Lecture 5: Maps

June 15, 2022

Overview

Course Administration

Good, Bad and Ugly

What and Why of Maps

Representing Maps Digitally

Maps in R

Course Administration

1. Didn't realize that summer classes end at 8:45. I'll stay till then.
2. You should have received your id number and feedback on your proposal
3. If I said come see me, come see me

Course Administration

1. Didn't realize that summer classes end at 8:45. I'll stay till then.
2. You should have received your id number and feedback on your proposal
3. If I said come see me, come see me
4. Please come to office hours! Sign up with scheduler
5. Two Monday holidays have Tuesday office hours

Course Administration

1. Didn't realize that summer classes end at 8:45. I'll stay till then.
2. You should have received your id number and feedback on your proposal
3. If I said come see me, come see me
4. Please come to office hours! Sign up with scheduler
5. Two Monday holidays have Tuesday office hours
6. Comments next week on your charts
7. This week's tutorial is in html, not pdf – let me know if you hate it

Course Administration

1. Didn't realize that summer classes end at 8:45. I'll stay till then.
2. You should have received your id number and feedback on your proposal
3. If I said come see me, come see me
4. Please come to office hours! Sign up with scheduler
5. Two Monday holidays have Tuesday office hours
6. Comments next week on your charts
7. This week's tutorial is in html, not pdf – let me know if you hate it
8. June 29: come prepared to work on your policy brief storyline

Course Administration

1. Didn't realize that summer classes end at 8:45. I'll stay till then.
2. You should have received your id number and feedback on your proposal
3. If I said come see me, come see me
4. Please come to office hours! Sign up with scheduler
5. Two Monday holidays have Tuesday office hours
6. Comments next week on your charts
7. This week's tutorial is in html, not pdf – let me know if you hate it
8. June 29: come prepared to work on your policy brief storyline
9. Questions? Concerns?

Next Week's Assignment

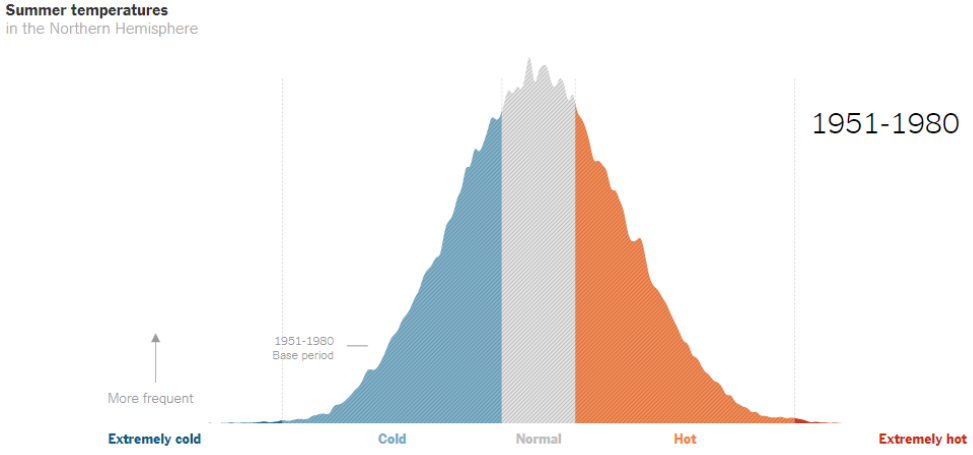
Find a descriptive or choropleth map. Post link to google sheet by Friday noon.

Finder	Commenter
David G	Dylan
Danielle G	Christina
Richa	Brady
Brady	Evan L.

This Week's Good Bad and Ugly

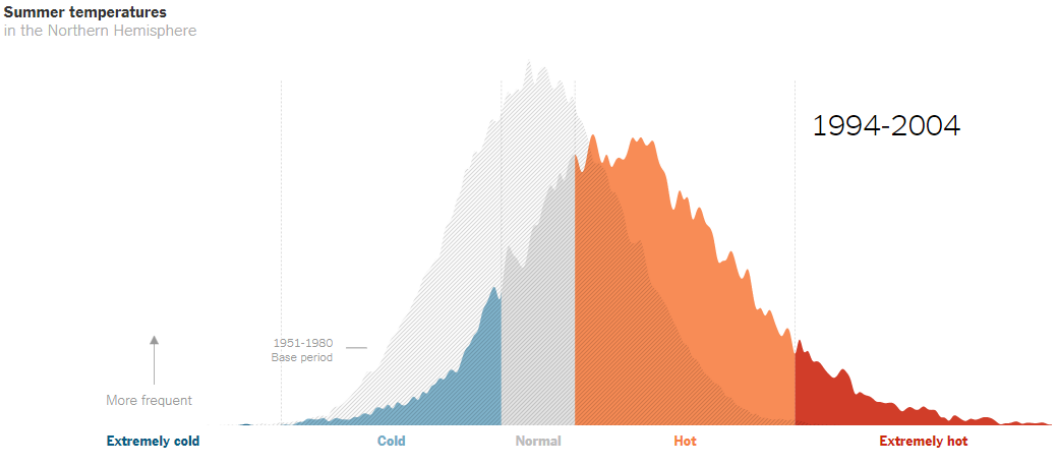
Finder	Commenter
Morre	Timberley
Spencer	Danielle
Evan L.	Mary
Dustin	Nicole

