Spatial Analysis When our data contains geographic coordinates, it becomes *spatial data*. Spatial data often requires different analytic techniques. An introduction to some important ideas in spatial analysis follows.

- An example from earlier in the course, contextualized by spatial analysis:
  - Snow's cholera map: use modern capabilities to help quantify visual observations
- Understanding spatial data: spatial data contains instructions about how to draw points, lines, and polygons. This data can become quite large, and is commonly stored in a *shapefile*.
- Common misinterpretations: spatial data is commonly misinterpreted or misunderstood. You should be aware of issues like the *ecological fallacy*, in which conclusions drawn from aggregate data are erroneously applied to individual subjects.
- Projections: We live in three dimensions, but we typically want to display geographic data on a two-dimensional plot. Thus, some *projection* system is required. There is no one best projection system – each has advantages and disadvantages. Generally, the best types of projections preserve *shape* or preserve *area*. You should be aware that many "default" projections can be very misleading (e.g. Mercator). The choice of how to project your data can have a direct influence on what viewers will take away from your data maps.
- Normalization: On choropleth maps, we almost always want to show some sort of density or ratio rather than raw values.
- Scales: Color scales can be linear, logarithmic, categorical, etc.
- Color schemes: Cynthia Brewer maintains lists of "smart" color palettes. Categorial variables should be displayed using a *qualitative* palette, while quantitative variables should be displayed using a *sequential* or *diverging* palette.
- Map types:
  - Choropleth: color or shade regions based on the value of a variable
  - Proportional symbol: Associate a symbol with each location, but scale its size to reflect the value of a variable
  - Dot density (bubble map): place dots for each data point, and view their accumulation
- ArcGIS is the state-of-the-art Geographic Information System software. It is avaialable through the VPN.

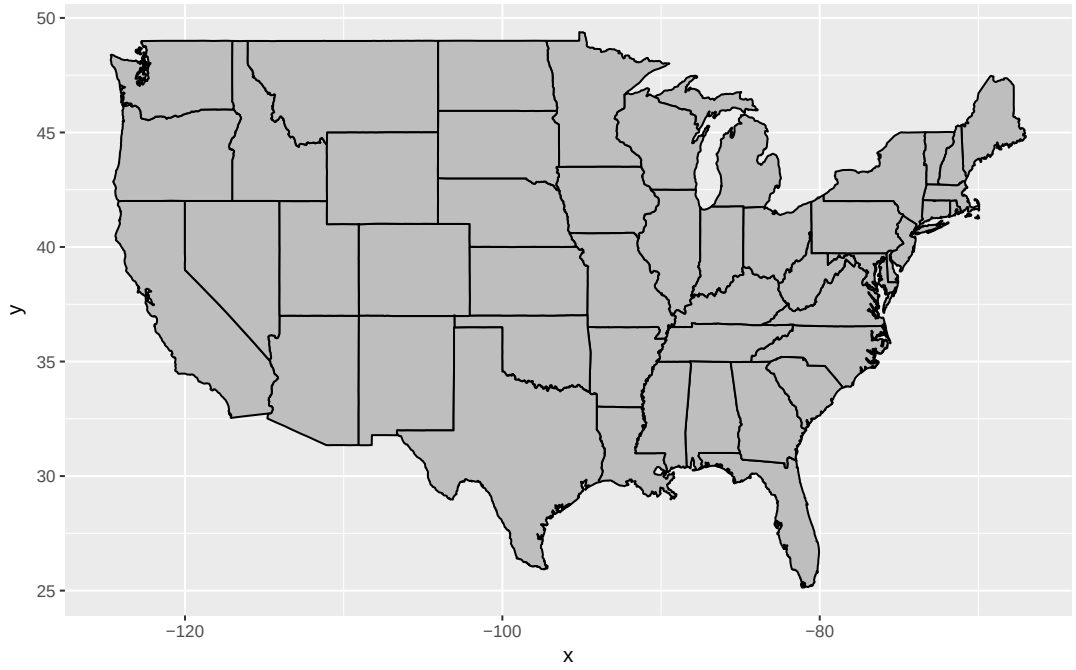
Mapping Data Like many tasks in R, creating data maps can be done in a variety of ways. Here, we summarize a few important concepts and useful packages.

- Shapefiles: Mapping data is stored in *shapefiles*. Shapefiles contain vector-based instructions for drawing geographic objects, including the boundaries of countries, counties, and towns, etc. Shapefiles for a specific purpose can often be downloaded online in a ZIP file, and can be quite large depending on the level of detail offered.

