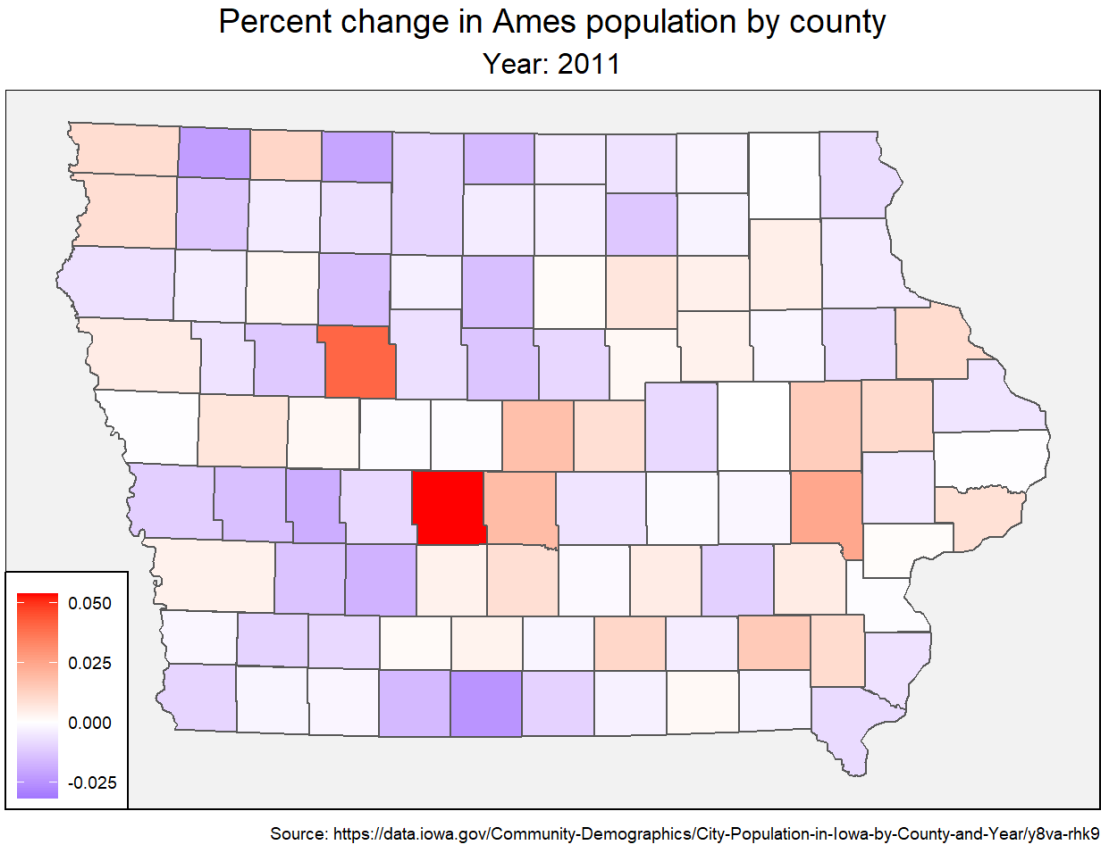
Map visualization basics

Paul Villanueva

11/20/2020

Today's goal:

- Reproduce this animation:



Along the way, we'll:

- Learn some basics about shapefiles
- Work with some open data from data.iowa.gov and geodata.iowa.gov
- Use shapefiles with `ggplot2`
- Do a basic animation

About me

- Bioinformatics PhD student in Adina Howe's lab
- Background in math and computer science
- Previous project was MetaFunPrimer, a primer design pipeline for high-throughput qPCR.
- Current research is on the impact of human activity on biodiversity and environmental multifunctionality.
- Interested in machine learning, statistical modeling, Python, R, data visualiziation, reproducible research, and VIM

Getting started

Clone the repo:

```
git clone https://github.com/pommevilla/lunchinatoR.git
```

Install the packages we'll use today

- `tidyverse`: data manipulation and visualization
- `sf`: working with shapefiles
- `gganimate`: animations
- `ggthemes`: we'll use `theme_map()`
- `geofacet`: some bonus visualizations

Reading in a shapefile

```
library(sf)

iowa.sf <- st_read('data/iowa_county_shapes')
```

Source: https://geodata.iowa.gov/dataset/county-boundaries-iowa

What's in a shapefile?