Timberley: Morre's Overlapping Histograms



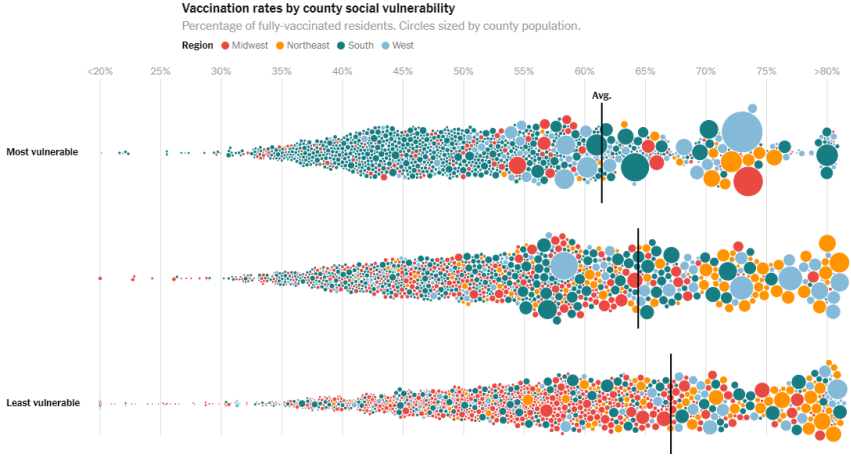
New York Times. Headline is "It's not your imagination. Summers are getting hotter."

Timberley: Morre's Overlapping Histograms



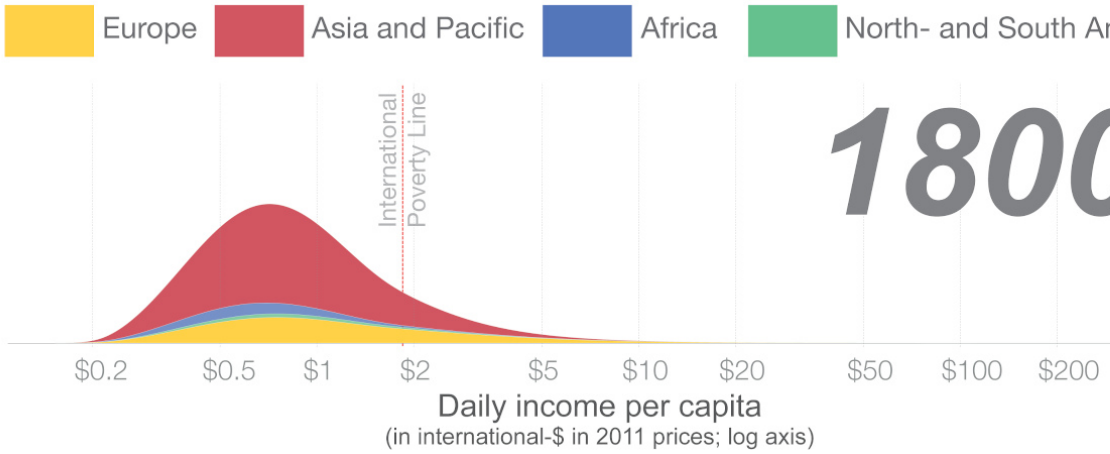
New York Times. Headline is "It's not your imagination. Summers are getting hotter."

Danielle: Spencer's *New York Times* Vaccination by Vulnerability



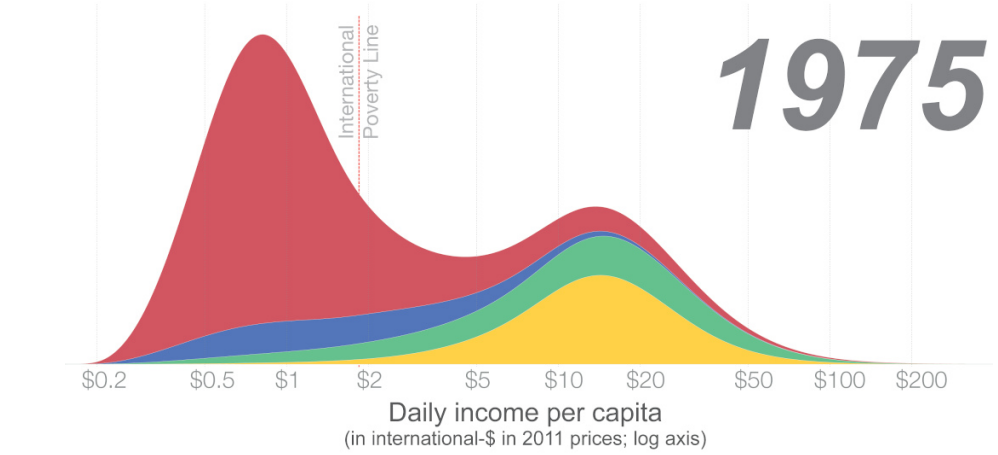
"See How Vaccinations Are Going...," *NYT*, updated June 14, 2022.

Mary: Evan's Global Income Over Time



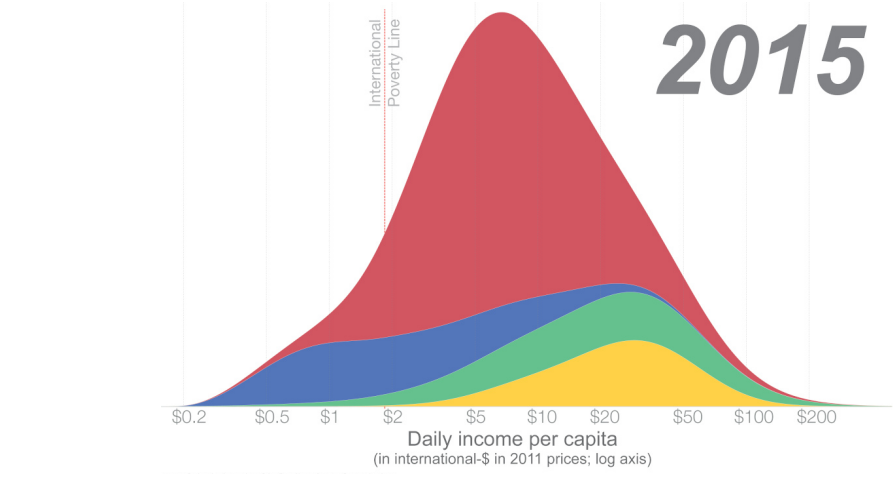
Neufeld, Dorothy, "Visualizing Global Income Distribution Over 200 Years," *Visual Capitalist*, May 19, 2020.

