MDSR2e Ch 17: Working with geospatial data

This is an abbreviated version of the full version in R > DS2, just to play with GitHub. Focus on NC Congressional districts. No leaflet since that needs html.

Goals

- To

Required reading

- Chapter 5 of your textbook

New Code

- `mosaic::favstats(dataset$var)`, provides summary statistics for variable `var` from `dataset`

Before class

Let's start by loading a subset of data used for the story by doing the following command in R

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(mdsr)
library(sf)
```

```
## Linking to GEOS 3.7.2, GDAL 3.0.4, PROJ 6.3.2; sf_use_s2() is TRUE
```

```
library(ggspatial)

#| Section 17.4: Extended example - Congressional districts

#| Section 17.4.1: Election results

library(fec12)
results_house |>
  group_by(state, district_id) |>
  summarize(N = n()) |>
  nrow()
```

```
## `summarise()` has grouped output by 'state'. You can override using the
## `.groups` argument.
```

```
## [1] 445
```

```
results_house |>
  left_join(candidates, by = "cand_id") |>
  select(state, district_id, cand_name, party, general_votes) |>
  arrange(desc(general_votes)) |> filter(state == "NC") |> count(party) |>
  print(n = Inf)
```

```
## # A tibble: 4 x 2
##   party     n
##   <chr> <int>
## 1 D        24
## 2 LIB       3
## 3 R        48
## 4 W         1
```

```
district_elections <- results_house |>
  mutate(district = parse_number(district_id)) |>
  group_by(state, district) |>
  summarize(   # pretty crude - e.g. won't count 14 DFL candidates
    N = n(),   #   but maybe okay for NC (24 D, 48 R, 3 LIB, 1 W)
    total_votes = sum(general_votes, na.rm = TRUE),
    d_votes = sum(ifelse(party == "D", general_votes, 0), na.rm = TRUE),
    r_votes = sum(ifelse(party == "R", general_votes, 0), na.rm = TRUE)
  ) |>
  mutate(
    other_votes = total_votes - d_votes - r_votes,
    r_prop = r_votes / total_votes,
    winner = ifelse(r_votes > d_votes, "Republican", "Democrat")
  )
```

```
## `summarise()` has grouped output by 'state'. You can override using the
## `.groups` argument.
```

```r
nc_results <- district_elections |>
  filter(state == "NC")
nc_results |>
  ungroup() |>   # book code doesn't work without ungroup()
  select(-state)
```

```
## # A tibble: 13 x 8
##    district     N total_votes d_votes r_votes other_votes r_prop winner
##       <dbl> <int>       <dbl>   <dbl>   <dbl>       <dbl>  <dbl> <chr>
##  1        1     4      338066  254644   77288        6134  0.229 Democrat
##  2        2     8      311397  128973  174066        8358  0.559 Republican
##  3        3     3      309885  114314  195571           0  0.631 Republican
##  4        4     4      348485  259534   88951           0  0.255 Democrat
##  5        5     3      349197  148252  200945           0  0.575 Republican
##  6        6     4      364583  142467  222116           0  0.609 Republican
##  7        7     4      336736  168695  168041           0  0.499 Democrat
##  8        8     8      301824  137139  160695        3990  0.532 Republican
##  9        9    13      375690  171503  194537        9650  0.518 Republican
## 10       10     6      334849  144023  190826           0  0.570 Republican
## 11       11    11      331426  141107  190319           0  0.574 Republican
## 12       12     3      310908  247591   63317           0  0.204 Democrat
## 13       13     5      370610  160115  210495           0  0.568 Republican
```

```r
nc_results |>   # summary stats of total_votes column (no na's to worry about)
  skim(total_votes) |>
  select(-na)
```

Variable type: numeric

var	state	n	mean	sd	p0	p25	p50	p75	p100
total_votes	NC	13	337204.3	24175.2	301824	311397	336736	349197	375690

```r
nc_results |>
  summarize(
    N = n(),
    state_votes = sum(total_votes),
    state_d = sum(d_votes),
    state_r = sum(r_votes)
  ) |>
  mutate(
    d_prop = state_d / state_votes,
    r_prop = state_r / state_votes
  )
```

```
## # A tibble: 1 x 7
##   state     N state_votes state_d state_r d_prop r_prop
##   <chr> <int>       <dbl>   <dbl>   <dbl>  <dbl>  <dbl>
## 1 NC       13     4383656 2218357 2137167  0.506  0.488
```

```r
nc_results |>
  select(district, r_prop, winner) |>
  arrange(desc(r_prop))
```