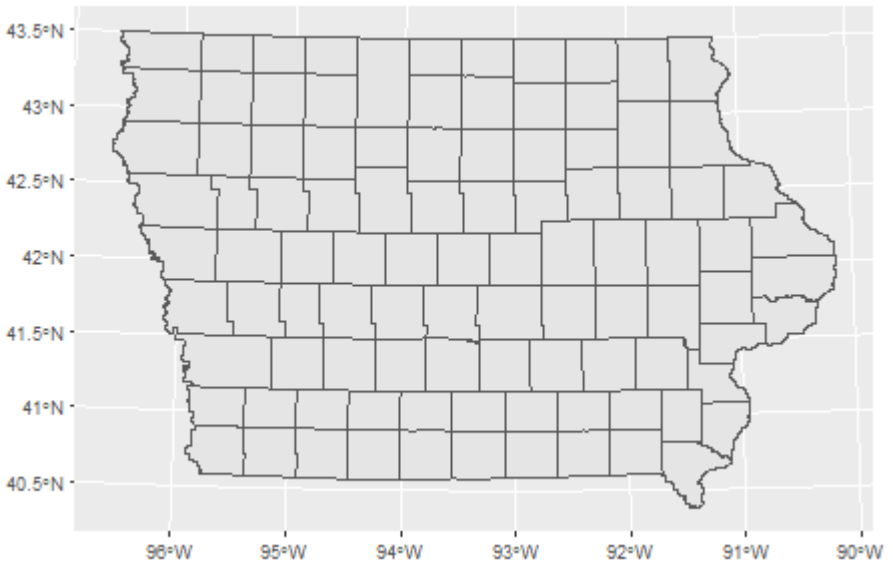
```
iowa.sf
```

```
## Simple feature collection with 99 features and 10 fields
## geometry type:  POLYGON
## dimension:      XY
## bbox:           xmin: 202073.8 ymin: 4470598 xmax: 736849.2 ymax: 4822674
## projected CRS:  NAD83 / UTM zone 15N
## First 10 features:
##    Shape_Leng Shape_Area       AREA PERIMETER CO_NUMBER CO_FIPS    ACRES
## 1    192784.4 1394779996 1394779996  192783.7        56     111 344657.6 1
## 2    146566.4 1335646711 1335646711  146566.5         4       7 330045.5 1
## 3    147784.7 1364467491 1364467491  147785.1        93     185 337167.3 1
## 4    148600.6 1381358612 1381358612  148600.8        27      53 341341.1 1
## 5    144994.2 1306355106 1306355106  144995.3        26      51 322807.4 1
## 6    146043.0 1270590821 1270590821  146043.4        89     177 313969.8 1
## 7    149323.9 1394401127 1394401127  149323.5        80     159 344564.0 1
## 8    150053.8 1384384206 1384384206  150052.3        87     173 342088.8 1
## 9    151615.0 1386916974 1386916974  151614.2        73     145 342714.6 1
## 10   152472.0 1338533928 1338533928  152470.5        36      71 330758.9 1
##       COUNTY ST                          geometry
## 1        Lee IA POLYGON ((634170.7 4519205,...
## 2  Appanoose IA POLYGON ((530372.2 4527603,...
```

Use `ggplot::geom_sf` to plot shapefiles

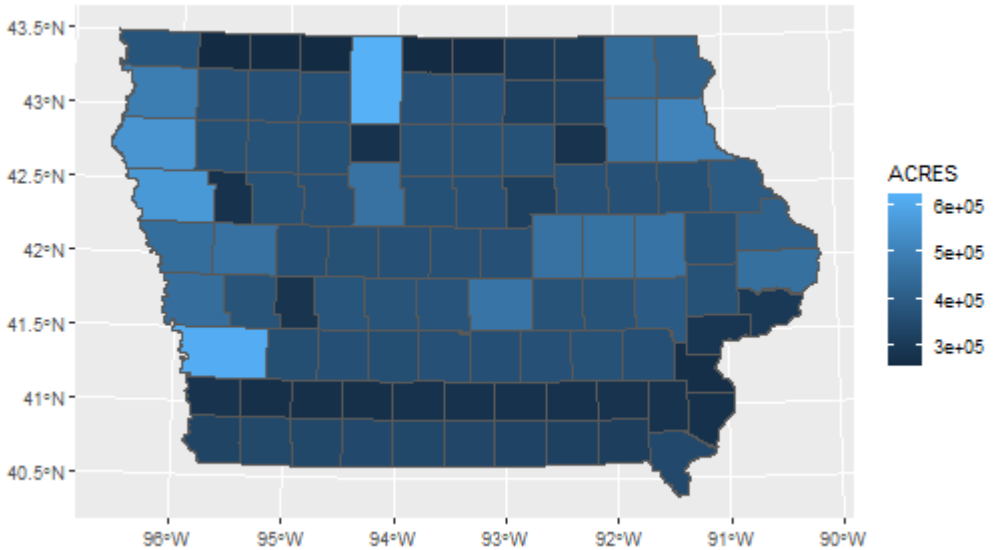
```
library(tidyverse)
```

```
iowa.sf %>%
  ggplot() +
  geom_sf()
```