Mary: Evan's Global Income Over Time



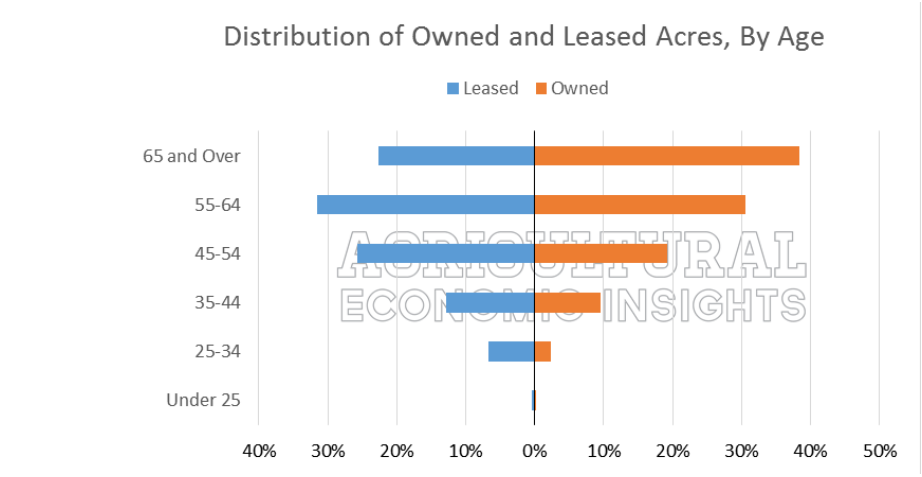
Neufeld, Dorothy, "Visualizing Global Income Distribution Over 200 Years," *Visual Capitalist*, May 19, 2022.

Mary: Evan's Global Income Over Time



Neufeld, Dorothy, "Visualizing Global Income Distribution Over 200 Years," *Visual Capitalist*, May 19, 2022.

Nicole: Dustin's Farmers



Widmar, David, "Aging American Farmers and Their Farmland," *Agricultural Economic Insights*, February 3, 2015.

Lecture 5: Maps

Today

1. Maps in general
   - 1.1 What is a map?
   - 1.2 Why maps?
   - 1.3 What are the components of maps?
   - 1.4 When do maps deceive?
2. Digital maps
   - 2.1 What they are
   - 2.2 What they can do

What and Why of Maps

1. What is a Map?

- Something that tries to describe two-dimensional space
- "scale model of reality" (Monmonier)
- "almost always smaller" than reality

2. Why Maps?

- Use a map if you want to locate something in two-dimensional geographic space
- Use a map when you want to show a **spatial** relationship
- Don't use a map if you want to compare geographic units

When is Space Important?

1. To show relationship between two geographic things. Examples?

When is Space Important?

1. To show relationship between two geographic things. Examples?
   - metro stops relative to average home prices
   - population density relative to the equator
2. To show a geographic pattern in an outcome. Examples?

When is Space Important?

1. To show relationship between two geographic things. Examples?
   - metro stops relative to average home prices
   - population density relative to the equator
2. To show a geographic pattern in an outcome. Examples?
   - voting outcomes correlated over space
   - geographic features that change smoothly and sharply over space

When is Space Important?

1. To show relationship between two geographic things. Examples?
   - metro stops relative to average home prices
   - population density relative to the equator
2. To show a geographic pattern in an outcome. Examples?
   - voting outcomes correlated over space
   - geographic features that change smoothly and sharply over space

Don't use a map if you can do something simpler!

3. What Do You Have to Decide to Make a Map?

In distilling reality, there are three key choices

1. scale
2. projection
3. symbolization

Projection

- We want to show both
  - equivalence: size proportional to physical size
  - conformality: shape proportional to true shape

Projection

- We want to show both
  - equivalence: size proportional to physical size
  - conformality: shape proportional to true shape
- But you cannot do both!

Projection

- We want to show both
  - equivalence: size proportional to physical size
  - conformality: shape proportional to true shape
- But you cannot do both!
- Put differently, you want to take something round and make something flat
- Because you cannot, you have to approximate
- Projection = choice of type of wrongness in going from round to flat

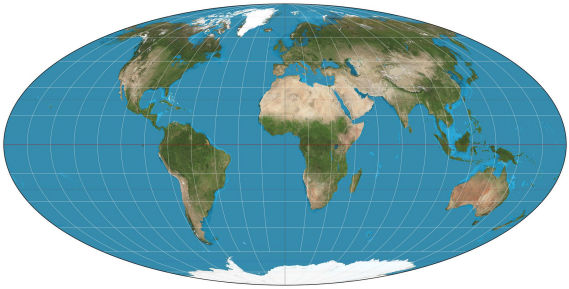
When Does Choice of Projection Matter?