```
## Adding missing grouping variables: `state`
```

```
## # A tibble: 13 x 4
## # Groups:   state [1]
##    state district r_prop winner
##    <chr>    <dbl>  <dbl> <chr>
##  1 NC           3  0.631 Republican
##  2 NC           6  0.609 Republican
##  3 NC           5  0.575 Republican
##  4 NC          11  0.574 Republican
##  5 NC          10  0.570 Republican
##  6 NC          13  0.568 Republican
##  7 NC           2  0.559 Republican
##  8 NC           8  0.532 Republican
##  9 NC           9  0.518 Republican
## 10 NC           7  0.499 Democrat
## 11 NC           4  0.255 Democrat
## 12 NC           1  0.229 Democrat
## 13 NC          12  0.204 Democrat
```

```r
#| Section 17.4.2: Congressional districts

#src <- "http://cdmaps.polisci.ucla.edu/shp/districts113.zip"
#dsn_districts <- usethis::use_zip(src, destdir = fs::path("data_large"))

#| didn't use code above but downloaded zip file and uploaded it into R
dsn_districts <- fs::path("~/R/DS2/districtShapes")

st_layers(dsn_districts)
```

```
## Driver: ESRI Shapefile
## Available layers:
##     layer_name geometry_type features fields crs_name
## 1 districts113       Polygon      436     15    NAD83
```

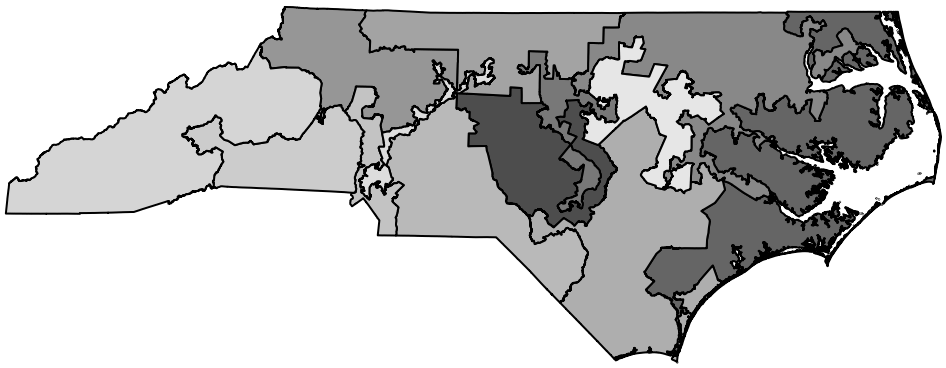
```r
districts <- st_read(dsn_districts, layer = "districts113") |>
  mutate(DISTRICT = parse_number(as.character(DISTRICT)))
```

```
## Reading layer `districts113' from data source
##   `/home/rstudio/users/roback/R/DS2/districtShapes' using driver `ESRI Shapefile'
## Simple feature collection with 436 features and 15 fields (with 1 geometry empty)
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -179.1473 ymin: 18.91383 xmax: 179.7785 ymax: 71.35256
## Geodetic CRS:  NAD83
```

```r
glimpse(districts)
```

```
## Rows: 436
## Columns: 16
## $ STATENAME  <chr> "Louisiana", "Maine", "Maine", "Maryland", "Maryland", "Mar~
## $ ID         <chr> "022113114006", "023113114001", "023113114002", "0241131140~
## $ DISTRICT   <dbl> 6, 1, 2, 1, 2, 3, 4, 5, 6, 7, 8, 1, 2, 3, 4, 5, 6, 7, 8, 9,~
## $ STARTCONG  <chr> "113", "113", "113", "113", "113", "113", "113", "113", "11~
## $ ENDCONG    <chr> "114", "114", "114", "114", "114", "114", "114", "114", "11~
## $ DISTRICTSI <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ COUNTY     <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ PAGE       <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ LAW        <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ NOTE       <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ BESTDEC    <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ FINALNOTE  <chr> "{\"From US Census website\"}", "{\"From US Census website\~
## $ RNOTE      <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ LASTCHANGE <chr> "2016-05-29 16:44:10.857626", "2016-05-29 16:44:10.857626",~
## $ FROMCOUNTY <chr> "F", "F", "F", "F", "F", "F", "F", "F", "F", "F", "F", "F",~
## $ geometry   <MULTIPOLYGON [°]> MULTIPOLYGON (((-91.82288 3..., MULTIPOLYGON (~
```

```r
nc_shp <- districts |>
  filter(STATENAME == "North Carolina")
nc_shp |>   # works but tons of white space around plot
  st_geometry() |>
  plot(col = gray.colors(nrow(nc_shp)))
```



