Spatial Data and Visualization

Dav King

1-27-2022

Main Ideas

- Spatial data is important
  - exploratory data analysis
  - detecting spatial patterns and trends
  - understanding spatial data relationships
  - analysis of spatial data should reflect spatial structure

Coming Up

- HW 1 is due tomorrow (Friday).
- HW 2 goes out today.
- Lab 3 is due on Friday.

Hot Keys

Task / function	**Windows & Linux**	**macOS**
Insert R chunk	Ctrl+Alt+I	Command+Option+I
Knit document	Ctrl+Shift+K	Command+Shift+K
Run current line	Ctrl+Enter	Command+Enter
Run current chunk	Ctrl+Shift+Enter	Command+Shift+Enter
Run all chunks above	Ctrl+Alt+P	Command+Option+P
`<-`	Alt + -	Option + -
`%>%`	Ctrl+Shift+M	Command+Shift+M

Lecture Notes and Exercises

```
library(tidyverse)
library(sf)
```

Spatial data is different.

Our typical "tidy" dataframe.

```
mpg
```

```
## # A tibble: 234 x 11
##    manufacturer model      displ  year   cyl trans drv     cty   hwy fl    class
##    <chr>        <chr>      <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
##  1 audi         a4           1.8  1999     4 auto~ f        18    29 p     comp~
##  2 audi         a4           1.8  1999     4 manu~ f        21    29 p     comp~
##  3 audi         a4           2    2008     4 manu~ f        20    31 p     comp~
##  4 audi         a4           2    2008     4 auto~ f        21    30 p     comp~
##  5 audi         a4           2.8  1999     6 auto~ f        16    26 p     comp~
##  6 audi         a4           2.8  1999     6 manu~ f        18    26 p     comp~
##  7 audi         a4           3.1  2008     6 auto~ f        18    27 p     comp~
##  8 audi         a4 quattro   1.8  1999     4 manu~ 4        18    26 p     comp~
##  9 audi         a4 quattro   1.8  1999     4 auto~ 4        16    25 p     comp~
## 10 audi         a4 quattro   2    2008     4 manu~ 4        20    28 p     comp~
## # ... with 224 more rows
```

A new simple feature object.

```
nc <- st_read("nc_regvoters.shp", quiet = TRUE)
nc
```

```
## Simple feature collection with 100 features and 14 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -84.32385 ymin: 33.88199 xmax: -75.45698 ymax: 36.58965
## Geodetic CRS:  NAD27
## First 10 features:
##       county   dem   gop lib  unaf white black ntv_a ntv_h other hispanic  male
## 1   ALAMANCE 38209 35967 670 35196 70330 21377   259     8 18068     4658 44651
## 2  ALEXANDER  4772 11750 123  7967 21103   921    33     2  2553      364 10947
## 3  ALLEGHANY  2030  3005  33  2466  6596    70     8     0   860      183  3319
## 4      ANSON  9130  2858  38  3599  6267  6198    25     0  3135       83  5800
## 5       ASHE  4261  8804 102  6232 17501   112    23     1  1762      257  8609
## 6      AVERY  1343  6994  55  3673 10714    44    20     0  1287       80  5283
## 7   BEAUFORT 10883 11873 124  9426 22052  6961    35     1  3257      463 13591
## 8     BERTIE  8178  1629  36  2835  4468  7283    19     1   907       38  5310
## 9     BLADEN  9847  5005  77  6784 12113  7412   374     2  1812      444  9472
## 10 BRUNSWICK 26797 46557 618 42602 92487  8384   344     4 15355     1454 48199
##    female  total                       geometry
## 1   54529 110042 MULTIPOLYGON (((-79.24619 3...
## 2   11768  24612 MULTIPOLYGON (((-81.10889 3...
## 3    3548   7534 MULTIPOLYGON (((-81.23989 3...
## 4    6980  15625 MULTIPOLYGON (((-79.91995 3...
## 5    9525  19399 MULTIPOLYGON (((-81.47276 3...
## 6    5829  12065 MULTIPOLYGON (((-81.94135 3...
## 7   16127  32306 MULTIPOLYGON (((-77.10377 3...
## 8    6610  12678 MULTIPOLYGON (((-76.78307 3...
## 9   11227  21713 MULTIPOLYGON (((-78.2615 34...
## 10  55644 116574 MULTIPOLYGON (((-78.65572 3...
```

Question: What differences do you observe when comparing a typical tidy data frame to the new simple feature object?

There are several differences. First of all, this simple feature object is not a tibble, meaning it does not give us dimensions in the same way (though it does give them in a different way) and it does not list variable types (eg `<chr>`, `<int>`, etc). It doesn't list how many more rows there are beyond what it displays, though again it has told us how many rows there are total. It includes additional information, such as the geometry type, dimension, bounding box, and coordinate reference system, and each feature contains its own geometry (which was not necessary for standard non-spatial data).

Simple features

A **simple feature** is a standard, formal way to describe how real-world spatial objects (country, building, tree, road, etc) can be represented by a computer.

The package `sf` implements simple features and other spatial functionality using **tidy** principles. Simple features have a geometry type. Common choices are shown in the slides associated with today's lecture.

Simple features are stored in a data frame, with the geographic information in a column called `geometry`. Simple features can contain both spatial and non-spatial data.

All functions in the `sf` package helpfully begin `st_`.