- Matters for maps of the world
- Practically irrelevant for a map of DC
- For small areas, we care about precision of distance
- Frequently use a UTM (Universal Transverse Mercator) projection: units in meters

Rules of Thumb for Projections for Medium Areas

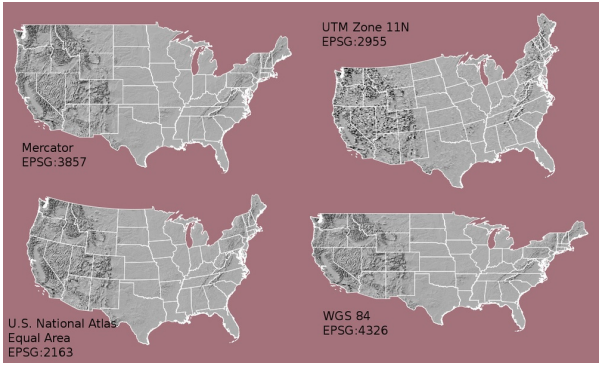
- Monmonier (p. 45) suggests for US either
  - Albers equal-area conic
  - Lambert conformal conic
- However, most maps you use should come with a projection defined

An Equal-Area Projection



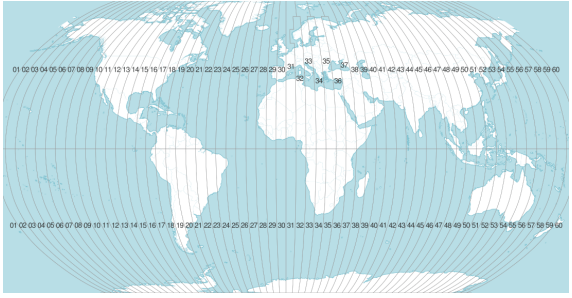
Thanks, Wikipedia.

The USA Four Ways



Thanks to Michael Corey.

UTM Zones

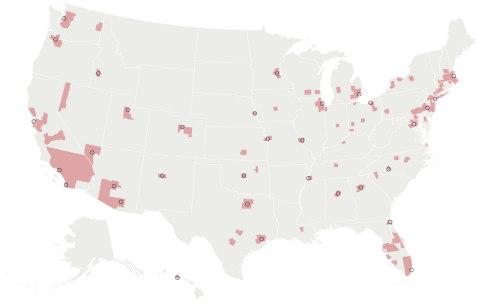


For small areas, use UTM projection if you need to calculate distances. Each number is a zone.
Thanks to Michael Corey.

4. Why Avoid Maps?

- They add complexity
- Geographic unit size infrequently related to importance
  - but remember that size indicates value
  - problematic!
- Examples?

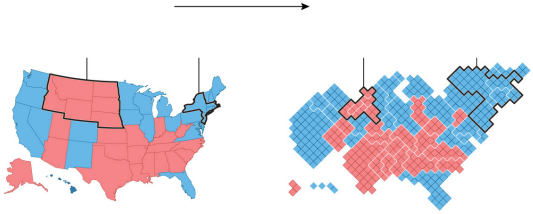
Red and Grey Areas Have About the Same Number of Votes Cast in 2012



With many thanks to the Washington Post

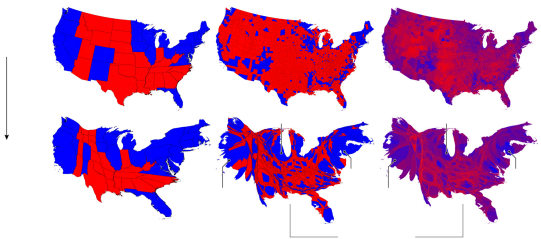
One Possible Solution

- A "cartogram" sizes locations by something: votes or people or electoral votes
- Five red midwestern states correspond to red block
- Mid-Atlantic corresponds to blue block

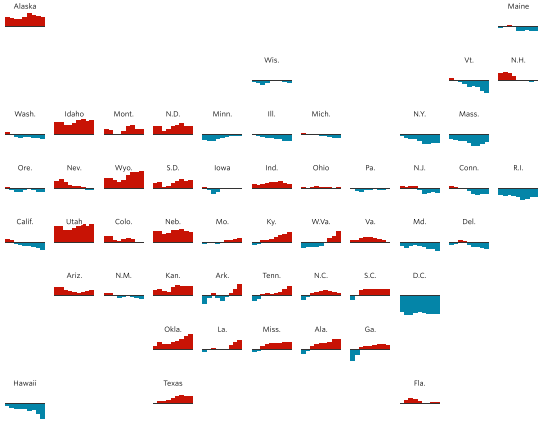


Another Possible Solution

- Thanks to U of Michigan physicist Newman
- Columns are state winner, county winner, county shaded by popular vote share
- Top is real map, bottom is cartogram
- Leftmost sized by electoral votes, others by votes cast



And a Quasi Map



Thanks to the Wall Street Journal, here.

Digital Maps

1. Digital Maps Have

- Units defined by coordinates in space
- Data for each unit

Examples of a map unit of observation, please!

Digital Maps

- A map is a representation of space
- A digital map is a file that tells a computer how to do this
- There are many formats, but we'll focus on shapefiles
- Shapefiles are a proprietary ArcInfo format, but can be read in R

Three Major Types of Shapes for Maps

1. points
2. lines
3. polygons

Points in Space

- location 1: (x, y)
- location 2: (x, y)
- location 3: (x, y)

What would you represent with points?

A Points Dataframe Example

LibID	X	Y	Name	Books
Ana	38.866	-76.980	Anacostia	500
CV	38.889	-76.932	Capitol View	501
Gtn	38.913	-77.068	Georgetown	499

Lines in Space

- location 1: $(x_1, y_1), (x_2, y_2)$
- location 2: $(x_1, y_1), (x_2, y_2)$
- location 3: $(x_1, y_1), (x_2, y_2)$

What would you represent with lines?