Mapping with ggplot

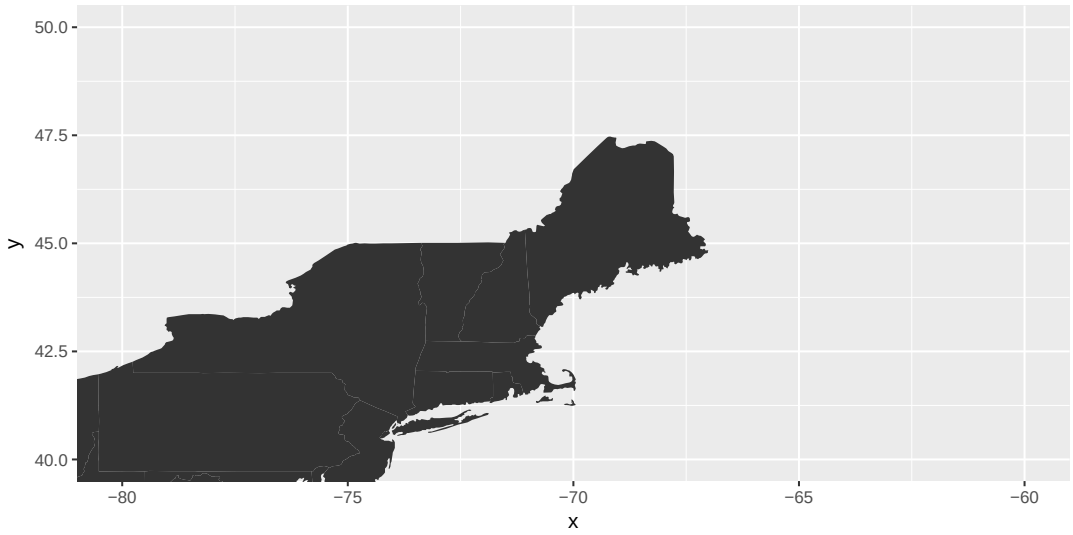
- We begin by looking at a basic shape file. Let's start with the contiguous United States.

```
require(ggplot2)
require(maps)
require(mosaic)

all_states <- map_data("state")

ggplot()+
  geom_map(data=all_states, aes(map_id=region), map=all_states,
           fill="grey", color="black")+
  expand_limits(x=all_states$long, y=all_states$lat)
```



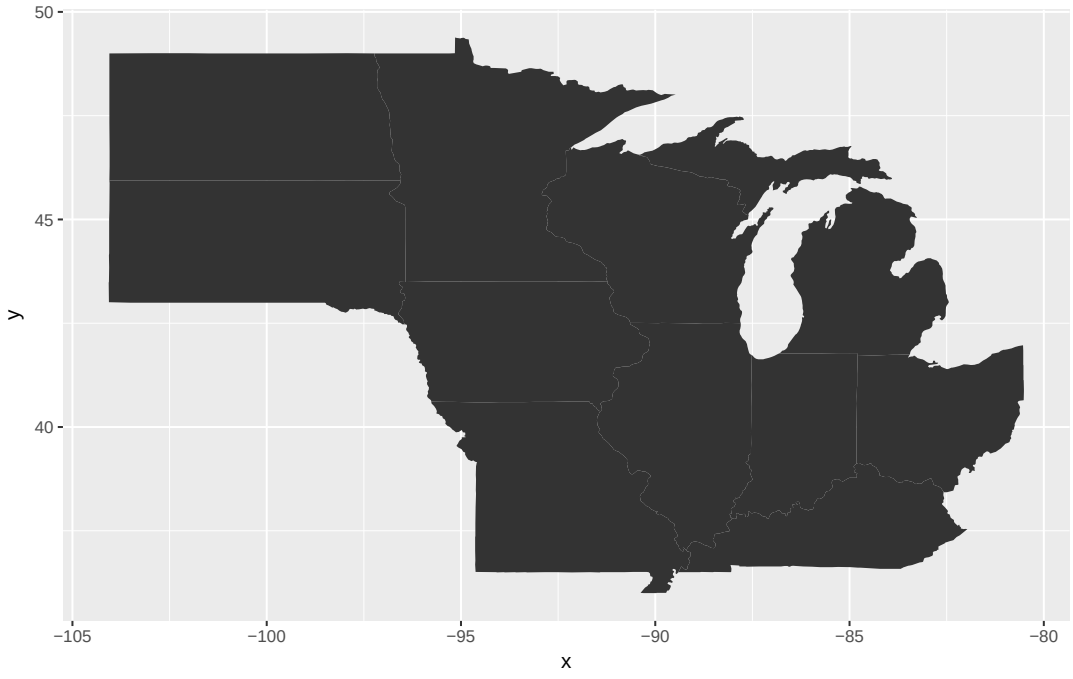
- We can zero on part of the map be selecting ranges of latitude and longitude that we are of interest. The size and shape look warped. Therefore, you should only change the limits if you are keeping the scale consistent

```
ggplot(data=all_states)+
  geom_map(map=all_states, aes(map_id=region))+
  expand_limits(x=c(-80,-60), y=c(40,50))
```