`sf` and `ggplot`

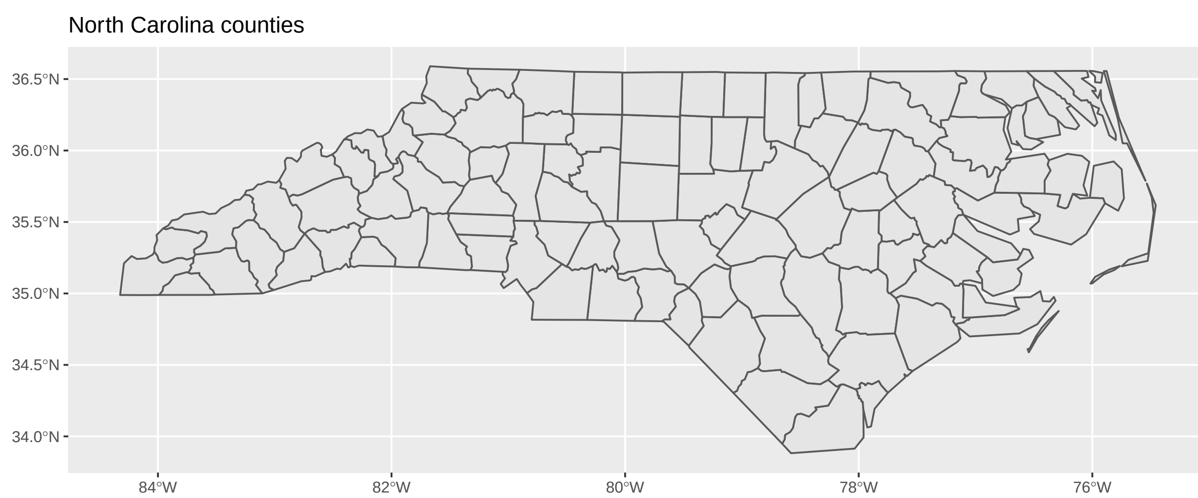
To read simple features from a file or database use the function `st_read()`.

```
nc <- st_read("nc_regvoters.shp", quiet = TRUE)
```

Notice `nc` contains both spatial and nonspatial information.

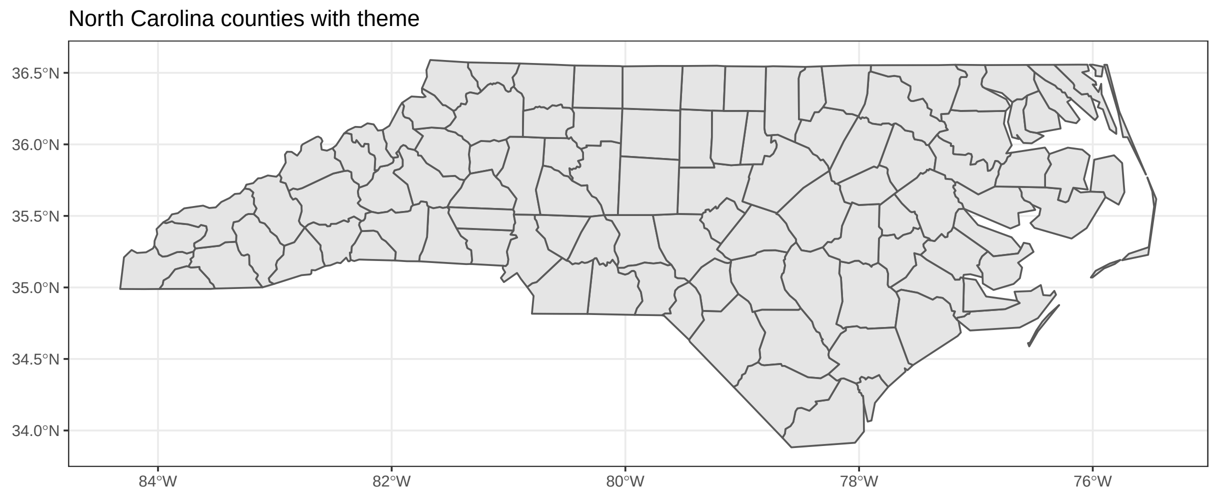
We can build up a visualization layer-by-layer beginning with `ggplot`. Let's start by making a basic plot of North Carolina counties.

```
ggplot(nc) +
  geom_sf() +
  labs(title = "North Carolina counties")
```