A Lines Dataframe Example

Int	X1	Y1	X2	Y2	Name	Condition
495	45	-62	26	-62	I495W	good
695	23	-50	25	-50	I695S	poor
10	15	-23	18	-24	I10	excellent

Polygons in Space

- location 1: $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_1, y_1)$
- location 2: $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5), (x_1, y_1)$
- location 3: $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_1, y_1)$

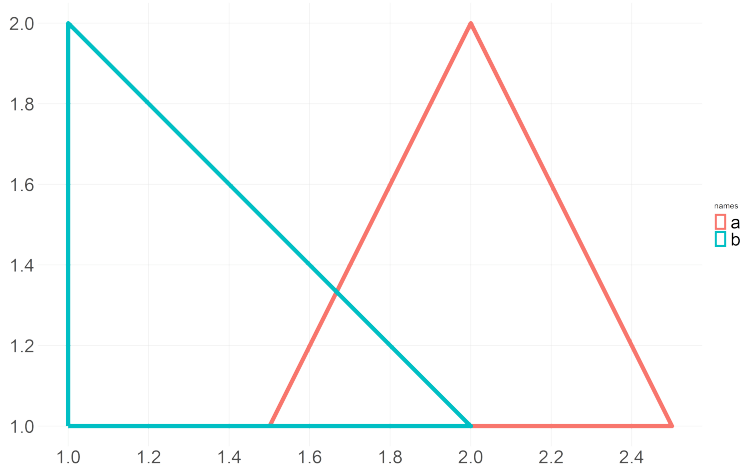
Note that last point is the same as the first point.[1]
What would you represent with polygons?

[1] Polygons can have holes; we can talk about this.

A Polygon Dataframe Example

Triangle	X1	Y1	X2	Y2	X3	Y3	X4	Y4
a	1.5	1	2	2	2.5	1	1.5	1
b	1	1	1	2	2	1	1	1

Plotting These



But Where Do the Points Go?

- A map file needs some instructions on what the points mean
- Map makers define coordinate systems so that everyone agrees on what $(x_1, y_1), (x_2, y_2)$ means
- Many maps have a geographic/global/spherical system: in latitude/longitude

But Where Do the Points Go?

- A map file needs some instructions on what the points mean
- Map makers define coordinate systems so that everyone agrees on what $(x_1, y_1), (x_2, y_2)$ means
- Many maps have a geographic/global/spherical system: in latitude/longitude
- And to lay flat, if we are not drawing on a globe
  - we need a projected coordinate system
  - have a defined unit of measurement: meters, feet, decimal degrees
  - usually tell you meters/feet/miles from a specific point

Implications for Mapping

- You can't put maps with two different coordinate systems on top of each other
- Easier to calculate distances and areas with projected coordinate systems
- You can go from one projection to another, but **use the right command**
- Digital maps usually come with a projection defined

R

Today

A. `sf` package

B. Reading

C. Plotting

D. Projections

E. Spatially combining

A. `sf` Package

- a new package as of 2018
- works with `tidyverse` and `ggplot`
- use all the other commands you've used to date
- ok for all map data except rasters
- thank you, Edzer Pebesma!

Install and Call `sf`

You'll need to install – once

```
install.packages("sf", dependencies = TRUE)
```

And call, at the top of your program

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

B.1. Reading a Digital Map

- R's `sf` can read many types of digital maps
- this class we will read shapefiles and geoJSON files
- these are different ways of codifying space into a file

B.1. Reading a Digital Map

- R's `sf` can read many types of digital maps
- this class we will read shapefiles and geoJSON files
- these are different ways of codifying space into a file
- we'll mostly use "shapefile" format
- a proprietary format from ESRI
- most downloads come in this format

B.1. Reading a Digital Map

- R's `sf` can read many types of digital maps
- this class we will read shapefiles and geoJSON files
- these are different ways of codifying space into a file
- we'll mostly use "shapefile" format
- a proprietary format from ESRI
- most downloads come in this format

Make sure you download the digital map file, not the data!

B.2. What is a Shapefile?

- shapefiles have 4 to 7 parts
- all have the same name and these extensions
  - .shp
  - .shx
  - .dbf
  - .prj
  - .xml
  - .cpg
- the first 3 are mandatory
- it's odd if you don't have a projection, but you can still draw a map

B.3. Read the shapefile – or any geographic file

The key command is `st_read("FILENAME.MAP_EXTENSION")`

```
shp.df <- st_read("c:/stuff/map.shp")
```

B.3. Read the shapefile – or any geographic file

The key command is `st_read("FILENAME.MAP_EXTENSION")`

```
shp.df <- st_read("c:/stuff/map.shp")
```

This new file

- works like a dataframe
- plus it has spatial information
- is called a "simple feature"

C.1 Plotting

Two main commands for plotting simple features in R

1. `plot()`
2. `ggplot()` using `geom_sf()`

Happily, `geom_sf()` works a lot like the other `geom_XXX()` commands you already know.

C.2. Example

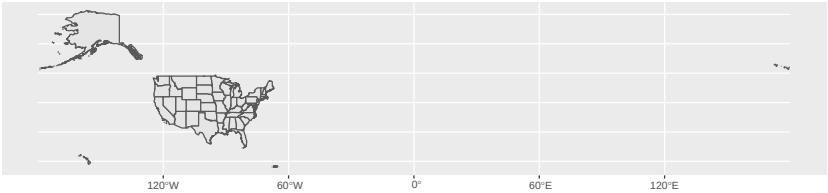
```
usmap <- st_read("H:/maps/united_states/census2010/states/gz_2010_us_040_0(
```

```
## Reading layer `gz_2010_us_040_00_20m' from data source
##   `H:\maps\united_states\census2010\states\gz_2010_us_040_00_20m.shp'
##   using driver `ESRI Shapefile'
## Simple feature collection with 52 features and 5 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -179.1473 ymin: 17.88481 xmax: 179.7785 ymax: 71.3
## Geodetic CRS:  NAD83
```

```
states <- ggplot() +
  geom_sf(data = usmap)
```