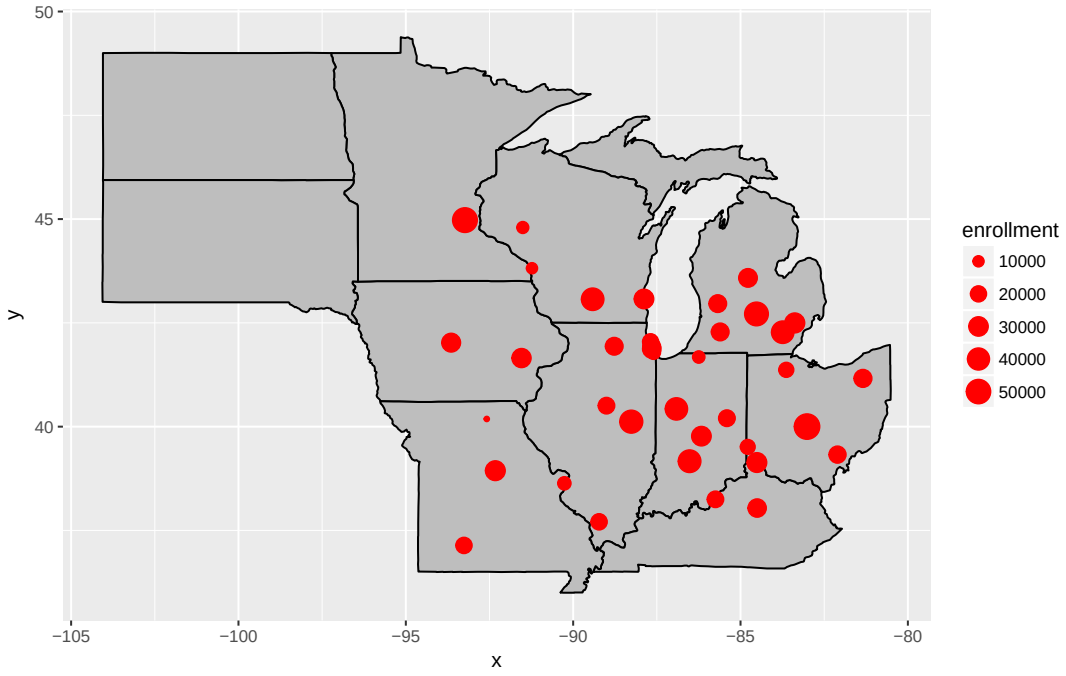
- Now, let's turn our focus to a selection of states that make up the midwest.

```
states_midwest <- filter(all_states, region %in%
                           c( "illinois", "indiana", "iowa", "kentucky",
                              "michigan", "minnesota","missouri",
                              "north dakota", "ohio", "south dakota",
                              "wisconsin" ) )
ggplot()+
  geom_map(data=states_midwest, aes(map_id=region), map=states_midwest)+
  expand_limits(x=states_midwest$long, y=states_midwest$lat)
```



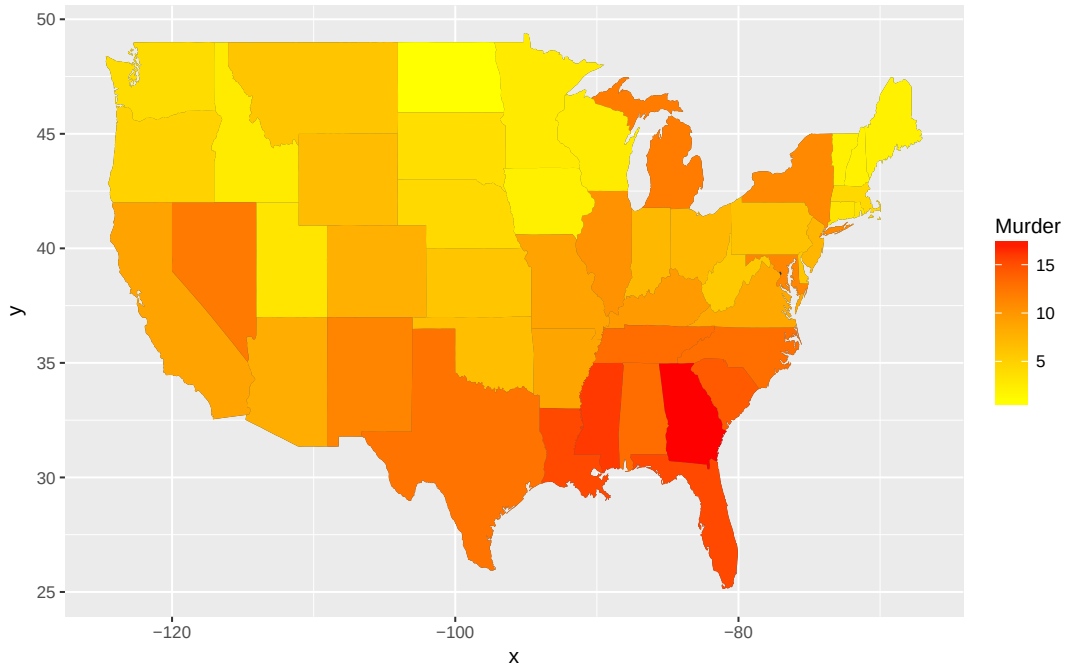
- I have a data set with various colleges in these states. Notice some key features about the data set. Suppose I want to visualize locations where student enrollments are highest.

```
college_enrollment <- read.csv("~/Desktop/Data/college_enrollment.csv")

ggplot()+
  geom_map(data=states_midwest, aes(map_id=region),
           map=states_midwest, fill="grey", color="black")+
  expand_limits(x=states_midwest$long, y=states_midwest$lat)+
  geom_point(data=college_enrollment, aes(x=long, y=lat, size = enrollment), color="red")
```