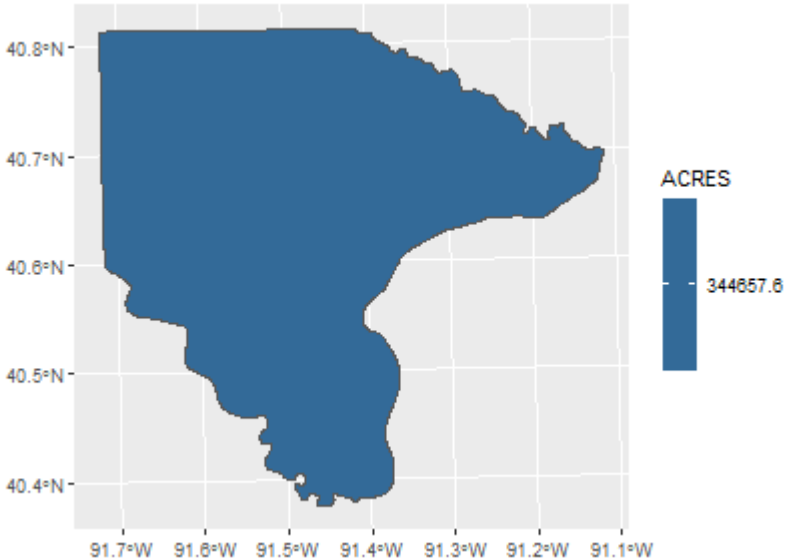
ggplot and dplyr things work with sf objects

```
iowa.sf %>%
  ggplot() +
  geom_sf(aes(fill = ACRES))
```



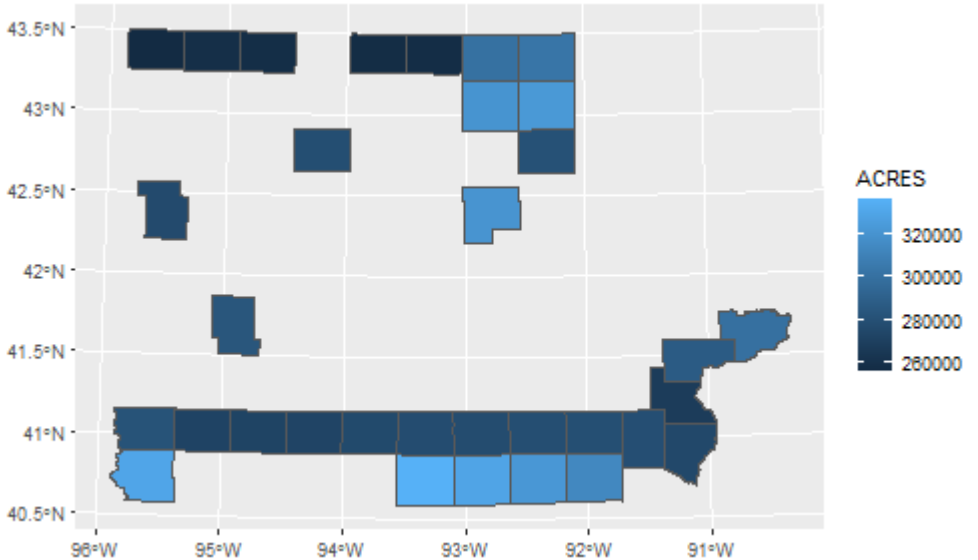
ggplot and dplyr things work with sf objects

```
iowa.sf %>%
  filter(COUNTY == "Lee") %>%
  ggplot() +
  geom_sf(aes(fill = ACRES))
```



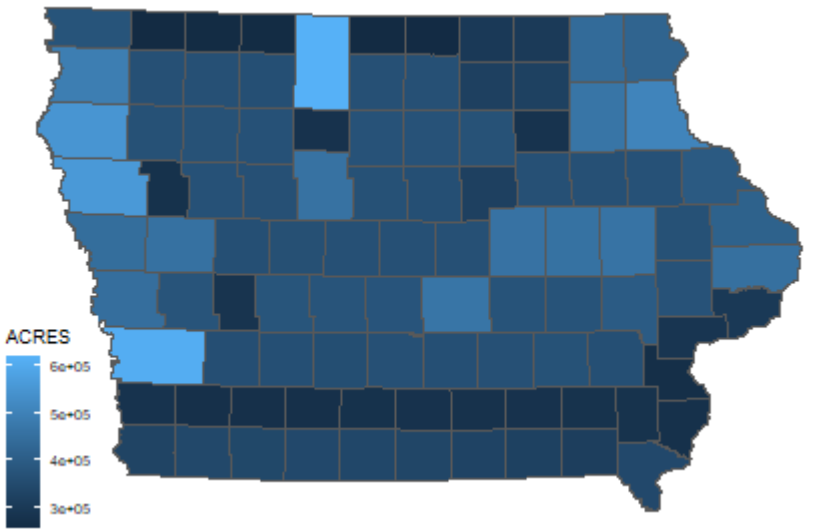
ggplot and dplyr things work with sf objects

```
iowa.sf %>%
  filter(ACRES < 340000) %>%
  ggplot() +
  geom_sf(aes(fill = ACRES))
```