C.3. Example plot

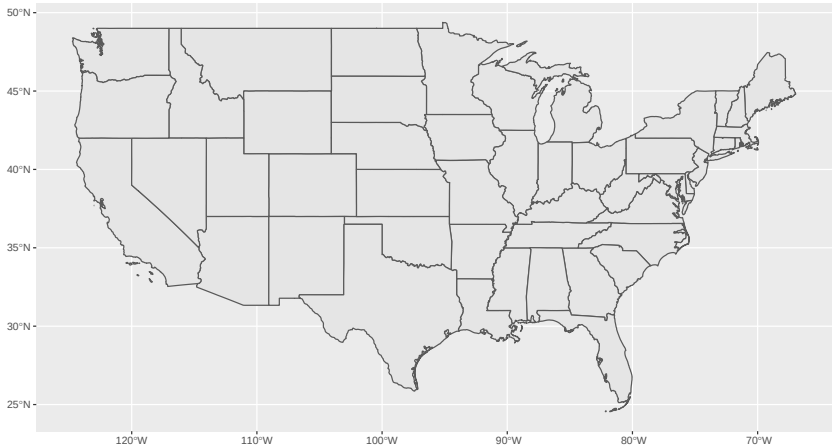
```
states
```



C.4. Just the Continental US

```
# omit AK, HI, PR
usmap.cont <- usmap[which(!(usmap$STATE %in% c("02","15","72"))),]
# or using filter
usmap.cont <- filter(.data = usmap, !(STATE %in% c("02","15","72")))
cont.us <-
  ggplot() + geom_sf(data = usmap.cont)
```

C.4. Just the Continental US



D. Projections

- maps should have a projection
- to tell R where to put points in space
- these are viewable

D. Projections

- maps should have a projection
- to tell R where to put points in space
- these are viewable

```
st_crs(usmap.cont)
```

```
## Coordinate Reference System:
##   User input: NAD83
##   wkt:
## GEOGCRS["NAD83",
##     DATUM["North American Datum 1983",
##         ELLIPSOID["GRS 1980",6378137,298.257222101,
##             LENGTHUNIT["metre",1]]],
##     PRIMEM["Greenwich",0,
##         ANGLEUNIT["degree",0.0174532925199433]],
##     CS[ellipsoidal,2],
##         AXIS["latitude",north,
##             ORDER[1]
```

You Can Change Projections

- if you don't do it properly, you will mess everything up!
- use `st_transform()`

Changing Projections

For example

```
usmap.cont2 <- st_transform(x = usmap.cont, crs = 2163)
st_crs(usmap.cont2)
```

```
## Coordinate Reference System:
##   User input: EPSG:2163
##   wkt:
## PROJCRS["NAD27 / US National Atlas Equal Area",
##     BASEGEOGCRS["NAD27",
##         DATUM["North American Datum 1927",
##             ELLIPSOID["Clarke 1866",6378206.4,294.978698213898,
##                 LENGTHUNIT["metre",1]]],
##         PRIMEM["Greenwich",0,
##             ANGLEUNIT["degree",0.0174532925199433]],
##         ID["EPSG",4267]],
##     CONVERSION["US National Atlas Equal Area",
##         METHOD["Lambert Azimuthal Equal Area (Spherical)"
```

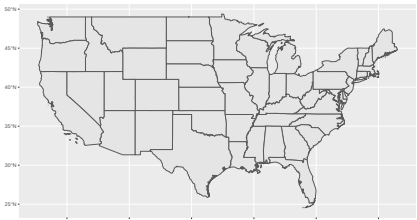
Look at New and Old Projections

Create map with new projection

```
cont.us2 <-
  ggplot() + geom_sf(data = usmap.cont2)
```

Look at New and Old Projections

```
cont.us
cont.us2
```



E. Spatially combining

Questions you can answer with `st_intersection()`

1. Which states are cities in?

E. Spatially combining

Questions you can answer with `st_intersection()`

1. Which states are cities in?
   - points and polygons: should return points

E. Spatially combining

Questions you can answer with `st_intersection()`

1. Which states are cities in?
   - points and polygons: should return points
2. What share of national park land area (polygons) is in cities (polygons)?

E. Spatially combining

Questions you can answer with `st_intersection()`

1. Which states are cities in?
   - points and polygons: should return points
2. What share of national park land area (polygons) is in cities (polygons)?
   - polygons and polygons: should return polygons

E. Spatially combining

Questions you can answer with `st_intersection()`

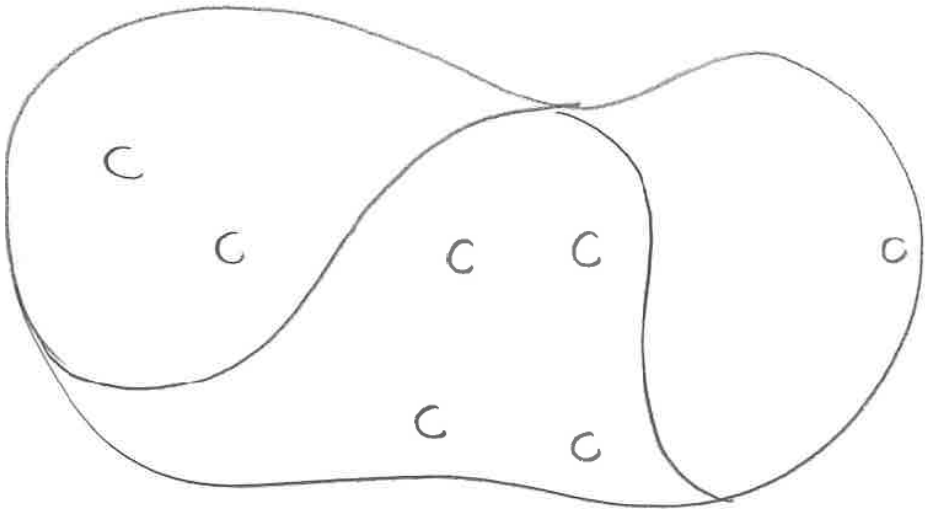
1. Which states are cities in?
   - points and polygons: should return points
2. What share of national park land area (polygons) is in cities (polygons)?
   - polygons and polygons: should return polygons
3. How many miles of roads (lines) are in the 3 western coastal states (polygons)?

