Econ 106

Lecture 15

slides adapted from: https://github.com/dlab-berkeley/R-Geospatial-Fundamentals/tree/master/docs

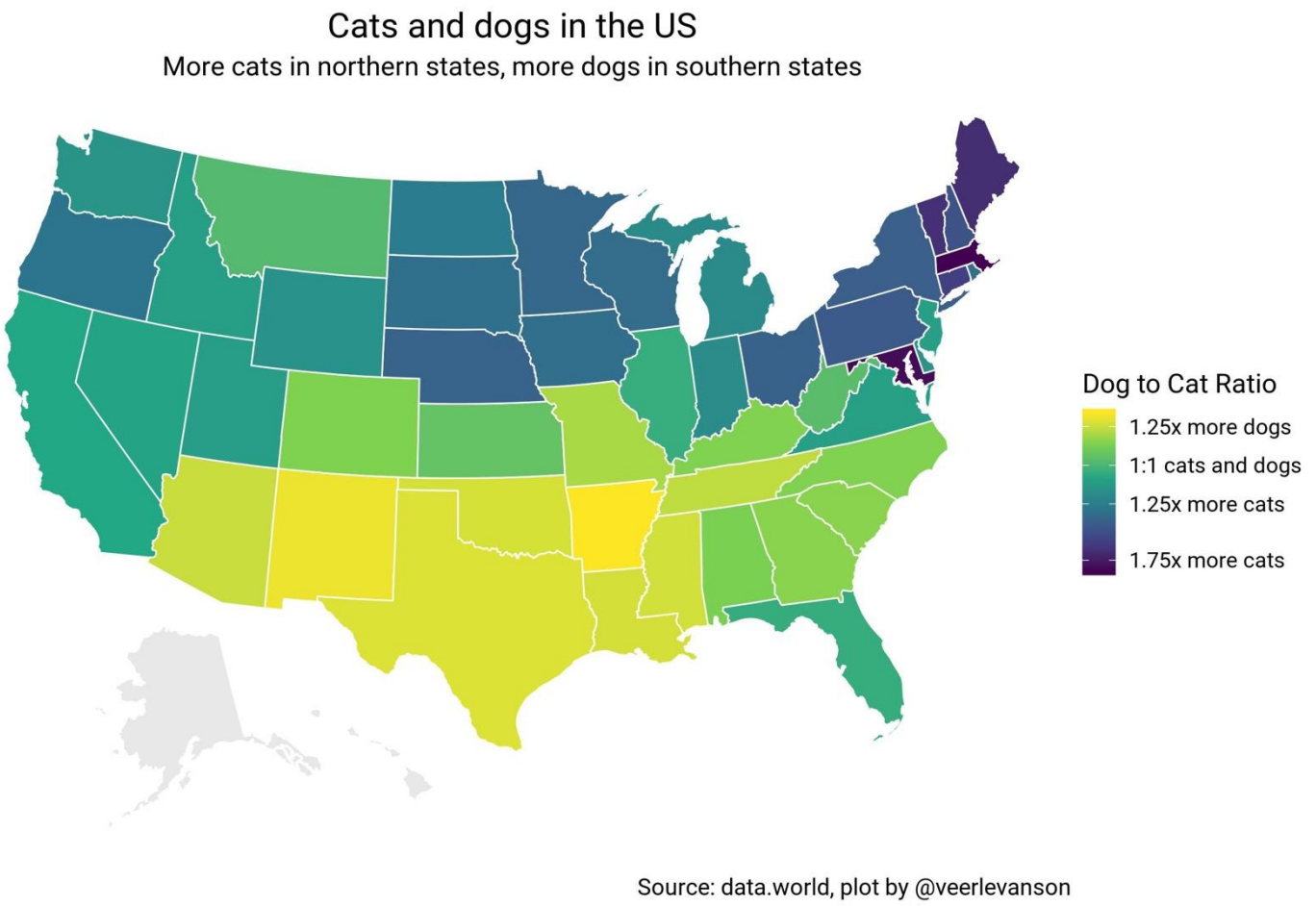
Updates

- Lab #4 is due Sunday 11:59pm

- Next week is Thanksgiving week:
  - Monday: in person lecture
  - Wednesday: no lecture
  - Work on MS# 3

https://pollev.com/vsovero

#tidytuesday



https://surroundedbydata.netlify.app/post/tidy-tuesday/

Outline

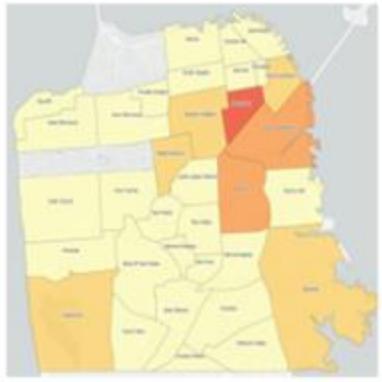
- What is spatial data?
- Spatial data as Vector data
- Coordinate Reference Systems
- Reading in Shape files as simple features
- plotting simple features
- Adding attributes to simple features
- Converting coordinates to simple features
- layering maps of simple features

