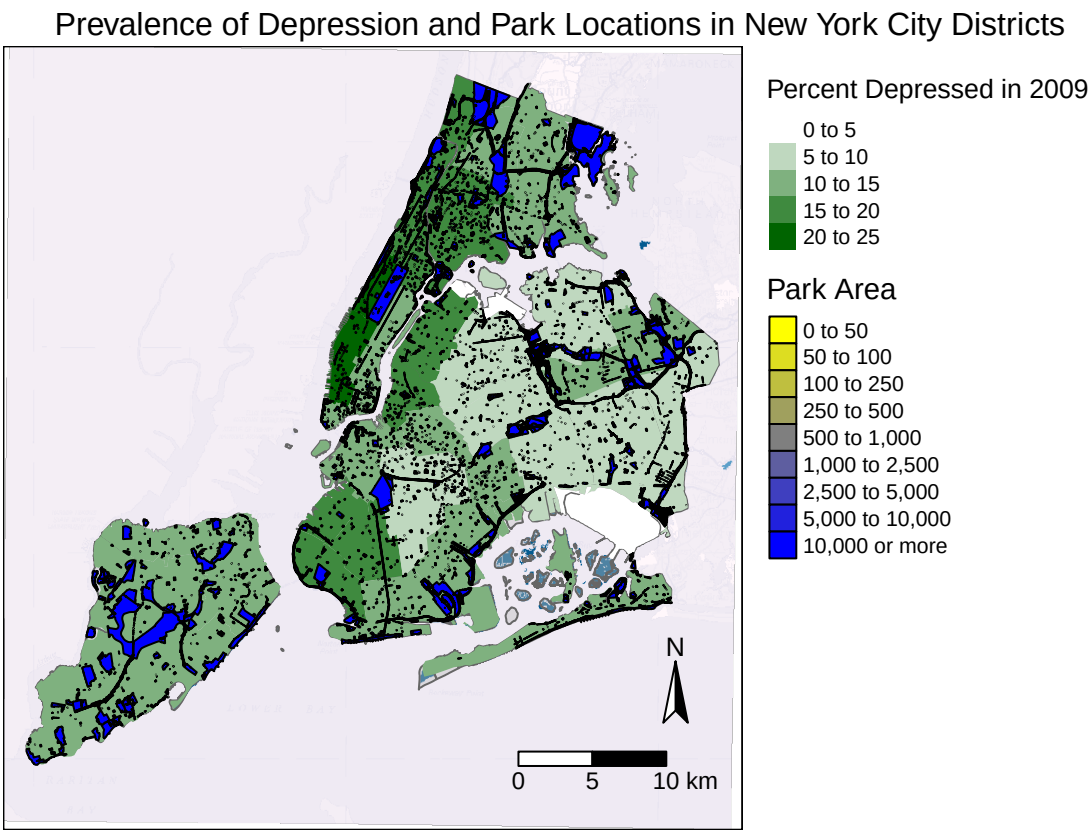
```
# New York City County Raster
nyc.ras <- raster::raster("../data/nyc250/nyc250.tif")
# Re-project NYC raster to WGS84 to align with districts and parks (below)
crs(nyc.ras) <- "+proj=longlat +zone=18 +datum=WGS84 +units=m +no_defs"

# New York City Community Districts
nyc.dist <- sf::read_sf("../data/Community Districts/geo_export_264abbf3-0041-4b44-8618-583376125eb3.sh

# New Yory City Parks
nyc.parks <- sf::read_sf("../data/Open Space (Parks)/geo_export_1a3abf2e-534d-4871-8173-c432dcb113f8.sh

# New Yory City Community Health Survey Data
nyc.chs <- sf::read_sf("../data/CHS_2009_DOHMH_2010B/CHS_2009_DOHMH_2010B.shp")
# Transform NYC Community Health Survey to WGS84 CRS to align with districts and parks
st_transform(nyc.chs, "WGS84")
```

```
## Simple feature collection with 35 features and 79 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -74.25589 ymin: 40.49593 xmax: -73.70027 ymax: 40.91577
## Geodetic CRS:  WGS 84
## # A tibble: 35 x 80
##    UHF34_CODE FIRST_BORO FIRST_UHF_        UHF_CODE UHF_Code_1 asthev2 asthev2_r
##  *      <dbl> <chr>      <chr>                <dbl>      <dbl>   <dbl>     <dbl>
##  1        101 Bronx      Kingsbridge - Ri~      101        101    13.4         0
##  2        102 Bronx      Northeast Bronx        102        102    14.7         0
##  3        103 Bronx      Fordham - Bronx ~      103        103    16.9         0
##  4        104 Bronx      Pelham - Throgs ~      104        104    16.7         0
##  5        201 Brooklyn   Greenpoint             201        201    12.6         0
##  6        202 Brooklyn   Downtown  - Heig~      202        202    19.1         0
##  7        203 Brooklyn   Bedford Stuyvesa~      203        203    11.6         0
##  8        204 Brooklyn   East New York          204        204    12.2         0
##  9        205 Brooklyn   Sunset Park            205        205     8.4         1
## 10        206 Brooklyn   Borough Park           206        206     4.1         1
## # ... with 25 more rows, and 73 more variables: asthev2_s <dbl>, bmicat3 <dbl>,
## #   bmicat3_r <dbl>, bmicat3_s <dbl>, bphigh2 <dbl>, bphigh2_r <dbl>,
## #   bphigh2_s <dbl>, bpmeds2 <dbl>, bpmeds2_r <dbl>, bpmeds2_s <dbl>,
## #   cndmal2 <dbl>, cndmal2_r <dbl>, cndmal2_s <dbl>, cotest2 <dbl>,
## #   cotest2_r <dbl>, cotest2_s <dbl>, depres2 <dbl>, depres2_r <dbl>,
```

```
## #   depres2_s <dbl>, diabet2 <dbl>, diabet2_r <dbl>, diabet2_s <dbl>,
## #   exercs2 <dbl>, exercs2_r <dbl>, exercs2_s <dbl>, flusht2 <dbl>, ...
```

```
## MAP
# Map the NYC raster - used raster.downsample = TRUE to reduce number of raster cell groups
tm_shape(nyc.ras, raster.downsample = TRUE) +
  # Color-coded the gradient using quantiles and a purple-blue color palette
    tm_raster(style = "quantile", palette = "PuBu", legend.show = FALSE) +
# Map the NYC Community Districts
tm_shape(nyc.dist) +
    tm_polygons(col = "gray", alpha = 0.4) +
# Map the Percent of People Experiencing Depression, by Area
# Read in the NYC Community Health Survey Data
tm_shape(nyc.chs) +
# Color-code community districts by rates of depression
    tm_fill(col = "depres2", title = "Percent Depressed in 2009",
          palette=c("white","darkgreen")) +
    tm_layout(legend.outside = TRUE, legend.outside.position = "right") +
# Map the NYC parks shapefile
tm_shape(nyc.parks) +
# Color-code by park area
  tm_fill(col = "shape_area", title = "Park Area",
          breaks = c(0,50,100,250,500,1000,2500,5000,10000,Inf),
          palette=c("yellow","blue")) +
  tm_borders(col = "black") +
# Add plot title and adjust legend
  tm_layout(legend.outside = TRUE, legend.outside.position = "right",
            main.title = "Prevalence of Depression and Park Locations in New York City Districts",
            main.title.position = "center", main.title.size = 1) +
# Add a north arrow
  tm_compass(type = "arrow", position = c("right", "bottom"), size = 2) +
# Add a scale bar
  tm_scale_bar(breaks = c(0,5,10), text.size = 0.75)
```

```
## Warning: Number of levels of the variable "NA" is 255, which is
## larger than max.categories (which is 30), so levels are combined. Set
## tmap_options(max.categories = 255) in the layer function to show all levels.
```