```
#| Section 17.4.3: Putting it all together

nc_merged <- nc_shp |>
  st_transform(4326) |>
  inner_join(nc_results, by = c("DISTRICT" = "district"))
glimpse(nc_merged)
```

```
## Rows: 13
## Columns: 24
## $ STATENAME   <chr> "North Carolina", "North Carolina", "North Carolina", "Nor~
## $ ID          <chr> "037113114002", "037113114003", "037113114004", "037113114~
## $ DISTRICT    <dbl> 2, 3, 4, 1, 5, 6, 7, 8, 9, 10, 11, 12, 13
## $ STARTCONG   <chr> "113", "113", "113", "113", "113", "113", "113", "113", "1~
## $ ENDCONG     <chr> "114", "114", "114", "114", "114", "114", "114", "114", "1~
## $ DISTRICTSI  <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ COUNTY      <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ PAGE        <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ LAW         <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ NOTE        <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ BESTDEC     <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ FINALNOTE   <chr> "{\"From US Census website\"}", "{\"From US Census website~
## $ RNOTE       <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
## $ LASTCHANGE  <chr> "2016-05-29 16:44:10.857626", "2016-05-29 16:44:10.857626"~
## $ FROMCOUNTY  <chr> "F", "F", "F", "F", "F", "F", "F", "F", "F", "F", "F", "F"~
## $ state       <chr> "NC", "NC", "NC", "NC", "NC", "NC", "NC", "NC", "NC", "NC"~
```

```
## $ N           <int> 8, 3, 4, 4, 3, 4, 4, 8, 13, 6, 11, 3, 5
## $ total_votes <dbl> 311397, 309885, 348485, 338066, 349197, 364583, 336736, 30~
## $ d_votes     <dbl> 128973, 114314, 259534, 254644, 148252, 142467, 168695, 13~
## $ r_votes     <dbl> 174066, 195571, 88951, 77288, 200945, 222116, 168041, 1606~
## $ other_votes <dbl> 8358, 0, 0, 6134, 0, 0, 0, 3990, 9650, 0, 0, 0, 0
## $ r_prop      <dbl> 0.5589842, 0.6311083, 0.2552506, 0.2286181, 0.5754488, 0.6~
## $ winner      <chr> "Republican", "Republican", "Democrat", "Democrat", "Repub~
## $ geometry    <MULTIPOLYGON [°]> MULTIPOLYGON (((-80.05325 3..., MULTIPOLYGON (((-78.27217 ~
```

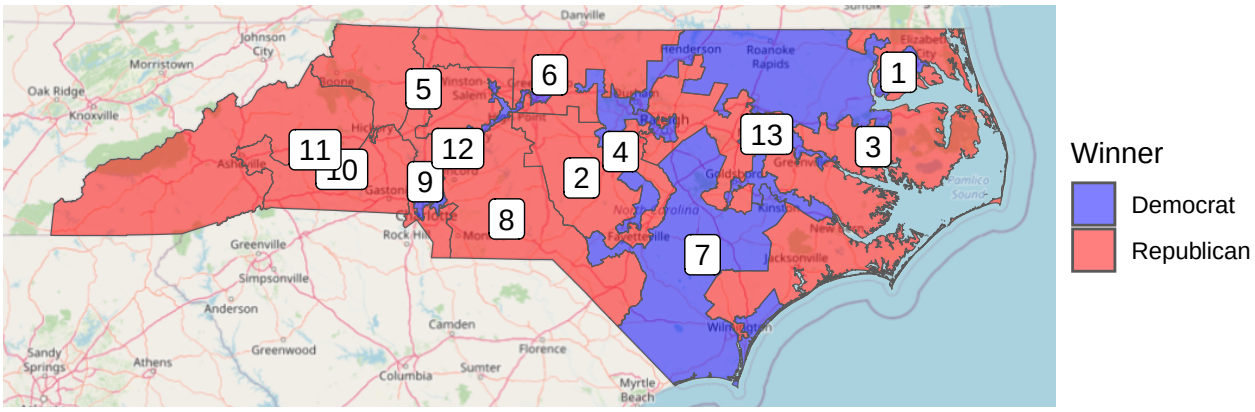
```
#| Section 17.4.4: Using ggplot2

nc <- ggplot(data = nc_merged, aes(fill = winner)) +
  annotation_map_tile(zoom = 7, type = "osm") +
  geom_sf(alpha = 0.5) +
  scale_fill_manual("Winner", values = c("blue", "red")) +
  geom_sf_label(aes(label = DISTRICT), fill = "white") +
  theme_void()
nc   # background pretty fuzzy if use, say, zoom = 5
```