Spatial Data

- Spatial data contains information regarding shape and location in space

Crime locations

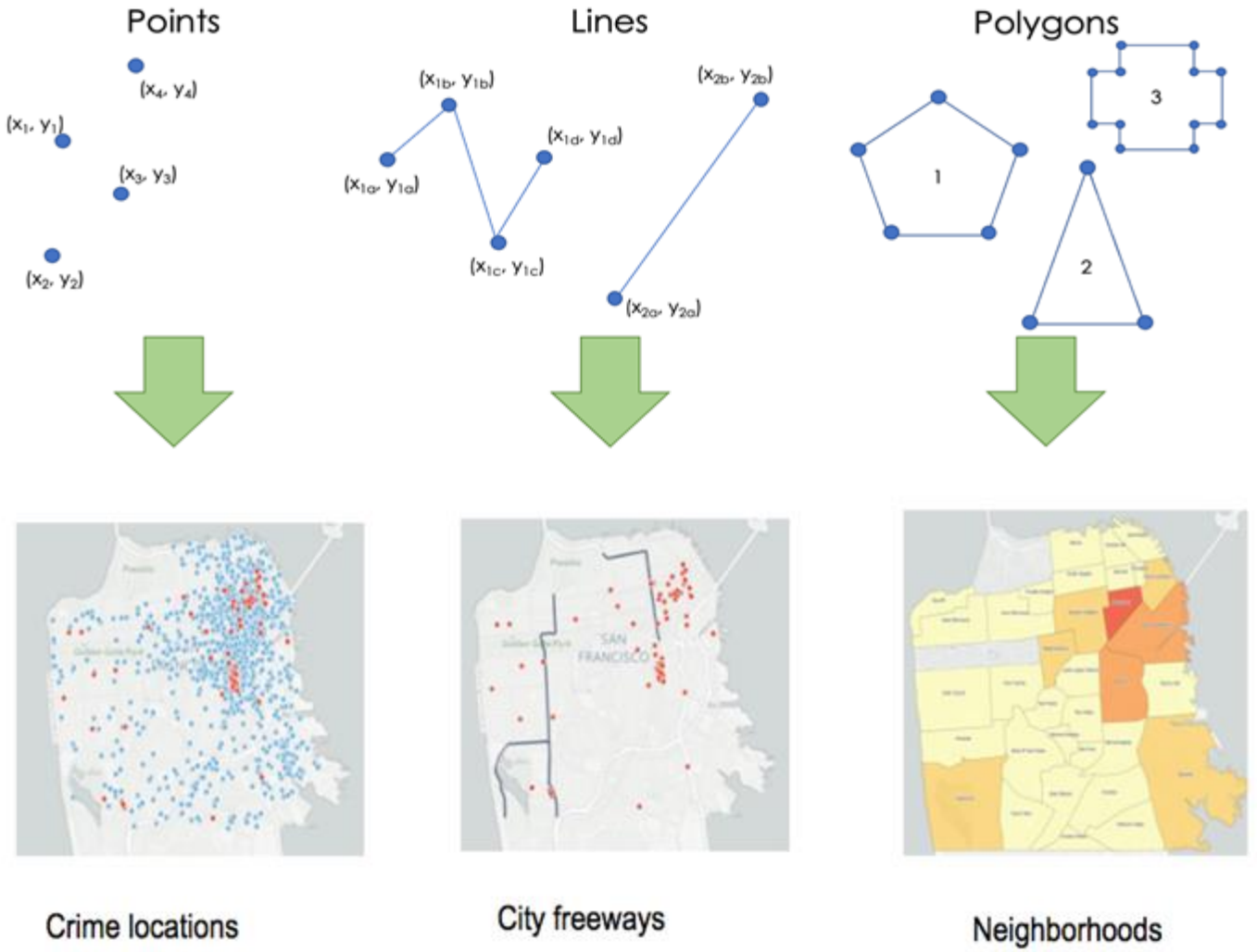
City freeways



Neighborhoods

Spatial Data: Shape

- Spatial data can be drawn using geometric shapes:
  - points: crime locations
  - lines: streets or highways
  - polygons: neighborhoods
- This is referred to as <u>vector</u> data