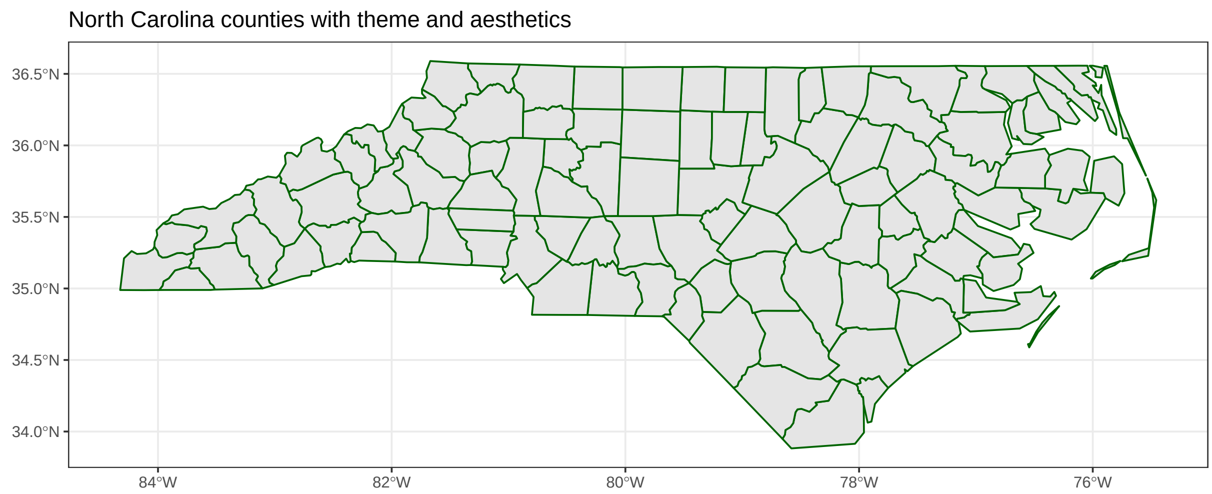
Now adjust the theme with `theme_bw()`.

```
ggplot(nc) +
  geom_sf() +
  labs(title = "North Carolina counties with theme") +
  theme_bw()
```



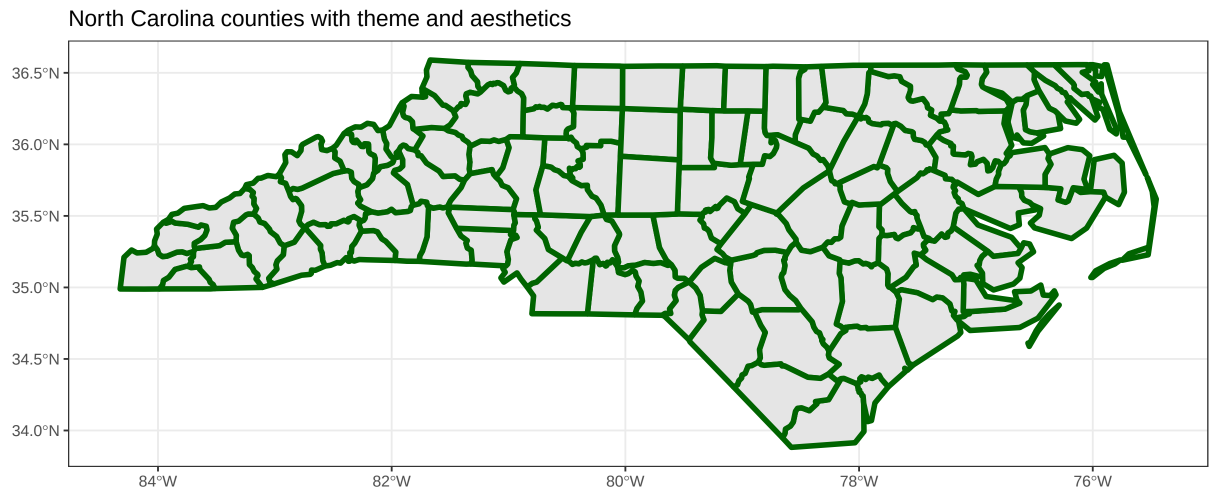
Now adjust `color` in `geom_sf` to change the color of the county borders.

```
ggplot(nc) +
  geom_sf(color = "darkgreen") +
  labs(title = "North Carolina counties with theme and aesthetics") +
  theme_bw()
```



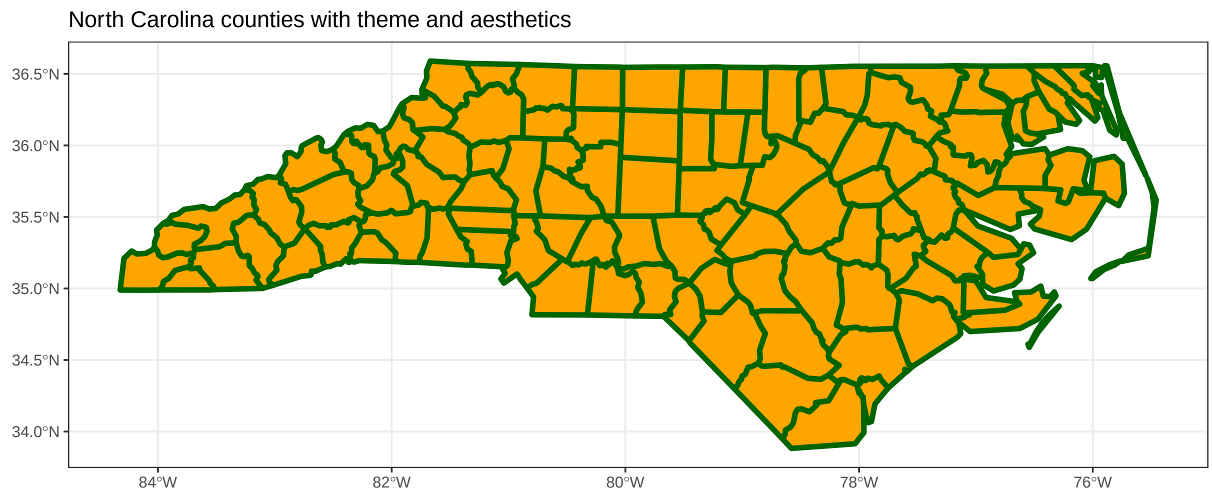
Then increase the width of the county borders using `size`.

```
ggplot(nc) +
  geom_sf(color = "darkgreen", size = 1.5) +
  labs(title = "North Carolina counties with theme and aesthetics") +
  theme_bw()
```