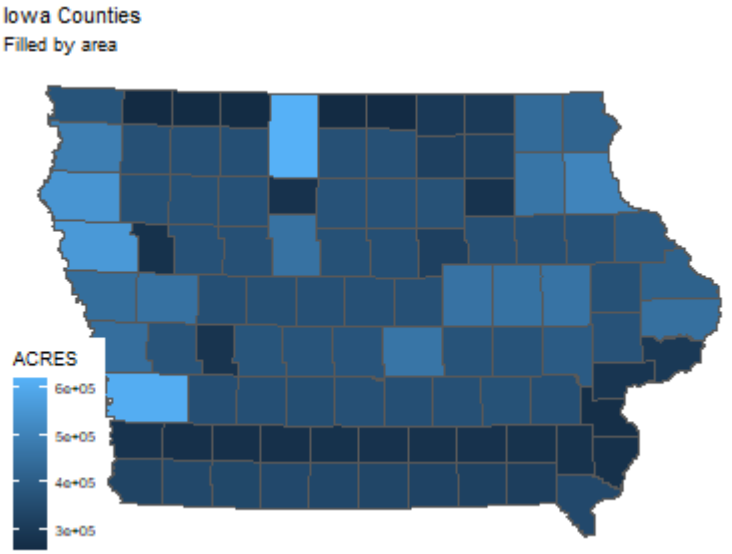


Let's make it look nicer

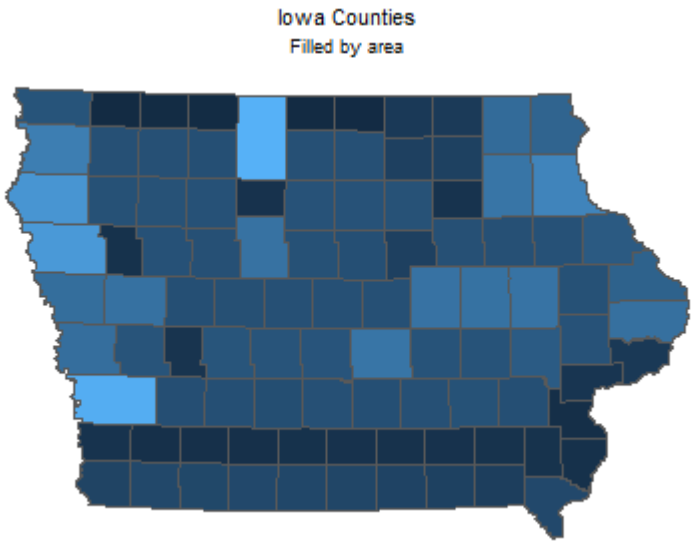
```
theme_set(ggthemes::theme_map())

iowa.sf %>%
  ggplot() +
  geom_sf(aes(fill = ACRES))
```



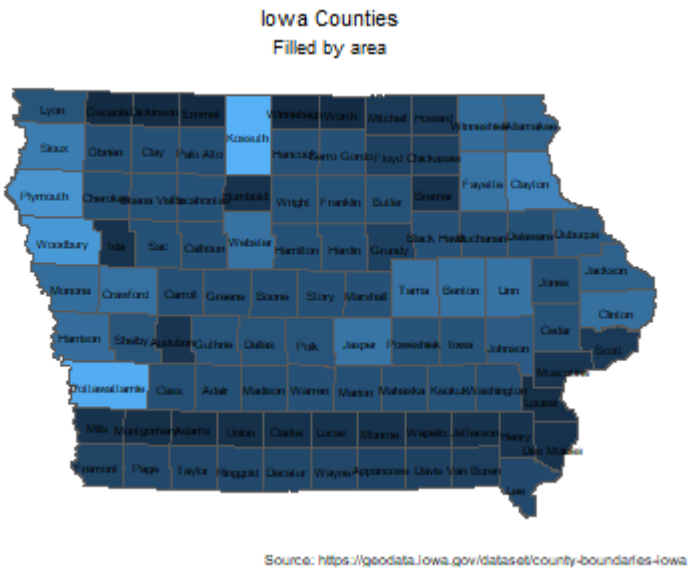
```
iowa.sf %>%
  ggplot() +
  geom_sf(aes(fill = ACRES)) +
  labs(title = "Iowa Counties",
       subtitle = "Filled by area")
```



```
iowa.sf %>%
  ggplot() +
  geom_sf(aes(fill = ACRES)) +
  labs(title = "Iowa Counties",
       subtitle = "Filled by area") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        legend.position = "none")
```



```r
iowa.sf %>%
  ggplot() +
  geom_sf(aes(fill = ACRES)) +
  labs(title = "Iowa Counties",
       subtitle = "Filled by area",
       caption = "Source: https://geodata.iowa.gov/dataset/county-bou
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        legend.position = "none") +
  geom_sf_text(aes(label = COUNTY), size = 2.25)
```