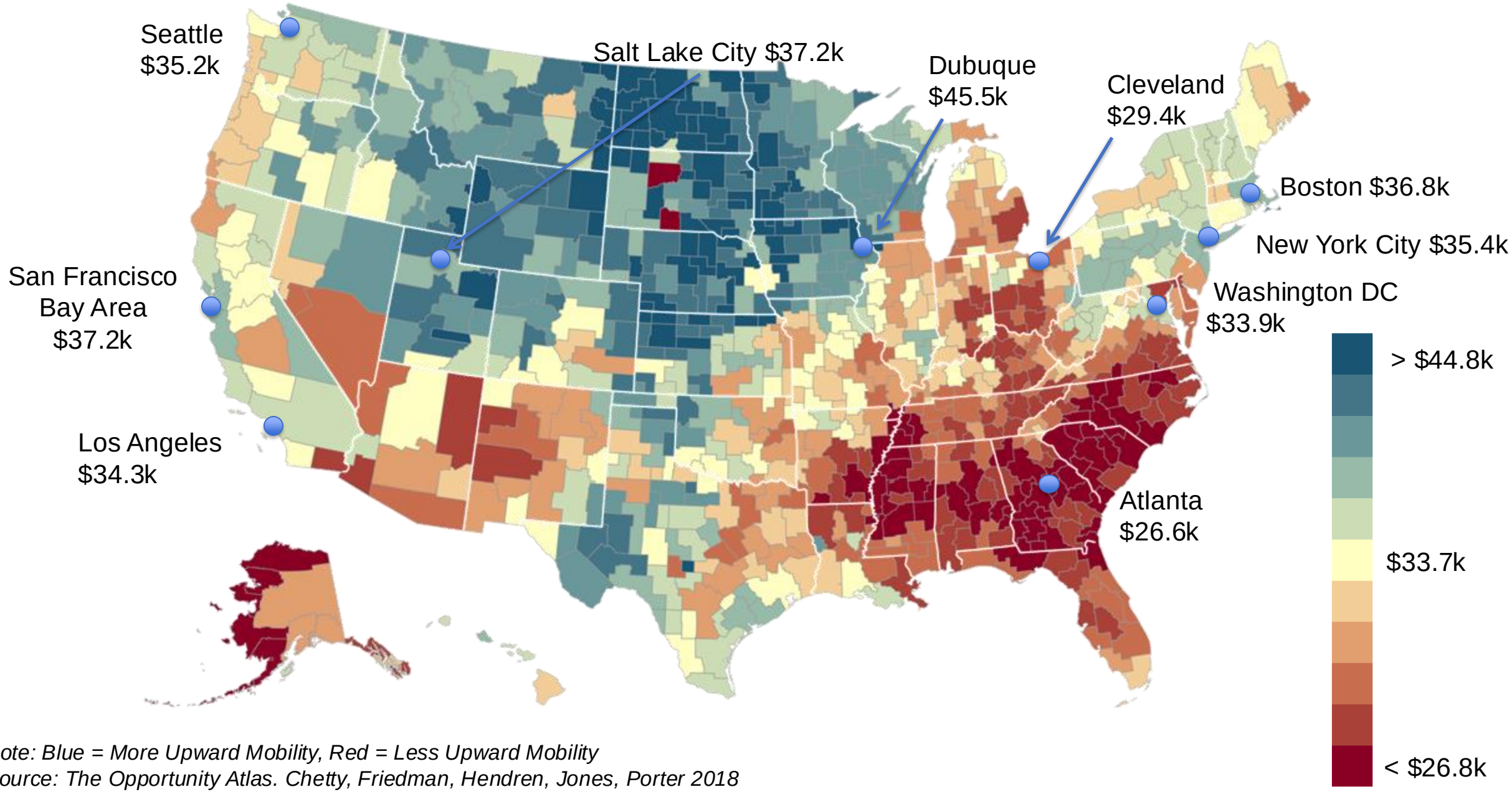
Spatial Data: Location

- Vector data can placed on a grid of the Earth to show location
- the grid is referred to as the Coordinate Reference System (CRS)
- There are many CRS's (more on this later)

https://pollev.com/vsovero

The Geography of Upward Mobility in the United States

Average Household Income for Children with Parents Earning $27,000 (25th percentile)

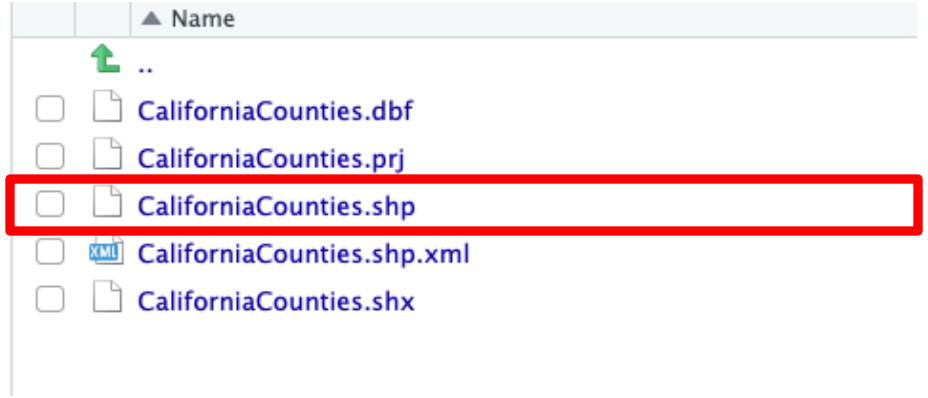


Note: Blue = More Upward Mobility, Red = Less Upward Mobility

Source: The Opportunity Atlas. Chetty, Friedman, Hendren, Jones, Porter 2018

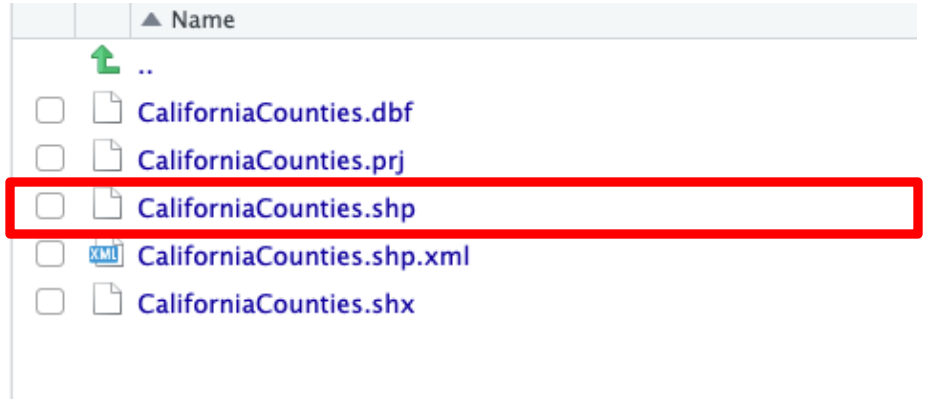
File formats for Vector Data

- The ESRI **shapefile** is the most widely used type of file format for storing geospatial vector data
- A shapefile is actually a collection of 3 or more files:
  - shp: The main file that stores the feature geometry
  - shx: A positional index for locating the feature geometry in the shp file
  - dbf: The data table that stores the attribute information for each feature



Keep your files together!

- You will read the .shp file extension into R
- You need to have the other files in the same folder for the .shp file to be read in correctly



sf Package

- We will be using the sf package to work with shapefiles in R

- sf = simple features

```
install.packages("sf")
library(sf)
```

Reading in a Shapefile

- We are going to read in a shapefile with California county data
- First: change your Files Pane so that it shows the folder with your shapefile
- Second: change sure your working directory is set to the location of the County_California folder:
  - Session->Set working directory->Files Pane Location

Reading in a Shapefile

- we will use st_read() to load the data into R

```
CA_Counties<-st_read("CaliforniaCounties.shp")
```

Filter

	NAME	STATE_NAME	POP2010	POP10_SQMI	POP2012	POP12_SQMI	WHITE	BLACK
1	Kern	California	839631	102.9	851089	104.282870	499766	48921
2	Kings	California	152982	109.9	155039	111.427421	83027	11014
3	Lake	California	64665	48.6	65253	49.082334	52033	1232
4	Lassen	California	34895	7.4	35039	7.422856	25532	2834
5	Los Angeles	California	9818605	2402.3	9904341	2423.264150	4936599	856874
6	Madera	California	150865	70.1	153025	71.065672	94456	5629
7	Marin	California	252409	480.2	255509	486.100489	201963	6987
8	Mariposa	California	18251	12.5	18455	12.613887	16103	138
9	Mendocino	California	87841	25.0	88094	25.083070	67218	622
10	Merced	California	255793	129.4	256841	129.897434	148381	9926
11	Modoc	California	9686	2.3	9791	2.329272	8084	82
12	Mono	California	14202	4.5	14418	4.604771	11697	47
13	Monterey	California	415057	125.2	420465	126.859300	230717	12785
14	Napa	California	136484	173.1	135855	172.308609	97525	2668
15	Nevada	California	98764	101.3	99951	102.564339	90233	389

What does our data look like?