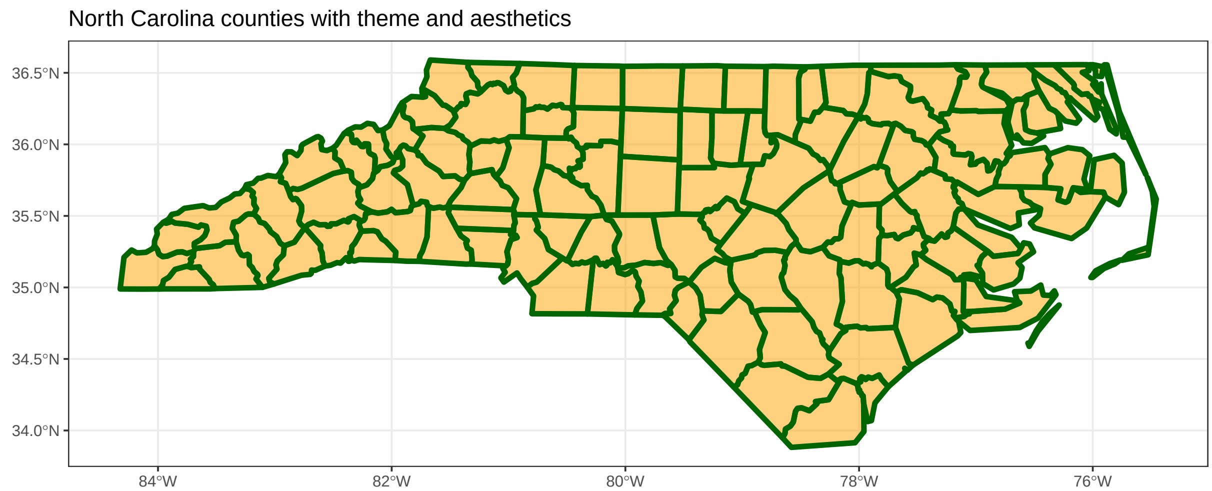
Fill the counties by specifying a `fill` argument.

```
ggplot(nc) +
  geom_sf(color = "darkgreen", size = 1.5, fill = "orange") +
  labs(title = "North Carolina counties with theme and aesthetics") +
  theme_bw()
```



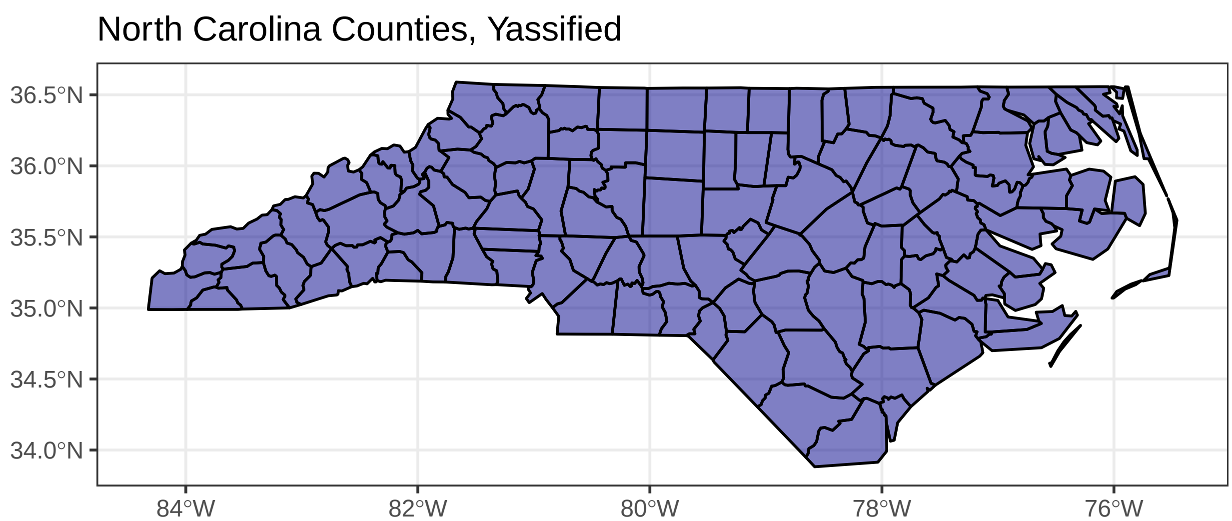
Finally, adjust the transparency using `alpha`.

```
ggplot(nc) +
  geom_sf(color = "darkgreen", size = 1.5, fill = "orange", alpha = 0.50) +
  labs(title = "North Carolina counties with theme and aesthetics") +
  theme_bw()
```



Our current map is a bit much. Adjust `color`, `size`, `fill`, and `alpha` until you have a map that effectively displays the counties of North Carolina.

```
ggplot(nc) +
  geom_sf(color = "black", size = 0.5, fill = "darkblue", alpha = 0.5) +
  labs(title = "North Carolina Counties, Yassified") +
  theme_bw()
```