E. Spatially combining

Questions you can answer with `st_intersection()`

1. Which states are cities in?
   - points and polygons: should return points
2. What share of national park land area (polygons) is in cities (polygons)?
   - polygons and polygons: should return polygons
3. How many miles of roads (lines) are in the 3 western coastal states (polygons)?
   - lines and polygons: should return lines, then sum to state level

E.1 Example: Which states are cities in?



E.1. Cities and States

```
cities
```

```
##   city.id temp  x  y
## 1       1   70 45 76
## 2       2   60 46 77
## 3       3   50 34 78
```

```
states
```

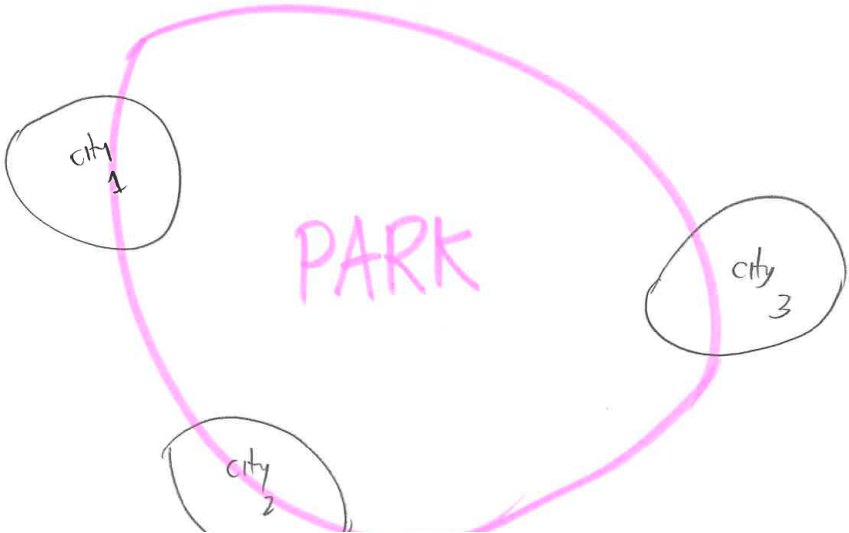
```
##   state.id area polygon
## 1        a  500  pinfo1
## 2        b  100  pinfo2
## 3        c  200  pinfo3
```

E.1. Cities in States

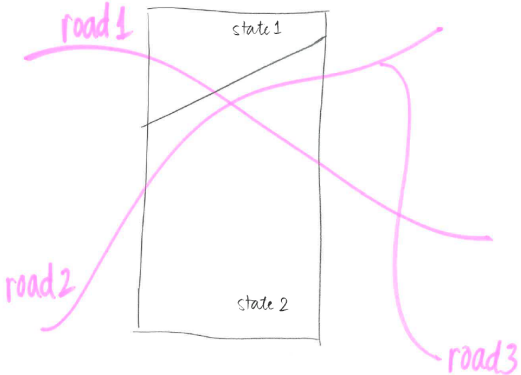
```
cities.in.states
```

```
##   city.id temp  x  y state.id area
## 1       1   70 45 76        a  500
## 2       2   60 46 77        a  500
## 3       3   50 34 78        b  100
```

E.2 Example: What share of national park land area is in cities?



E.3 Example: How many miles of roads in each state?



E.4. How to do it

Use `st_intersection()`

commands

Don't confuse with `st_intersects()` which does the same thing but returns a matrix, not a simple feature.

E.5. Example

Make a set of polygons, called X

```
b0 = st_polygon(list(rbind(c(-1,-1),
                           c(1,-1),
                           c(1,1),
                           c(-1,1),
                           c(-1,-1))))
b1 = b0 + 2
b2 = b0 + c(-0.2, 2)
x = st_sfc(b0, b1, b2)
```

E.5. Simple Feature X

```
x
```

```
## Geometry set for 3 features
## Geometry type: POLYGON
## Dimension:     XY
## Bounding box:  xmin: -1.2 ymin: -1 xmax: 3 ymax: 3
## CRS:           NA
```

```
## POLYGON ((-1 -1, 1 -1, 1 1, -1 1, -1 -1))
```

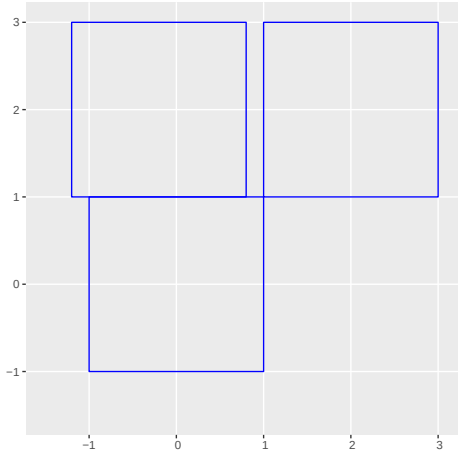
```
## POLYGON ((1 1, 3 1, 3 3, 1 3, 1 1))
```

```
## POLYGON ((-1.2 1, 0.8 1, 0.8 3, -1.2 3, -1.2 1))
```

E.5. Plot X

```
xplot <- ggplot() +
  geom_sf(data = x, color = "blue", fill = NA) +
  scale_x_continuous(limits = c(-1.5,3)) +
  scale_y_continuous(limits = c(-1.5,3))
```

E.5. Plot x



E.5. Example

Make a set of polygons, called Y

```
a0 = b0 * 0.8
a1 = a0 * 0.5 + c(2, 0.7)
a2 = a0 + 1
a3 = b0 * 0.5 + c(2, -0.5)
y = st_sfc(a0,a1,a2,a3)
```

Taken directly from `sf` vignette here.