- CA_Counties looks and can operate as a data frame
- But there is also an additional geometry column that stores the simple feature (polygons)

```
CA_Counties<-st_read("CaliforniaCounties.shp")
```

AVE_FAM_SZ	HSE_UNITS	VACANT	OWNER_OCC	RENTER_OCC	CountyFIPS	geometry
3.61	284367	29757	152828	101782	06103	MULTIPOLYGON (((213672.6 -2...
3.59	43867	2634	22329	18904	06089	MULTIPOLYGON (((12524.03 -1...
2.94	35492	8944	17472	9076	06106	MULTIPOLYGON (((-235734.3 1...
2.98	12710	2652	6590	3468	06086	MULTIPOLYGON (((12.28914 35...
3.58	3445076	203872	1544749	1696455	06073	MULTIPOLYGON (((173874.5 -4...
3.63	49140	5823	27726	15591	06102	MULTIPOLYGON (((16681.16 -1...
2.94	111214	8004	64637	38573	06066	MULTIPOLYGON (((-261758.2 2...
2.77	10188	2495	5227	2466	06111	MULTIPOLYGON (((-4613.85 -9...
3.02	40323	5378	20601	14344	06100	MULTIPOLYGON (((-307250.8 8...
3.74	83698	8056	41196	34446	06099	MULTIPOLYGON (((-34084.85 -...
2.80	5192	1128	2786	1278	06108	MULTIPOLYGON (((12.28914 35...
2.98	13912	8144	3228	2540	06067	MULTIPOLYGON (((191409.7 -5...
3.66	139048	13102	64077	61869	06070	MULTIPOLYGON (((-131040 -23...
3.23	54759	5883	30597	18279	06064	MULTIPOLYGON (((-183172 572...
2.80	52590	11063	29890	11637	06094	MULTIPOLYGON (((-266.9455 1...

What's the object type?

- We can use class() to see how CA_Counties is being stored in R
- This confirms that CA_Counties is a data frame with simple features (geometry)

```
class(CA_Counties)
```

```
> class(CA_Counties)
[1] "sf"         "data.frame"
```

Ok, so how do we visualize geospatial data?

- we'll use ggplot() and geom_sf()
- no mapping argument needed because the default is to use the geometry column

```
ggplot(data=CA_Counties) +
    geom_sf()
```

Map of California Counties

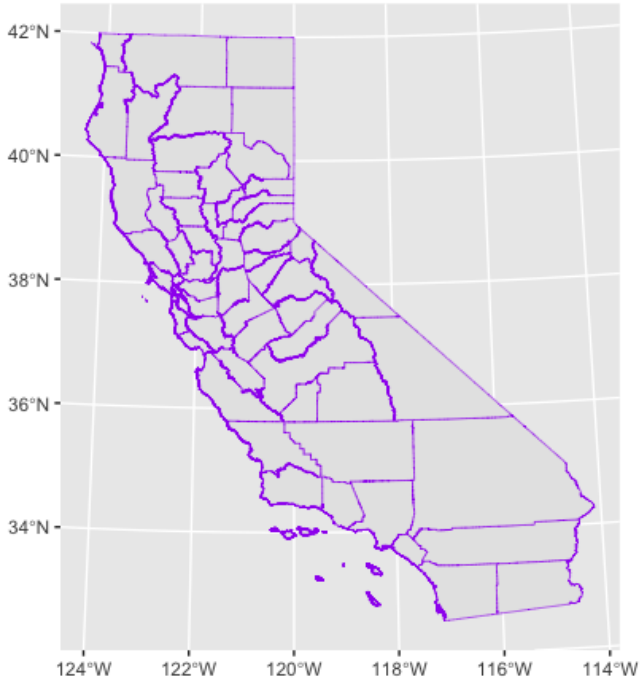
- Now we can see that CA_Counties has the outlines of all the counties in California

```
ggplot(data=CA_Counties) +
geom_sf()
```

Color Setting

- we can set the color of the outlines

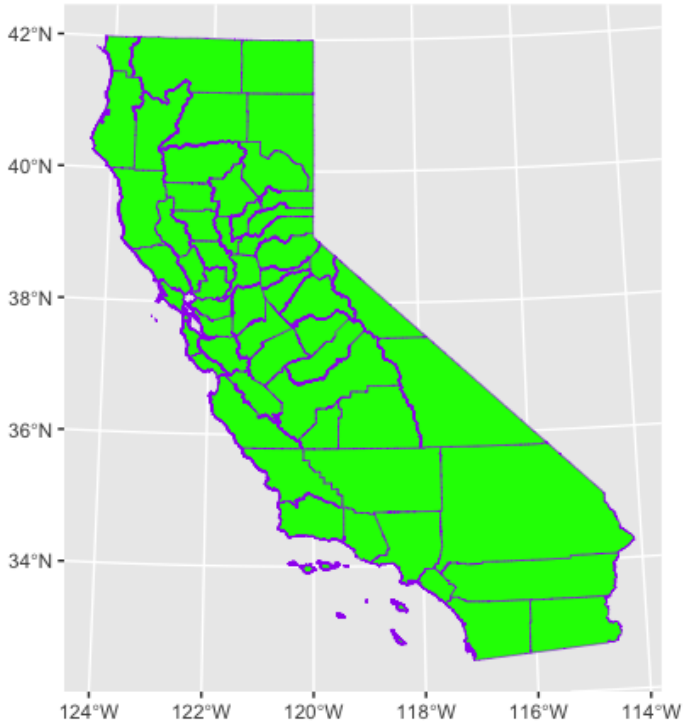
```
ggplot(data=CA_Counties) +
    geom_sf(color= 'purple')
```