North Carolina Registered Voters

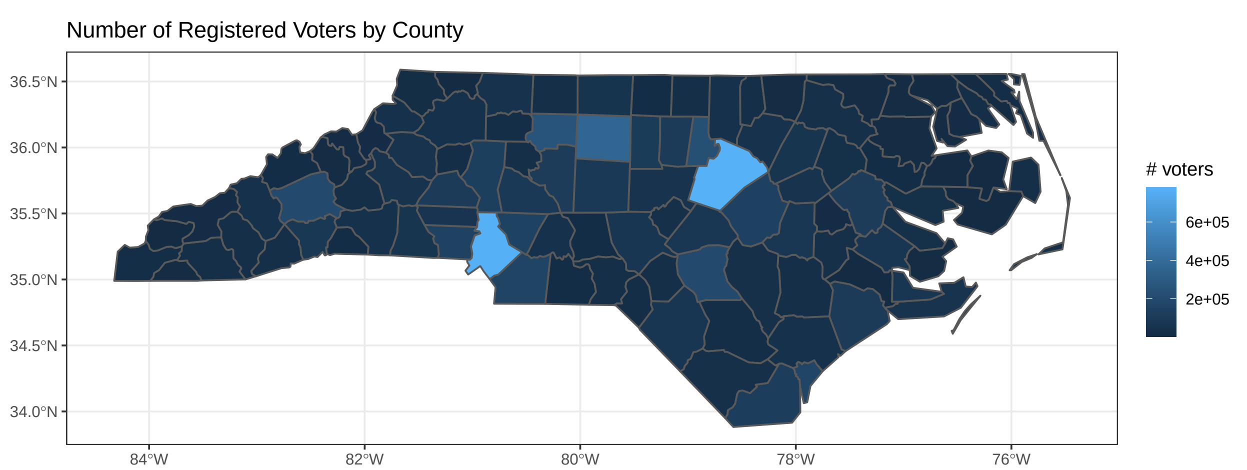
The `nc` data was obtained from the NC Board of Elections website and contains statistics on NC registered voters as of September 4, 2021.

The dataset contains the following variables on all North Carolina counties, categories provided by the NCSBE:

- `county`: county name
- `dem`: total number of voters who are registered Democrats
- `gop`: total number of voters who are registered Republicans
- `lib`: total number of voters who are registered Libertarians
- `unaf`: total number of voters who are unaffiliated
- `white`: total number of voters who are white
- `black`: total number of voters who are Black
- `ntv_a`: total number of voters who are Native American
- `ntv_h`: total number of voters who are Native Hawaiian
- `other`: total number of voters who are classified as "other" for race
- `hispanic`: total number of voters who are Hispanic
- `male`: total number of voters who identify as male
- `female`: total number of voters who identify as female
  - Please note- these are the only options given by the NCBSE, but male + female do not add up to total since some voters either decide not to disclose or have a different gender identity than these options.
- `total`: total number of registered voters in that county
- `geometry`: geographic coordinates of the county

Let's use the NCBSE data to generate a **choropleth map** of the number of registered voters by county.

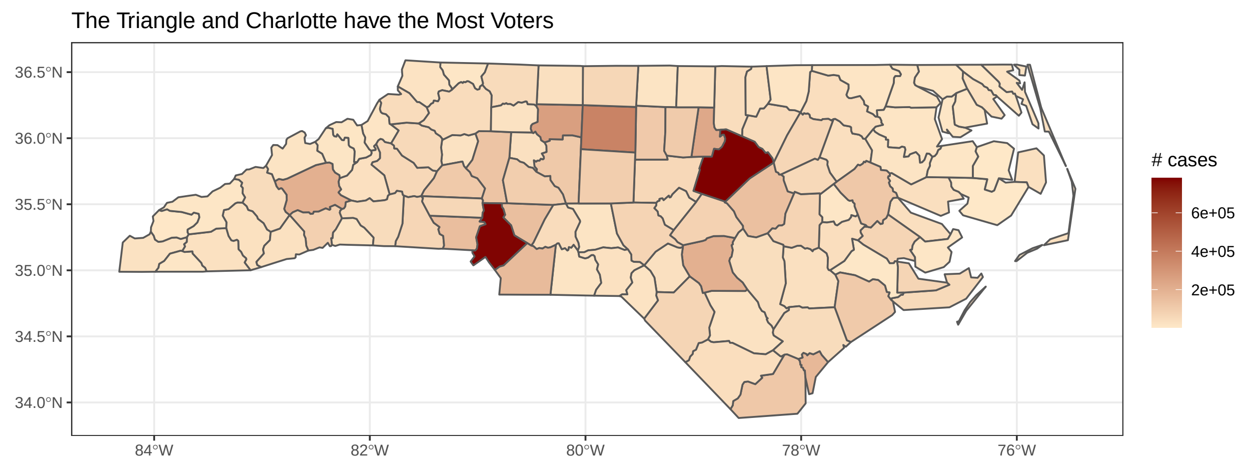
```
ggplot(nc) +
  geom_sf(aes(fill = total)) +
  labs(title = "Number of Registered Voters by County",
       fill = "# voters") +
  theme_bw()
```



It is sometimes helpful to pick diverging colors, colorbrewer2 can help.