Plotting multiple `sf` objects together

Let's work with another `sf` object:

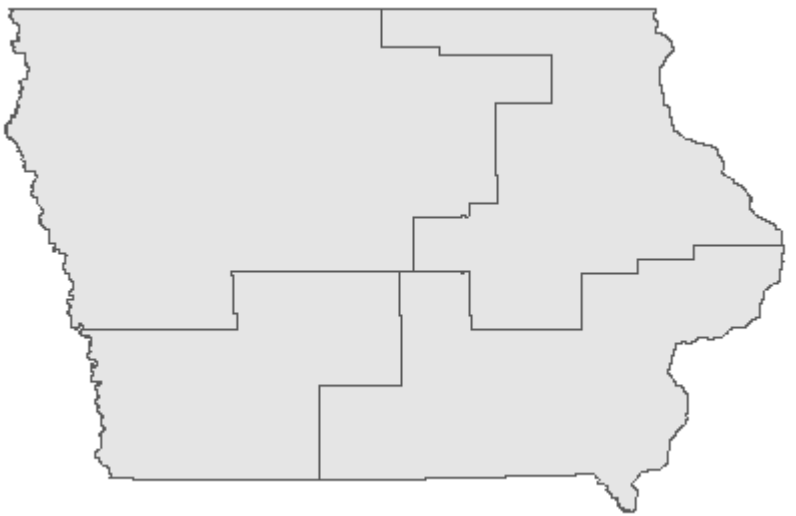
```
iowa.districts <-
  st_read('data/iowa_congressional_districts')
```

```
iowa.districts
```

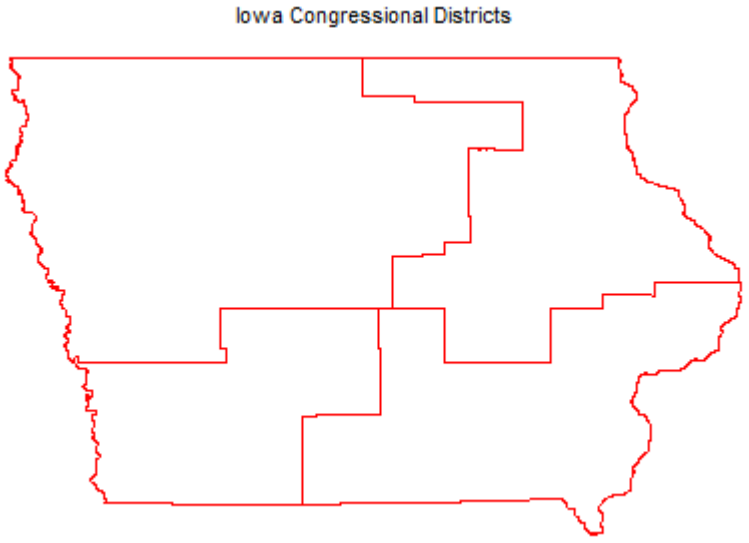
Source: http://cdmaps.polisci.ucla.edu

Let's see what the districts look like

```
iowa.districts %>%
  ggplot() +
  geom_sf()
```



```
iowa.districts %>%
  ggplot() +
  geom_sf(color = "red", fill = "white" ) +
  labs(title = "Iowa Congressional Districts") +
  theme(plot.title = element_text(hjust = 0.5))
```

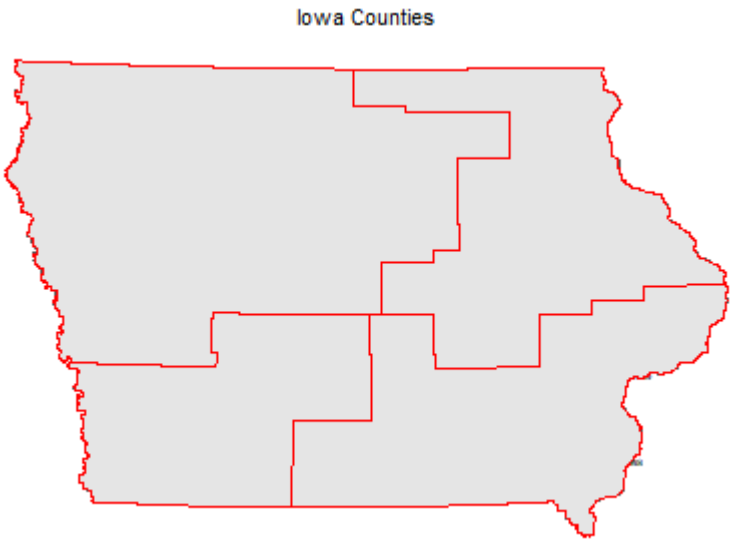


To plot multiple `sf` objects together, call `geom_sf` again and specify the new `sf` object:

```
iowa.sf %>%
  ggplot() +
  geom_sf(fill = "white") +
  labs(title = "Iowa Counties") +
  theme(plot.title = element_text(hjust = 0.5)) +
  geom_sf_text(aes(label = COUNTY), size = 2.25) + # inherits iowa.s
  geom_sf(data = iowa.districts, color = "red")
```