Color Setting

- we can fill the counties with color

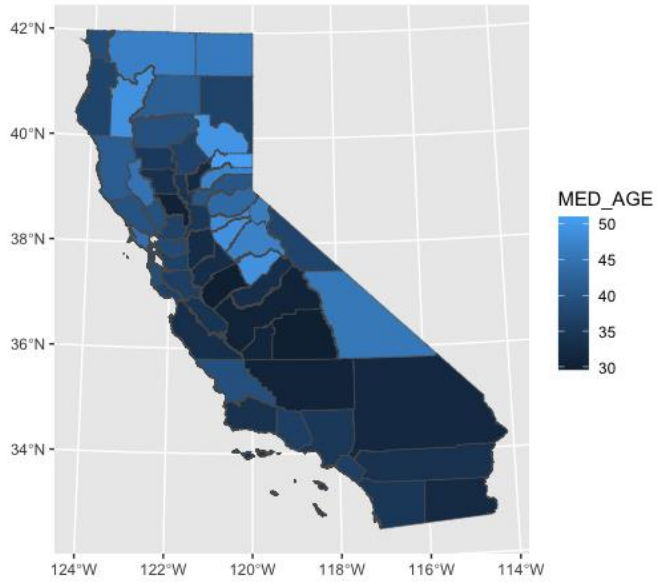
```
ggplot(data=CA_Counties) +
    geom_sf(color= 'purple' , fill= 'green')
```



Color Mapping

- we can map color to a variable in the CA_Counties data
- remember to wrap variable names with the aes() function

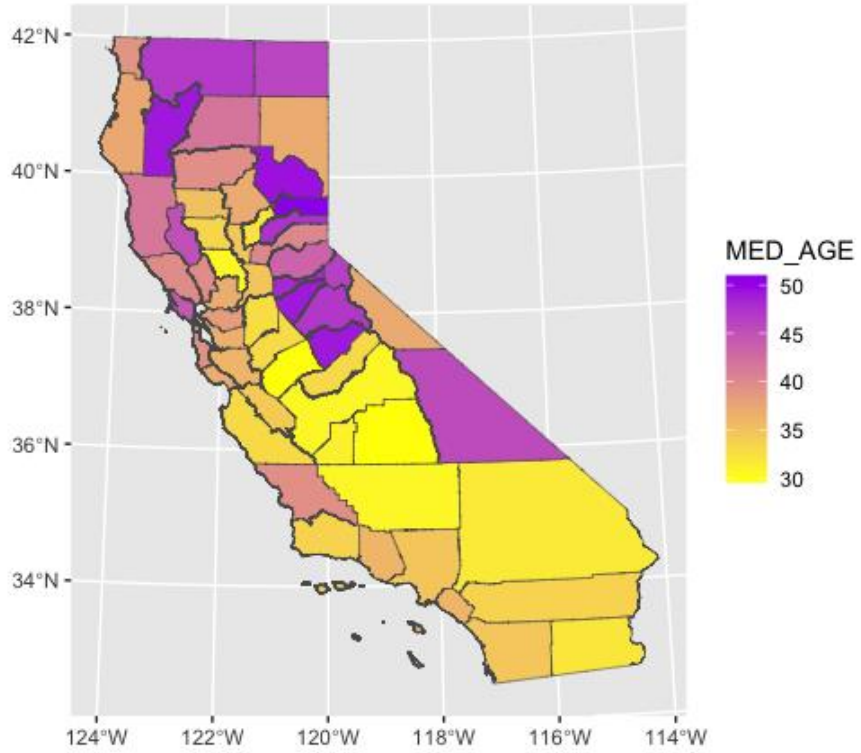
```
ggplot(data=CA_Counties) +
    geom_sf(aes(fill= MED_AGE ))
```



Adjust the Color

- we can customize the color scale to have more contrast

```
ggplot(data=CA_Counties) +
        geom_sf(aes(fill= MED_AGE )) +
scale_fill_gradient(low="yellow",  high="purple")
```



Class Exercise

- plot the California counties filled in with average household size
- adjust the color scale so low is blue and high is red

https://pollev.com/vsovero

Converting csv files to a sf data frame

- Sometimes we have a csv file with lat/lon coordinates
- Example: csv file of all postsecondary universities in the US
- Before we can map it or do any spatial analysis, we need to convert it to a sf data frame

First, read in the csv file

- We will read in the csv file as a data frame
- You can see that there are two variables that represent latitude and longitude

```
college_data <- read_csv("college locations.csv")
```

COUNTYCD	COUNTYNM	CNGDSTCD	LONGITUD	LATITUDE	DFRCGID	DFRCUSCG
1089	Madison County	105	-86.56850	34.78337	109	1
1073	Jefferson County	107	-86.79935	33.50570	93	1
1101	Montgomery County	102	-86.17401	32.36261	127	2
1089	Madison County	105	-86.64045	34.72456	93	2
1101	Montgomery County	107	-86.29568	32.36432	99	1
1125	Tuscaloosa County	107	-87.52959	33.20701	-2	-2
1125	Tuscaloosa County	107	-87.54598	33.21187	92	1
1123	Tallapoosa County	103	-85.94527	32.92478	74	2
1083	Limestone County	105	-86.96470	34.80679	136	1
1101	Montgomery County	102	-86.17754	32.36736	109	1
1081	Lee County	103	-85.48826	32.59938	92	1
1073	Jefferson County	107	-86.85055	33.51377	133	1
1113	Russell County	103	-85.03149	32.42391	71	2
1101	Montgomery County	102	-86.21649	32.34268	201	1
1031	Coffee County	102	-85.83696	31.29750	65	2

Next, filter for CSU's

- Let's filter for universities in the CSU system

```
CSU_data <-college_data %>%
  filter(F1SYSNAM == "California State University")
```

Finally, Convert to a sf object

- We will use st_as_sf() to convert the data frame to a sf object
- Arguments:
  - the data frame you want to convert
  - the variable names that contain the coordinates
  - the CRS (see next slide)

```
CSU_sf <- st_as_sf(CSU_data,
          coords = c("LONGITUD", "LATITUDE"),
          crs = 4269)
```

Geographic Coordinate Systems (GCS)

Data stored as coordinates typically use one of these two geographic CRS:

WGS84 (EPSG: 4326)

- Based on satellites, used by cell phones, GPS
- Best overall fit for most places on earth

NAD83 (EPSG: 4269)

- Based on satellites and survey data
- Best fit for USA
- Used by many federal data products, like Census data

st_as_sf()