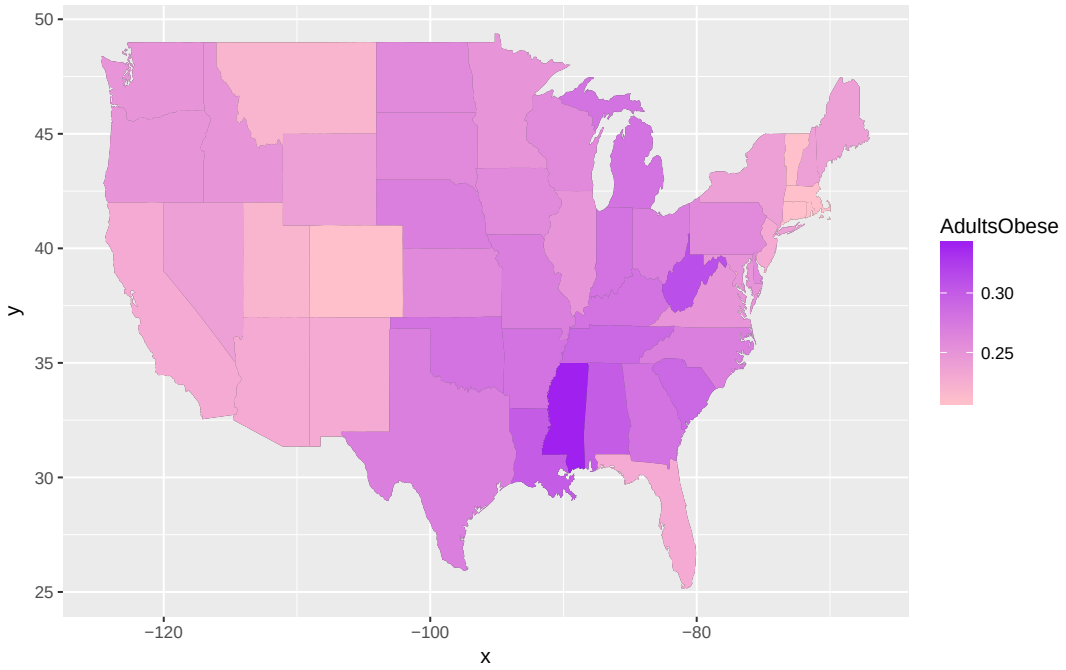
- Now suppose I have state-level data. USArrests is pre-loaded data in R. First let's take a look. Since the states are row names, we need to add them as a column in our data set. The `tolower()` function will turn the state names to all lower-case letters. Why does this matter?

```
crimes<-mutate(USArrests, region=tolower(rownames(USArrests)))

ggplot()+
  geom_map(data=all_states, aes(map_id= region), map = all_states) +
  geom_map(data=crimes, aes(map_id= region, fill = Murder), map = all_states) +
  expand_limits(x = all_states$long, y = all_states$lat)+
  scale_fill_gradient(low="yellow", high="red")
```



- Let's examine obesity rates by state as well.

```
Obese <- read.csv("~/Desktop/Data/Obese.csv")

Obese=mutate(Obese, region=tolower(Obese$State))

ggplot()+
  geom_map(data=all_states, aes(map_id=region), map=all_states)+
  geom_map(data=Obese, aes(map_id=region, fill = AdultsObese), map = all_states) +
  expand_limits(x = all_states$long, y = all_states$lat)+
  scale_fill_gradient(low="pink", high="purple")
```



- Your turn!

  1. For the crimes data set, take a look at other types of arrests. Are there regions with low murder arrests, but high arrests of another type?
  2. For the Obese data set, visualize childhood obesity proportions. Does that map look similar to the adult map?