One way to set fill colors is with `scale_fill_gradient()`.

```r
ggplot(nc) +
  geom_sf(aes(fill = total)) +
  scale_fill_gradient(low = "#fee8c8", high = "#7f0000") +
  labs(title = "The Triangle and Charlotte have the Most Voters",
       fill = "# cases") +
  theme_bw()
```



Challenges

1. Different types of data exist (raster and vector).
2. The coordinate reference system (CRS) matters.
3. Manipulating spatial data objects is similar, but not identical to manipulating data frames.

`dplyr`

The `sf` package plays nicely with our earlier data wrangling functions from `dplyr`.

`select()`

Maybe you are interested in the partisan breakdown of a county.

```r
nc %>%
  select(county, dem, gop, total)
```

```
## Simple feature collection with 100 features and 4 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -84.32385 ymin: 33.88199 xmax: -75.45698 ymax: 36.58965
## Geodetic CRS:  NAD27
## First 10 features:
##       county   dem   gop  total                       geometry
## 1   ALAMANCE 38209 35967 110042 MULTIPOLYGON (((-79.24619 3...
```

```
## 2  ALEXANDER  4772 11750  24612 MULTIPOLYGON (((-81.10889 3...
## 3  ALLEGHANY  2030  3005   7534 MULTIPOLYGON (((-81.23989 3...
## 4      ANSON  9130  2858  15625 MULTIPOLYGON (((-79.91995 3...
## 5       ASHE  4261  8804  19399 MULTIPOLYGON (((-81.47276 3...
## 6      AVERY  1343  6994  12065 MULTIPOLYGON (((-81.94135 3...
## 7   BEAUFORT 10883 11873  32306 MULTIPOLYGON (((-77.10377 3...
## 8     BERTIE  8178  1629  12678 MULTIPOLYGON (((-76.78307 3...
## 9     BLADEN  9847  5005  21713 MULTIPOLYGON (((-78.2615 34...
## 10 BRUNSWICK 26797 46557 116574 MULTIPOLYGON (((-78.65572 3...
```

`mutate()`

Maybe you are interested in the percentage of registered Democrats in a county.

```
nc %>%
  mutate(pct_dem = dem/total)
```

```
## Simple feature collection with 100 features and 15 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -84.32385 ymin: 33.88199 xmax: -75.45698 ymax: 36.58965
## Geodetic CRS:  NAD27
## First 10 features:
##       county   dem   gop lib  unaf white black ntv_a ntv_h other hispanic  male
## 1   ALAMANCE 38209 35967 670 35196 70330 21377   259     8 18068     4658 44651
## 2  ALEXANDER  4772 11750 123  7967 21103   921    33     2  2553      364 10947
## 3  ALLEGHANY  2030  3005  33  2466  6596    70     8     0   860      183  3319
## 4      ANSON  9130  2858  38  3599  6267  6198    25     0  3135       83  5800
## 5       ASHE  4261  8804 102  6232 17501   112    23     1  1762      257  8609
## 6      AVERY  1343  6994  55  3673 10714    44    20     0  1287       80  5283
## 7   BEAUFORT 10883 11873 124  9426 22052  6961    35     1  3257      463 13591
## 8     BERTIE  8178  1629  36  2835  4468  7283    19     1   907       38  5310
## 9     BLADEN  9847  5005  77  6784 12113  7412   374     2  1812      444  9472
## 10 BRUNSWICK 26797 46557 618 42602 92487  8384   344     4 15355     1454 48199
##    female  total                       geometry   pct_dem
## 1   54529 110042 MULTIPOLYGON (((-79.24619 3... 0.3472220
## 2   11768  24612 MULTIPOLYGON (((-81.10889 3... 0.1938892
## 3    3548   7534 MULTIPOLYGON (((-81.23989 3... 0.2694452
## 4    6980  15625 MULTIPOLYGON (((-79.91995 3... 0.5843200
## 5    9525  19399 MULTIPOLYGON (((-81.47276 3... 0.2196505
## 6    5829  12065 MULTIPOLYGON (((-81.94135 3... 0.1113137
## 7   16127  32306 MULTIPOLYGON (((-77.10377 3... 0.3368724
## 8    6610  12678 MULTIPOLYGON (((-76.78307 3... 0.6450544
## 9   11227  21713 MULTIPOLYGON (((-78.2615 34... 0.4535071
## 10  55644 116574 MULTIPOLYGON (((-78.65572 3... 0.2298712
```