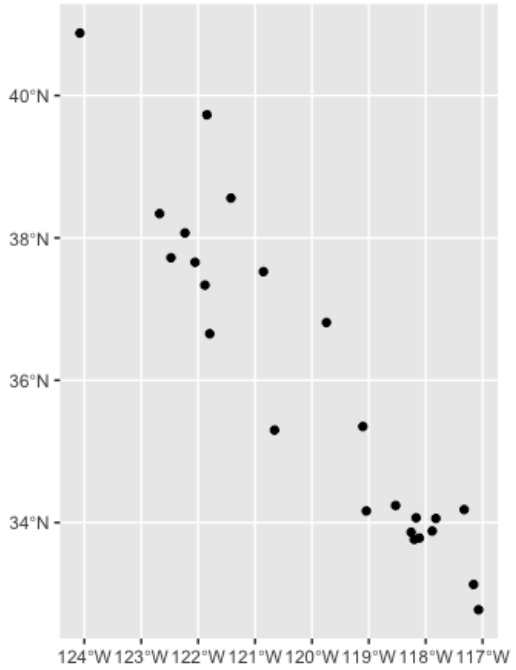
- When we convert the CSU_data to a sf object, the latitude and longitude columns disappear
- In its place is the geometry column
- We can see our spatial data is stored as point features

COUNTYCD	COUNTYNM	CNGDSTCD	DFRCGID	DFRCUSCG	geometry
6079	San Luis Obispo County	624	106	2	POINT (-120.6573 35.29951)
6029	Kern County	623	107	1	POINT (-119.1047 35.35001)
6099	Stanislaus County	610	107	2	POINT (-120.8525 37.52577)
6037	Los Angeles County	647	-2	-2	POINT (-118.2013 33.76434)
6071	San Bernardino County	631	95	1	POINT (-117.3238 34.18262)
6037	Los Angeles County	639	106	1	POINT (-117.8215 34.05791)
6007	Butte County	601	106	1	POINT (-121.8449 39.72971)
6037	Los Angeles County	644	106	2	POINT (-118.2559 33.86477)
6019	Fresno County	622	95	2	POINT (-119.7446 36.81115)
6059	Orange County	639	95	1	POINT (-117.8854 33.88151)
6001	Alameda County	615	96	2	POINT (-122.0541 37.65769)
6037	Los Angeles County	647	95	2	POINT (-118.112 33.78282)
6037	Los Angeles County	634	106	1	POINT (-118.169 34.06693)
6037	Los Angeles County	630	106	1	POINT (-118.5293 34.24013)
6067	Sacramento County	606	106	1	POINT (-121.4235 38.55942)

Let’s plot the CSU’s

```
ggplot(data=CSU_sf) +
geom_sf()
```

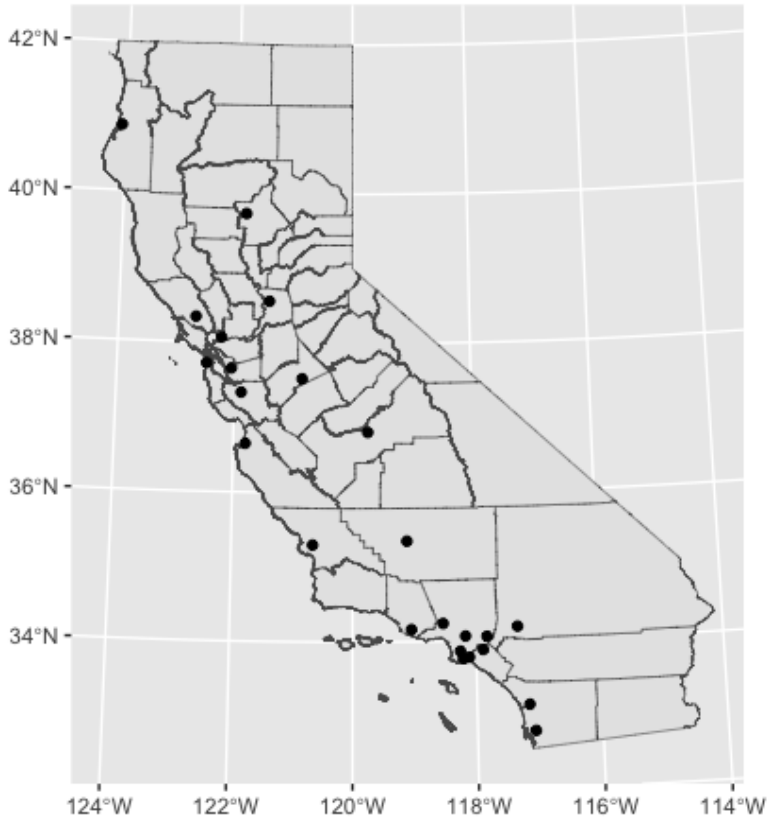
- CSU locations on the coordinate grid
- It would look better overlayed on a map of California



https://pollev.com/vsovero

Layering with ggplot

- In some instances, you may want to overlay one map on top of another.
- To do this, you simply add geom_sf() layers to a plot



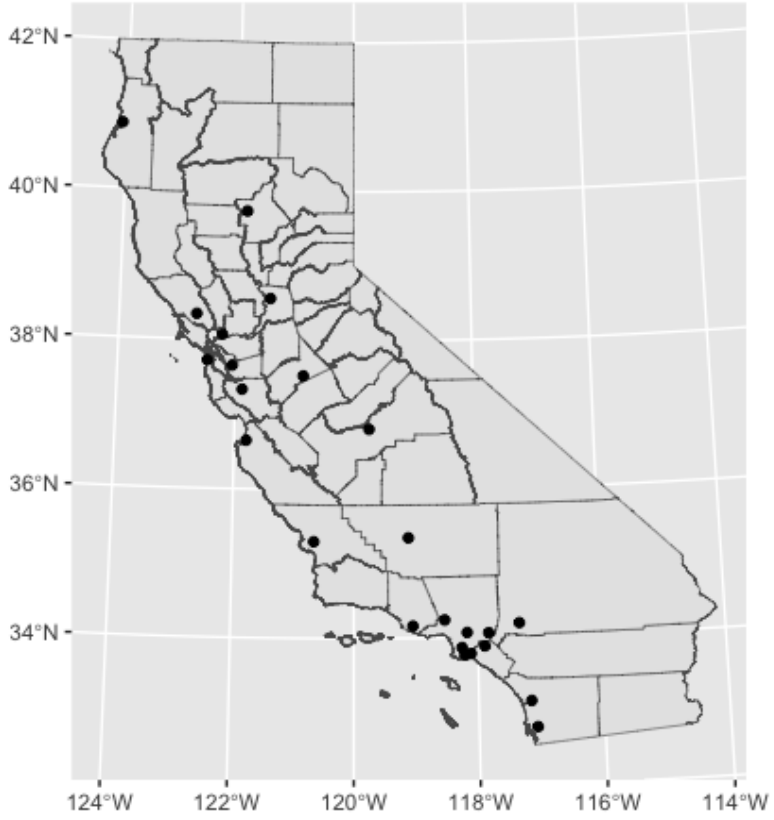
Layering CA Counties and the CSU campuses