Be careful when plotting multiple `sf` objects together...

```
iowa.sf %>%
  ggplot() +
  geom_sf(fill = "white") +
  labs(title = "Iowa Counties") +
  theme(plot.title = element_text(hjust = 0.5)) +
  geom_sf_text(aes(label = COUNTY), size = 2.25) +
  geom_sf(data = iowa.districts, color = "red")
```



Be careful when plotting multiple `sf` objects together...

```
iowa.sf %>%
  ggplot() +
  geom_sf(fill = "white") +
  labs(title = "Iowa Counties") +
  theme(plot.title = element_text(hjust = 0.5)) +
  geom_sf_text(aes(label = COUNTY), size = 2.25) +
  geom_sf(data = iowa.districts, color = "red", fill = NA)
```

Converting coordinates to `sf` objects

We can add points by latitude and longitude, but...

```
sampling_site_coords <-
  read.csv("./data/sampling_site_coordinates.csv")
```

```
iowa.sf %>%
  ggplot() +
  geom_sf(fill = "white") +
  geom_sf(data = sampling_site_coords)
```

```
## Error: stat_sf requires the following missing aesthetics: geometry
```

Use `st_as_sf` to convert lat/long coordinates to an `sf` object

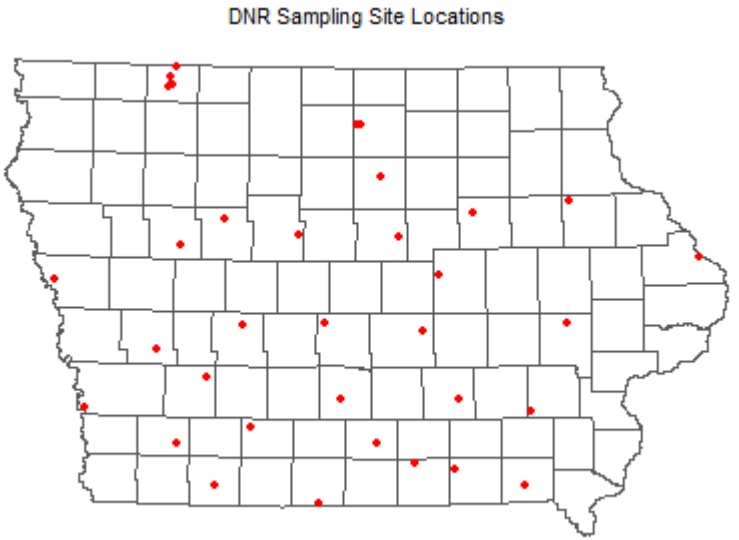
```
sampling_site_coords <- sampling_site_coords %>%
  st_as_sf(coords = c("site_longitude", "site_latitude"),
           crs = 4269)

head(sampling_site_coords)
```

```
## Simple feature collection with 6 features and 1 field
## geometry type:  POINT
## dimension:      XY
## bbox:           xmin: -95.02906 ymin: 41.79395 xmax: -91.5359 ymax: 43.125
## geographic CRS: NAD83
##            site_name                    geometry
## 1     Backbone Beach   POINT (-91.5359 42.6015)
## 2  Beed's Lake Beach POINT (-93.23654 42.77043)
## 3    Big Creek Beach POINT (-93.73185 41.79395)
## 4   Black Hawk Beach POINT (-95.02906 42.29637)
## 5 Brushy Creek Beach POINT (-93.97857 42.38858)
## 6   Clear Lake Beach POINT (-93.41469 43.12524)
```

See https://epsg.io/ for more information about coordinate reference systems. In general, mapping the coordinates to the same CRS as the shape object you're using will work out fine.