`filter()`

You could filter for the percentage of Dems being over 50% (a majority).

```
nc %>%
  mutate(pct_dem = dem/total) %>%
  filter(pct_dem > 0.5)
```

```
## Simple feature collection with 12 features and 15 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -80.32528 ymin: 34.30457 xmax: -76.35819 ymax: 36.55629
## Geodetic CRS:  NAD27
## First 10 features:
##         county    dem   gop  lib  unaf  white black ntv_a ntv_h other hispanic
## 1        ANSON   9130  2858   38  3599   6267  6198    25     0  3135       83
## 2       BERTIE   8178  1629   36  2835   4468  7283    19     1   907       38
## 3       DURHAM 124870 24486 1250 78361 104673 72667   529    22 51076     9373
## 4    EDGECOMBE  21636  6398   96  5668  11776 19123    63     1  2835      443
## 5      HALIFAX  21686  4925  101  9335  13634 18419  1063     0  2931      247
## 6     HERTFORD   9627  1533   35  3113   4509  8673    93     2  1031       92
## 7       MARTIN   8446  3490   40  4001   8411  6466    27     0  1073      140
## 8  NORTHAMPTON   8480  1528   40  3091   5064  7208    30     0   837       47
## 9      ROBESON  37068 11953  197 20567  19741 18206 24034     8  7796     1718
## 10       VANCE  17256  4814   93  6249  11135 14052    55     1  3169      604
##     male female  total                       geometry   pct_dem
## 1   5800   6980  15625 MULTIPOLYGON (((-79.91995 3... 0.5843200
## 2   5310   6610  12678 MULTIPOLYGON (((-76.78307 3... 0.6450544
## 3  88337 112200 228967 MULTIPOLYGON (((-79.01814 3... 0.5453624
## 4  13708  18124  33798 MULTIPOLYGON (((-77.67122 3... 0.6401562
## 5  15178  18796  36047 MULTIPOLYGON (((-77.33221 3... 0.6016035
## 6   5946   7690  14308 MULTIPOLYGON (((-76.74506 3... 0.6728404
## 7   6813   8457  15977 MULTIPOLYGON (((-77.17846 3... 0.5286349
## 8   5637   6729  13139 MULTIPOLYGON (((-77.21767 3... 0.6454068
## 9  29767  37329  69785 MULTIPOLYGON (((-78.86451 3... 0.5311743
## 10 11557  14778  28412 MULTIPOLYGON (((-78.49252 3... 0.6073490
```

`summarize()`

We can also calculate summary statistics for our new variable.

```
nc %>%
  mutate(pct_dem = dem/total) %>%
  summarize(mean_pct_dem = mean(pct_dem),
            min_pct_dem = min(pct_dem),
            max_pct_dem = max(pct_dem))
```

```
## Simple feature collection with 1 feature and 3 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -84.32385 ymin: 33.88199 xmax: -75.45698 ymax: 36.58965
## Geodetic CRS:  NAD27
##   mean_pct_dem min_pct_dem max_pct_dem                       geometry
## 1    0.3428258   0.1008724   0.6728404 MULTIPOLYGON (((-76.46926 3...
```

Geometries are "sticky". They are kept until deliberately dropped using `st_drop_geometry`.

```
nc %>%
  select(county, total) %>%
  st_drop_geometry()
```

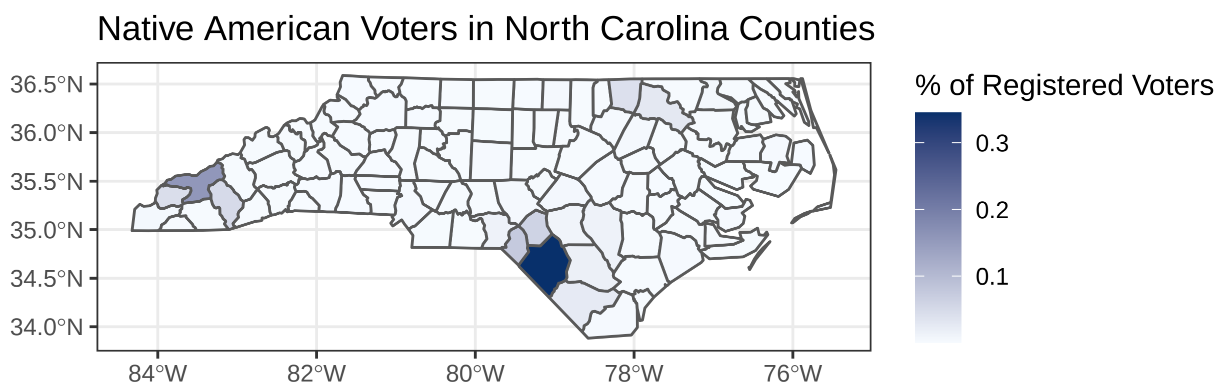
```
##           county  total
## 1       ALAMANCE 110042
## 2      ALEXANDER  24612
## 3      ALLEGHANY   7534
## 4          ANSON  15625
## 5           ASHE  19399
## 6          AVERY  12065
## 7       BEAUFORT  32306
## 8         BERTIE  12678
## 9         BLADEN  21713
## 10     BRUNSWICK 116574
## 11      BUNCOMBE 201401
## 12         BURKE  57481
## 13      CABARRUS 148489
## 14      CALDWELL  53537
## 15        CAMDEN   7646
## 16      CARTERET  52097
## 17       CASWELL  15195
## 18       CATAWBA 107060
## 19       CHATHAM  57602
## 20      CHEROKEE  22010
## 21        CHOWAN   9685
## 22          CLAY   9129
## 23     CLEVELAND  66186
## 24      COLUMBUS  35646
## 25        CRAVEN  68989
## 26    CUMBERLAND 201336
## 27     CURRITUCK  21189
## 28          DARE  30151
## 29      DAVIDSON 111819
## 30         DAVIE  31265
## 31        DUPLIN  30586
## 32        DURHAM 228967
## 33     EDGECOMBE  33798
## 34       FORSYTH 263103
## 35      FRANKLIN  47475
## 36        GASTON 150351
## 37         GATES   8050
## 38        GRAHAM   5944
## 39     GRANVILLE  39468
## 40        GREENE  10565
## 41      GUILFORD 366867
## 42       HALIFAX  36047
## 43       HARNETT  79170
## 44       HAYWOOD  45241
## 45     HENDERSON  85808
## 46      HERTFORD  14308
## 47          HOKE  32002
## 48          HYDE   3003
```

```
## 49        IREDELL 129972
## 50        JACKSON  28551
## 51       JOHNSTON 144074
## 52          JONES   6826
## 53            LEE  37792
## 54         LENOIR  35854
## 55        LINCOLN  63412
## 56          MACON  26868
## 57        MADISON  16636
## 58         MARTIN  15977
## 59       MCDOWELL  29049
## 60    MECKLENBURG 773683
## 61       MITCHELL  11004
## 62     MONTGOMERY  16821
## 63          MOORE  72611
## 64           NASH  66185
## 65    NEW HANOVER 172138
## 66    NORTHAMPTON  13139
## 67         ONSLOW 107577
## 68         ORANGE 105638
## 69        PAMLICO   9157
## 70     PASQUOTANK  27127
## 71         PENDER  45024
## 72     PERQUIMANS   9813
## 73         PERSON  27017
## 74           PITT 113718
## 75           POLK  15772
## 76       RANDOLPH  93805
## 77       RICHMOND  27216
## 78        ROBESON  69785
## 79     ROCKINGHAM  60497
## 80          ROWAN  95376
## 81     RUTHERFORD  45278
## 82        SAMPSON  37263
## 83       SCOTLAND  20153
## 84         STANLY  42752
## 85         STOKES  31547
## 86          SURRY  46850
## 87          SWAIN   9774
## 88   TRANSYLVANIA  25854
## 89        TYRRELL   2268
## 90          UNION 161006
## 91          VANCE  28412
## 92           WAKE 780519
## 93         WARREN  12940
## 94     WASHINGTON   8050
## 95        WATAUGA  43127
## 96          WAYNE  73786
## 97         WILKES  43527
## 98         WILSON  54424
## 99         YADKIN  24494
## 100        YANCEY  14197
```