- When overlaying data from different sources, you need to move the data argument to the geom_sf()

```
ggplot() +
        geom_sf(data=CA_Counties ) +
        geom_sf(data=CSU_sf)
```

Layering LA County and CA Counties

```
ggplot() +
        geom_sf(data=CA_Counties ) +
        geom_sf(data=CSU_sf)
```

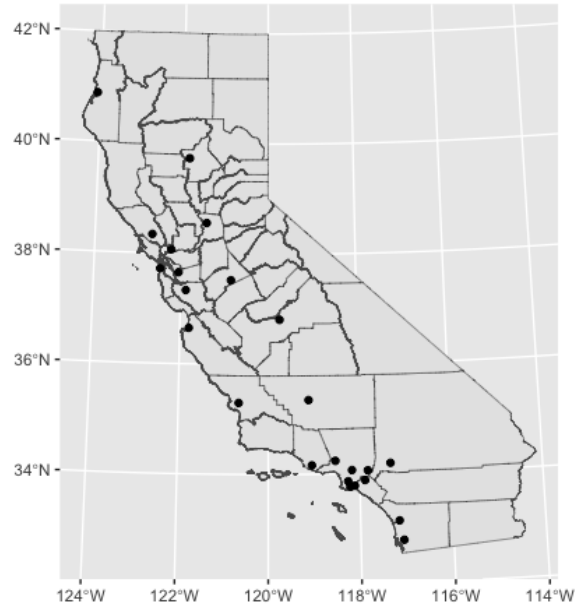


https://pollev.com/vsovero

Layer Order Matters

- ggplot is literally layering one map on top of the other
- Currently, the CSU campuses are layered over CA_Counties
- What happens when I change the order of the layers?

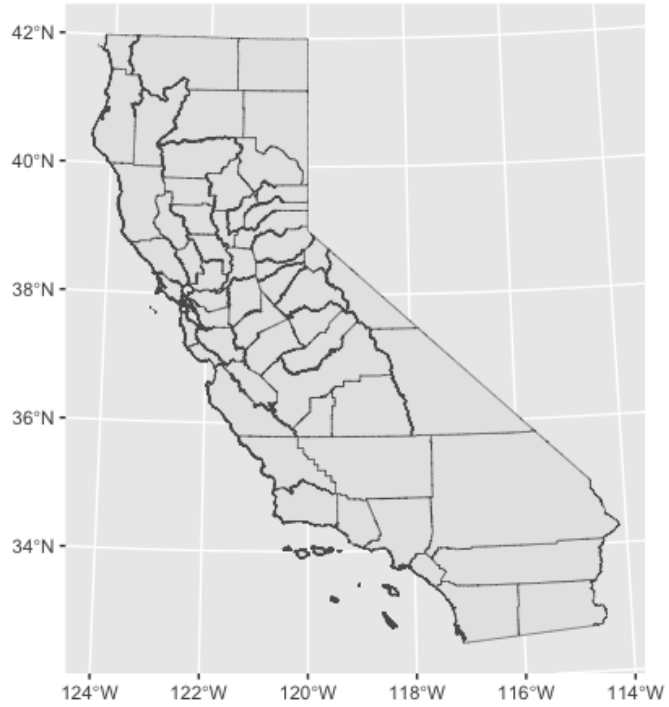
```
ggplot() +
        geom_sf(data=CA_Counties ) +
        geom_sf(data=CSU_sf)
```



Layer Order Matters

- What happened?
- In CA_Counties, all of the counties are filled in with the default gray color
- This covered the black points that represent the CSU campuses

```
ggplot() +
        geom_sf(data=CSU_sf) +
        geom_sf(data=CA_Counties)
```



Data Wrangling and Shapefiles

- Because the shapefile is essentially a data frame with an extra geometry column, we can:
  - create new variables
  - filter for cases
  - join with other data frames

Filtering

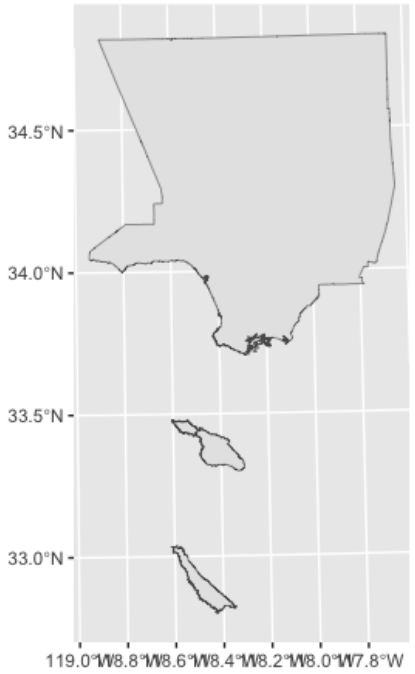
- Let’s say I only want to map LA county
- I use dplyr to filter for LA county
- Importantly, LA_County is a sf object with the geometry column

```
LA_County<-CA_Counties%>%
  filter(NAME=="Los Angeles")
```

LA County

- LA_County is still an sf object with the geometry column, which means we can map it

```
ggplot(data=LA_County) +
geom_sf()
```