Questions:

1. Describe your choices in making map 1

I decided to color code Nebraska's counties by median resident age, as this is an informative way to look at population demographics throughout the state. I also made sure to highlight Lancaster County (Co.) on the state map inset; given the shape of the county, it may not have otherwise initially been clear to a non-initiated person what extent of the larger-area map was being shown. In creating the Lancaster Co. map, I first established the extent of the bounding box by examining the x and y min/max for Lancaster Co. I layered on the county DEM first, next adding the transparent county shapefile followed by the municipal boundaries (on which I had performed a spatial intersection to restrict these to only within the county). I then added the Lancaster Co. state parks as blue triangles; because I did not include their names and it would otherwise not have been clear, I also decided to add a manual legend specifiying that each triangle is a park. The last layer that I added was the streams, color-coded by impairment, and finally the north arrow, scale bar, and title to facilitate interpretation. I initally had some trouble with the viewport() function when plotting the inset map (RMarkdown's extent doesn't allow you to see the whole map..), but I understand how it works now!

2. Describe your choices in making map 2. Include why you chose the problem and where you obtained your data. Finally, your map is a communication piece. What was the intent of your communication and do you feel as though you achieved your goal?

I initially had a difficult time finding a simple raster file that was not huge, and finally found a simple one for the county of New York City. Then, I searched for corresponding data on natural features and demographics. I first found a shapefile with all of the parks in NYC, and then found a political boundary map of the NYC community districts and associated data from a city-wide Community Health Survey. As natural areas have been shown to have positive effects on mental health, I was interested in comparing the spatial distributions of city parks versus the prevalence (percent) of depressed individuals by district. I again started with my base raster, then adding the community districts, color-coding them by the percent of depressed people (as of the 2009 survey data). I then overlayed a shapefile of the city parks, color-coded from yellow to blue by area (the yellow are challenging to see because they are so small compared to the map extent). Finally, I added a title, north arrow, and scale bar to make the map more readily interpretable. Without running any statistical analysis (on observation alone), it does not appear that communities with lower percentages of depression are closer to city parks compared to high-depression communities. This is not particularly surprising, as there are a multitude of genetic and socioeconomic factors influencing mental health outcomes that are not represented here.

3. What did you learn?

I learned a lot about the functionality of the different arguments in tmap() - how to format legends in particular was useful. I also am glad to know how to make a map with an inset for future reference! I also learned that I need to sharpen my skills on looking for good spatial data online! It took me longer than it should have to find a solid, not huge tif raster file that I could bring into R easily to make the second map. I can see why it would be useful to transform files in ESRI prior to bringing them into R sometimes!