Practice

(1) Construct an effective visualization investigating the percentage of all voters in NC that are Native American. Use #f7fbff as "low" on the color gradient and #08306b as "high". Which county has the highest percentage of Native American voters? (You might want to use Google here.)

```
nc %>%
  mutate(pct_native = ntv_a / total) %>%
  ggplot(.) +
  geom_sf(aes(fill = pct_native)) +
  scale_fill_gradient(low = "#f7fbff", high = "#08306b") +
  labs(title = "Native American Voters in North Carolina Counties",
       fill = "% of Registered Voters") +
  theme_bw()
```



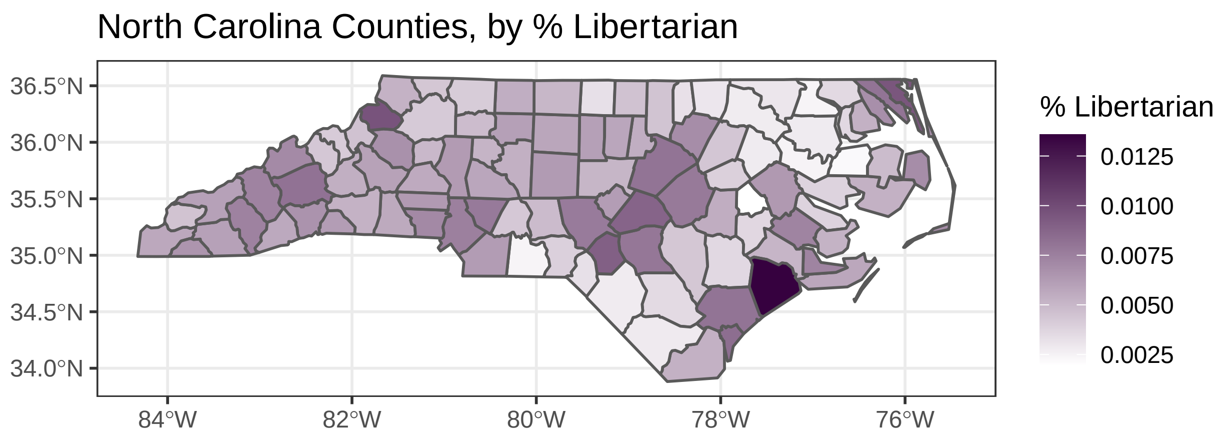
Robeson County, North Carolina, has the highest percentage of Native American registered voters in North Carolina.

(2) Write a brief research question that you could answer with this dataset and then investigate it here.

Where in the state do you tend to find relatively more Libertarians?

```
nc %>%
  mutate(pct_lib = lib / total) %>%
  ggplot(.) +
```

```
geom_sf(aes(fill = pct_lib)) +
scale_fill_gradient(low = "white", high = "#36013F") +
labs(title = "North Carolina Counties, by % Libertarian",
     fill = "% Libertarian") +
theme_bw()
```



(3) What are limitations of your visualizations above?

They can only display a single variable, as counties cannot with these methods be shaded by two different colors. If we were to apply features like lines and patterns, we would solve that issue but the graph would become difficult to look at. Alongside being low data, they have very little ability for comparison, as neither one is faceted to display other variabls and I am unsure whether that can even be done (certainly, it would be difficult to do so for data structured in this fashion).

Additional Resources

- Simple features in `R`
- Coordinate references systems
- Geographic data in `R`