Joins

- Let's say you have additional county level data you would like to add to your spatial data:
  - csv file from https://www.opportunityatlas.org/
- How do we join it to CA_Counties?

```
college_county<-read_csv("cty_coll_rP_gP_p25.csv")
```

	cty	Name	College_Graduation_Rate_rP_gP_p25
1	cty38001	Adams County, ND	0.8348
2	cty16023	Butte County, ID	0.7376
3	cty20071	Greeley County, KS	0.6687
4	cty08061	Kiowa County, CO	0.5765
5	cty20179	Sheridan County, KS	0.5370
6	cty30015	Chouteau County, MT	0.5357
7	cty20153	Rawlins County, KS	0.5295
8	cty38025	Dunn County, ND	0.5196
9	cty38091	Steele County, ND	0.5143
10	cty30039	Granite County, MT	0.5142
11	cty08027	Custer County, CO	0.5122
12	cty08047	Gilpin County, CO	0.5076
13	cty38039	Griggs County, ND	0.5065
14	cty46021	Campbell County, SD	0.4851
15	cty31073	Gosper County, NE	0.4789

Joins

- First, we have extract the FIPS code from the cty variable
- This new variable will be used as the linking variable with CA_Counties

```
county_college_fips<-county_college%>%
  mutate(CountyFIPS=str_replace(cty,"cty", ""))
```

	cty	Name	College_Graduation_Rate_rP_gP_p25	CountyFIPS
1	cty38001	Adams County, ND	0.8348	38001
2	cty16023	Butte County, ID	0.7376	16023
3	cty20071	Greeley County, KS	0.6687	20071
4	cty08061	Kiowa County, CO	0.5765	08061
5	cty20179	Sheridan County, KS	0.5370	20179
6	cty30015	Chouteau County, MT	0.5357	30015
7	cty20153	Rawlins County, KS	0.5295	20153
8	cty38025	Dunn County, ND	0.5196	38025
9	cty38091	Steele County, ND	0.5143	38091
10	cty30039	Granite County, MT	0.5142	30039
11	cty08027	Custer County, CO	0.5122	08027
12	cty08047	Gilpin County, CO	0.5076	08047
13	cty38039	Griggs County, ND	0.5065	38039
14	cty46021	Campbell County, SD	0.4851	46021
15	cty31073	Gosper County, NE	0.4789	31073

sf always on the left

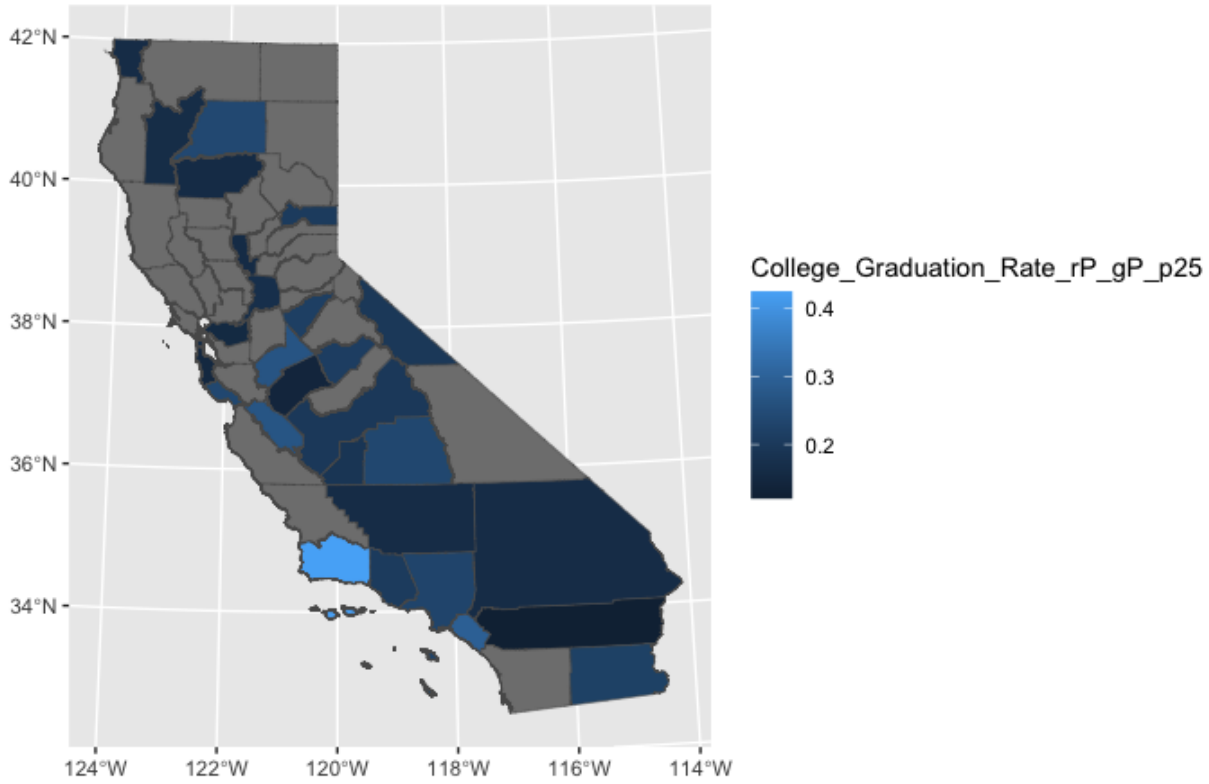
- When joining a data frame with a sf object, you have to specify the sf on the left (x)
- This ensures that the new object is also a sf object (check it's class to confirm)

```
CA_Counties_x<-left_join(x=CA_Counties, y=county_college_fips)

class(CA_Counties_x)
```

College attainment by county

- Why do we have counties in gray?
- Counties that have missing college attainment info (couldn’t be matched to the county education data)



Class Exercise

- Select the universities in the UC system (do not include the office of the president)
- convert the data frame into a sf object
- layer the points on the county map of California using ggplot
  - color the UC campuses white
  - color the CSU campuses red
  - fill the counties based on the proportion of college graduates
  - Do counties with a UC campus or CSU campus have more college graduates?