E.5. Simple Feature Y

```
y
```

```
## Geometry set for 4 features
## Geometry type: POLYGON
## Dimension:     XY
## Bounding box:  xmin: -0.8 ymin: -1 xmax: 2.5 ymax: 1.8
## CRS:           NA
```

```
## POLYGON ((-0.8 -0.8, 0.8 -0.8, 0.8 0.8, -0.8 0....
```

```
## POLYGON ((1.6 0.3, 2.4 0.3, 2.4 1.1, 1.6 1.1, 1...
```

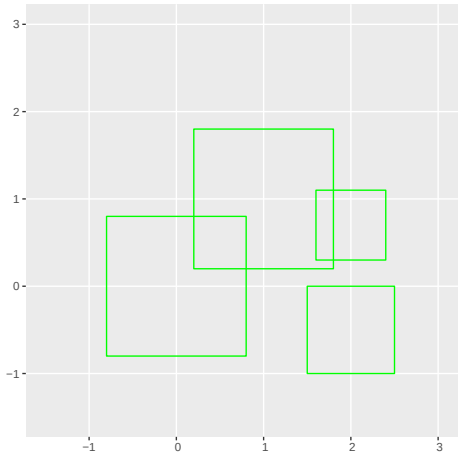
```
## POLYGON ((0.2 0.2, 1.8 0.2, 1.8 1.8, 0.2 1.8, 0...
```

```
## POLYGON ((1.5 -1, 2.5 -1, 2.5 0, 1.5 0, 1.5 -1))
```

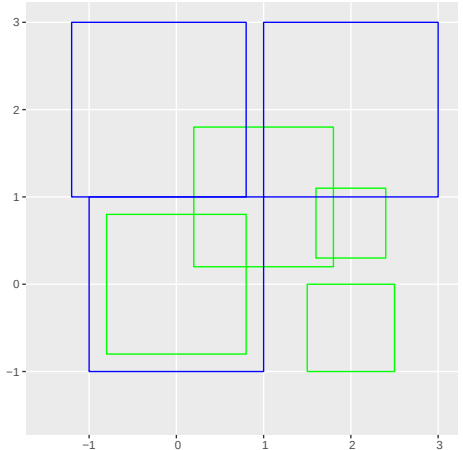
E.5. Plot Y

```
yplot <- ggplot() +
  geom_sf(data = y, color = "green", fill = NA) +
  scale_x_continuous(limits = c(-1.5,3)) +
  scale_y_continuous(limits = c(-1.5,3))
```

E.5. Plot Y



E.5. Plot `X` and `Y` Together



E.6. Intersection

```
xy <- st_intersection(x,y)
```

E.7. How the New Simple Feature Looks

```
xy
```

```
## Geometry set for 5 features
## Geometry type: POLYGON
## Dimension:     XY
## Bounding box:  xmin: -0.8 ymin: -0.8 xmax: 2.4 ymax: 1.8
## CRS:           NA
```

```
## POLYGON ((-0.8 0.8, 0.8 0.8, 0.8 -0.8, -0.8 -0....
```

```
## POLYGON ((1.6 1, 1.6 1.1, 2.4 1.1, 2.4 1, 1.6 1))
```

```
## POLYGON ((1 1, 1 0.2, 0.2 0.2, 0.2 1, 1 1))
```

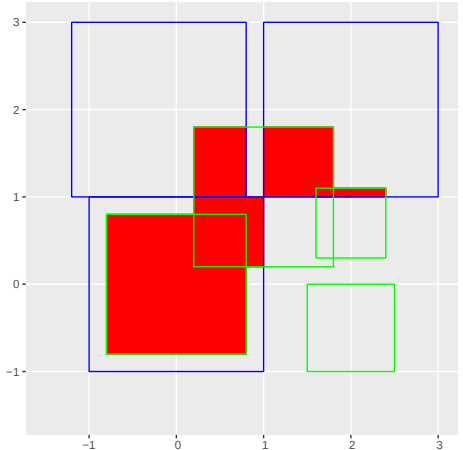
```
## POLYGON ((1 1, 1 1.8, 1.8 1.8, 1.8 1, 1 1))
```

```
## POLYGON ((0.8 1, 0.2 1, 0.2 1.8, 0.8 1.8, 0.8 1))
```

E.8 What the Picture Looks Like

```
xyplot <- ggplot() +
  geom_sf(data = xy, color = "red", fill = "red") +
  geom_sf(data = x, color = "blue", fill = NA) +
  geom_sf(data = y, color = "green", fill = NA) +
  scale_x_continuous(limits = c(-1.5,3)) +
  scale_y_continuous(limits = c(-1.5,3))
```

E.8 What the Picture Looks Like: Red is Intersected Part



Next Lecture

- Next week: line charts
- Following week: come prepared to work on your policy brief storyline

Playing Around in R

- Make a few maps
- Do some choropleths