Now we're ready to plot

```
iowa.sf %>%
  ggplot() +
  geom_sf(fill = "white") +
  labs(title = "DNR Sampling Site Locations") +
  theme(plot.title = element_text(hjust = 0.5)) +
  geom_sf(data = sampling_site_coords, color = "red")
```



Preparing county population data

```
iowa.county_pops <- read.csv('data/iowa_county_pops.csv')

head(iowa.county_pops)
```

```
##      FIPS    County                    City          Year Estimate
## 1   19185     Wayne Balance of Wayne County  July 01 2011     2656
## 2 1917985 Palo Alto                Cylinder  July 01 2017       85
## 3 1981840      Linn           Walford (pt.) April 01 2010      382
## 4 1978195  Ringgold                 Tingley April 01 2010      184
## 5 1907750     Boone                 Boxholm April 01 2010      195
## 6 1934500    Shelby                  Harlan  July 01 2014     4975
##                    Primary.Point
## 1 POINT (-93.3273639 40.7394702)
## 2  POINT (-94.5511492 43.089664)
## 3 POINT (-91.8305169 41.8796623)
## 4   POINT (-94.195803 40.852747)
## 5 POINT (-94.1062028 42.1736058)
## 6 POINT (-95.3268616 41.6495154)
```

Source: https://data.iowa.gov/

Date prep

- Extract year from `Year` column
- Convert year to integer
- Rename O'Brien county to match the `iowa.sf` object.

```
iowa.county_pops <- iowa.county_pops %>%
  separate('Year',
           c(NA, NA, 'Year'),
           sep = ' ') %>%
  mutate(County = replace(County,
                          County == "O'Brien", "Obrien"),
         Year = as.integer(Year))

head(iowa.county_pops)
```

```
##      FIPS    County                    City Year Estimate
## 1   19185     Wayne Balance of Wayne County 2011     2656
## 2 1917985 Palo Alto                Cylinder 2017       85
## 3 1981840      Linn           Walford (pt.) 2010      382
## 4 1978195  Ringgold                 Tingley 2010      184
## 5 1907750     Boone                 Boxholm 2010      195
## 6 1934500    Shelby                  Harlan 2014     4975
##                     Primary.Point
## 1 POINT (-93.3273639 40.7394702)
```

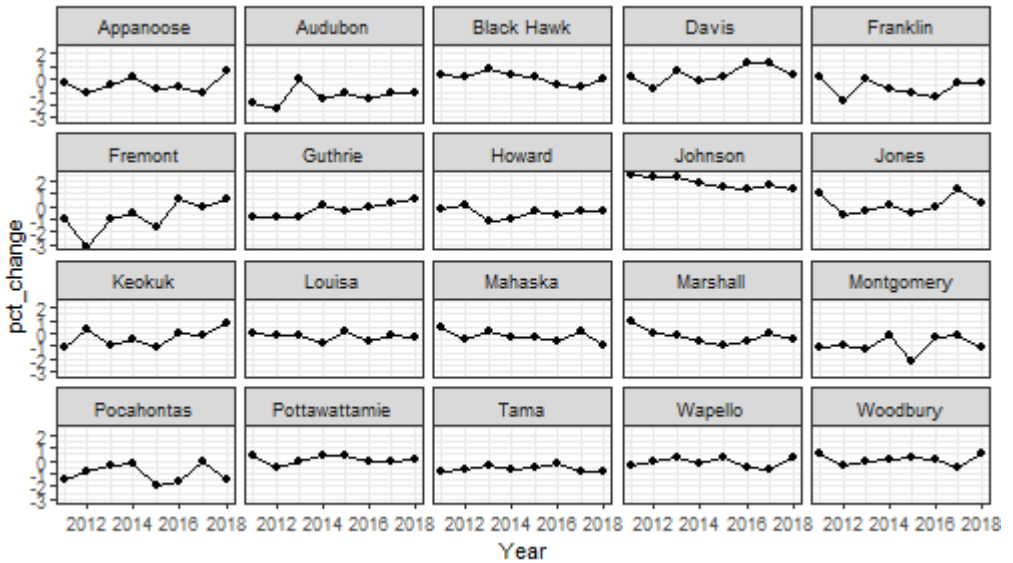
Summarize year by year change

```
iowa.county_pops.by_year <- iowa.county_pops %>%
  group_by(County, Year) %>%
  summarise(total_pop = sum(Estimate, na.rm = TRUE)) %>%
  mutate(last_year_pop = lag(total_pop)) %>%
  mutate(pct_change = (total_pop / last_year_pop - 1) * 100) %>%
  ungroup() %>%
  filter(Year > 2010)
```

```
## `summarise()` regrouping output by 'County' (override with `.groups` argum
```

Let's do a quick visualization:

```
iowa.county_pops.by_year %>%
  filter(Year > 2010,
         County %in% sample(unique(County), 20)) %>%
  ggplot(aes(Year, pct_change)) +
  geom_point() +
  geom_line() +
  theme_bw() +
  facet_wrap(~ County)
```