```
## Warning in st_point_on_surface.sfc(sf::st_zm(x)): st_point_on_surface may not
## give correct results for longitude/latitude data
```

```
## Loading required namespace: raster
## The legacy packages maptools, rgdal, and rgeos, underpinning the sp package,
## which was just loaded, will retire in October 2023.
## Please refer to R-spatial evolution reports for details, especially
## https://r-spatial.org/r/2023/05/15/evolution4.html.
## It may be desirable to make the sf package available;
## package maintainers should consider adding sf to Suggests:.
## The sp package is now running under evolution status 2
##      (status 2 uses the sf package in place of rgdal)
## Please note that rgdal will be retired during October 2023,
## plan transition to sf/stars/terra functions using GDAL and PROJ
## at your earliest convenience.
## See https://r-spatial.org/r/2023/05/15/evolution4.html and https://github.com/r-spatial/evolution
## rgdal: version: 1.6-7, (SVN revision 1203)
## Geospatial Data Abstraction Library extensions to R successfully loaded
## Loaded GDAL runtime: GDAL 3.0.4, released 2020/01/28
## Path to GDAL shared files: /usr/share/gdal
## GDAL binary built with GEOS: TRUE
## Loaded PROJ runtime: Rel. 6.3.2, May 1st, 2020, [PJ_VERSION: 632]
## Path to PROJ shared files: /usr/share/proj
## Linking to sp version:2.0-0
## To mute warnings of possible GDAL/OSR exportToProj4() degradation,
## use options("rgdal_show_exportToProj4_warnings"="none") before loading sp or rgdal.
## Zoom: 7
## Fetching 15 missing tiles
```

```
##   |                                                                              |
```

```
## ...complete!
```

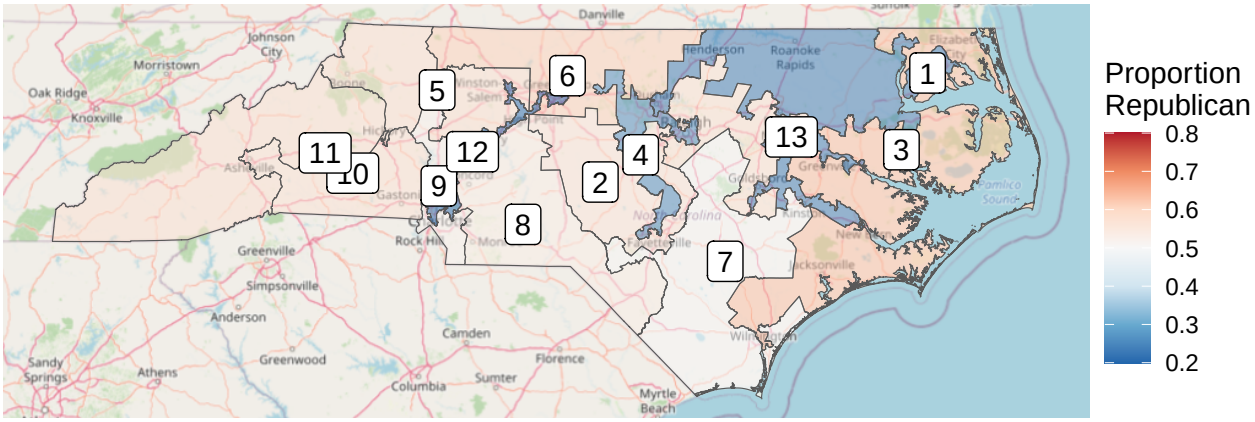



```
nc +
  aes(fill = r_prop) +
  scale_fill_distiller(
    "Proportion\nRepublican",
    palette = "RdBu",
    limits = c(0.2, 0.8)
  )
```

```
## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.
```

```
## Warning in st_point_on_surface.sfc(sf::st_zm(x)): st_point_on_surface may not
## give correct results for longitude/latitude data
```

```
## Zoom: 7
```


Danville
Johnson City
Morristown
Oak Ridge
Knoxville
Henderson
Roanoke Rapids
Greenville
Simpsonville
Anderson
Greenwood
Sandy Springs
Athens
Camden
Columbia
Florence
Sumter
Myrtle Beach
Jacksonville
1
2
3
4
5
6
7
8
9
10
11
12
13
Proportion Republican
0.8
0.7
0.6
0.5
0.4
0.3
0.2