Working towards an animation

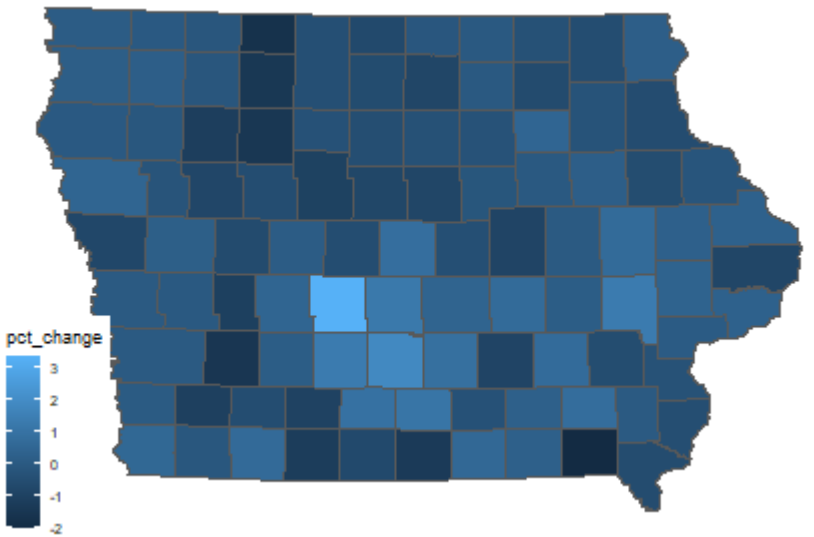
Let's combine our population data with the `iowa.sf` object:

```
iowa.joined_sf <-
  inner_join(iowa.sf,
             iowa.county_pops.by_year,
             by = c("COUNTY" = "County"))
```

Sanity check: How many rows does `iowa.joined_sf` have? How many should it have?

Let's begin by plotting one year:

```
iowa.joined_sf %>%
  filter(Year == 2018) %>%
  ggplot(aes(fill = pct_change)) +
  geom_sf()
```



Let's clean this up:

```
iowa.joined_sf %>%
  filter(Year == 2018) %>%
  ggplot(aes(fill = pct_change)) +
  geom_sf() +
  scale_fill_viridis_c(name = "% Change\nfrom prev. year") +
  labs(title = "Percentage change in Iowa county populations by year'
       subtitle = "Year: 2018") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        legend.background = element_rect(fill = NA)) +
  geom_sf_text(aes(label = COUNTY), size = 2.5)
```

Now that we have our graph mostly how we want it, let's animate it

```
library(gganimate)

iowa.joined_sf %>%
  ggplot(aes(fill = pct_change)) +
  geom_sf() +
  scale_fill_viridis_c(name = "% Change\nfrom prev. year") +
  labs(title = "Percentage change in Iowa county populations by year'
       subtitle = "Year: { current_frame }") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        legend.background = element_rect(fill = NA),
        legend.position = "right") +
  geom_sf_text(aes(label = COUNTY), size = 2.5) +
  transition_manual(Year)
```

```
## nframes and fps adjusted to match transition
```

Bonus: `geofacet`

The `geofacet` package makes it easy to make complex maps with the US map

As an example, let's plot the change in state population over the past few years on the US map.

The data:

```
state_pops <- read.csv("data/us_state_pops.csv", check.names = FALSE)
  as_tibble()

state_pops_long <- state_pops %>%
  pivot_longer(cols = `2010`:`2019`, names_to = "year", values_to = '
  mutate(year = as.integer(year)) %>%
  group_by(state) %>%
  mutate(pct_change = (pop / lag(pop) - 1) * 100) %>%
  ungroup() %>%
  filter(year > 2010)

head(state_pops_long)
```

Source: https://www.census.gov/data/datasets/time-series/demo/popest/2010s-state-total.html

A quick visualization:

```
state_pops_long %>%
  filter(state != sample(unique(state), 1)) %>% # to make faceting n
  ggplot(aes(year, pct_change, color = state)) +
  geom_point() +
  geom_line() +
  labs(x = "",
       y = "",
       title = "Yearly percent change in population by state from 20
       caption = "Source: U.S. Census Bureau, Population Division") +
  facet_wrap(~ state, ncol = 10) +
  scale_x_continuous(breaks = seq(2011, 2019, 2), expand = c(0, 0)) +
  scale_y_continuous(labels = function (x) paste0(x, "%")) +
  theme_bw() +
  geom_hline(yintercept = 0, alpha = 0.5, linetype = "dashed") +
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_blank(),
        axis.ticks.x = element_blank() )
```

```r
library(geofacet)

state_pops_long %>%
  ggplot(aes(year, pct_change, color = state)) +
  geom_point() +
  geom_line() +
  labs(x = "",
       y = "",
       title = "Yearly percent change in population by state") +
  facet_geo(~ state) +
  scale_x_continuous(breaks = seq(2011, 2019, 2)) +
  theme_minimal() +
  geom_hline(yintercept = 0, alpha = 0.5, linetype = "dashed") +
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        axis.ticks = element_blank())
```

Thanks for your time! Have a nice day!

- Email me at pev@iastate.edu if you want to talk about coding, visualizations, Vim, or Teamfight Tactics :)