

Introduction to the 2018 election data

INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R



Adam Loy

Statistician, Carleton College

The 2018 election

- 435 seats in the House of Representatives
- 35 seats in the Senate
- 36 governors

Democrats

- Recoup losses
- Energize voters
- Take control of legislative branch

Republicans

- Keep voters engaged
- Strengthen their position
- Maintain momentum

Voter turnout

McDonald, Michael P. 2018. "2018 November General Election Turnout Rates" United States Elections Project. Accessed Nov. 12, 2018.

```
glimpse(turnout)
```

```
Rows: 51
Columns: 7
$ state      <fct> Alabama, Alaska, Arizona, Arkansas, California, ...
$ state.abbr <fct> AL, AK, AZ, AR, CA, CO, CT, DE, DC, FL, GA, HI, ...
$ turnout2018 <dbl> 0.474, 0.537, 0.486, 0.412, 0.478, 0.619, 0.526, ...
$ turnout2014 <dbl> 0.332, 0.548, 0.341, 0.403, 0.307, 0.547, 0.425, ...
$ ballots    <int> 1725000, 280000, 2385000, 895000, 12250000, 2540...
$ vep        <int> 3641209, 521777, 4910625, 2171940, 25635139, 410...
$ vap        <int> 3802714, 554426, 5519036, 2319740, 30836229, 444...
```

Fundraising

```
glimpse(fundraising)
```

```
Rows: 2,412
Columns: 13
$ name      <fct> CUMMINGS, ELIJAH E, BROWN, ALVIN, ANGLE, SHARRON E, S...
$ id        <fct> H6MD07160, H8FL05140, H6NV02172, H2IL20042, H2PA15099...
$ office    <fct> H, H, H, H, H, H, H, H, H, H, H, H, H, H, H, H, H, H, H...
$ state     <fct> MD, FL, NV, IL, PA, NJ, NH, AZ, KS, FL, MT, NC, CA, N...
...
$ receipts  <dbl> 944365.35, 450694.05, 52390.03, 1890067.19, 3926.00, ...
$ disbursement <dbl> 1051584.48, 447395.51, 34936.59, 1567978.53, 4705.16, ...
$ cash.on.hand <dbl> 847733.69, 3298.54, 17568.45, 1158378.38, 64.36, -299...
$ debt.owed <dbl> 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 314610.47, 0.00, ...
$ end.date  <fct> 10/17/18, 9/30/18, 9/30/18, 10/17/18, 9/30/18, 10/19/...
$ start.date <fct> 1/1/17, 1/1/18, 10/1/17, 1/1/17, 1/1/17, 2/18/17, 4/1...
```

Congressional results

```
glimpse(senate_winners)
```

```
Rows: 33
```

```
Columns: 7
```

```
$ name      <fct> SINEMA, KYRSTEN, FEINSTEIN, DIANNE, MURPHY, CHRISTOPH...  
$ id        <fct> S8AZ00197, S0CA00199, S2CT00132, S8DE00079, S8FL00273...  
$ state     <fct> AZ, CA, CT, DE, FL, HI, IN, MA, MD, ME, MI, MN, MO, M...  
$ party     <fct> DEM, DEM, DEM, DEM, REP, DEM, REP, DEM, DEM, IND, DEM...  
$ incumbent <fct> OPEN, INCUMBENT, INCUMBENT, INCUMBENT, CHALLENGER, IN...  
$ votes     <int> 938976, 4777661, 818614, 217358, 4097689, 276133, 115...  
$ pct.vote  <dbl> 48.87364, 54.44503, 59.42544, 61.31725, 50.07701, 71....
```

Let's explore

INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R

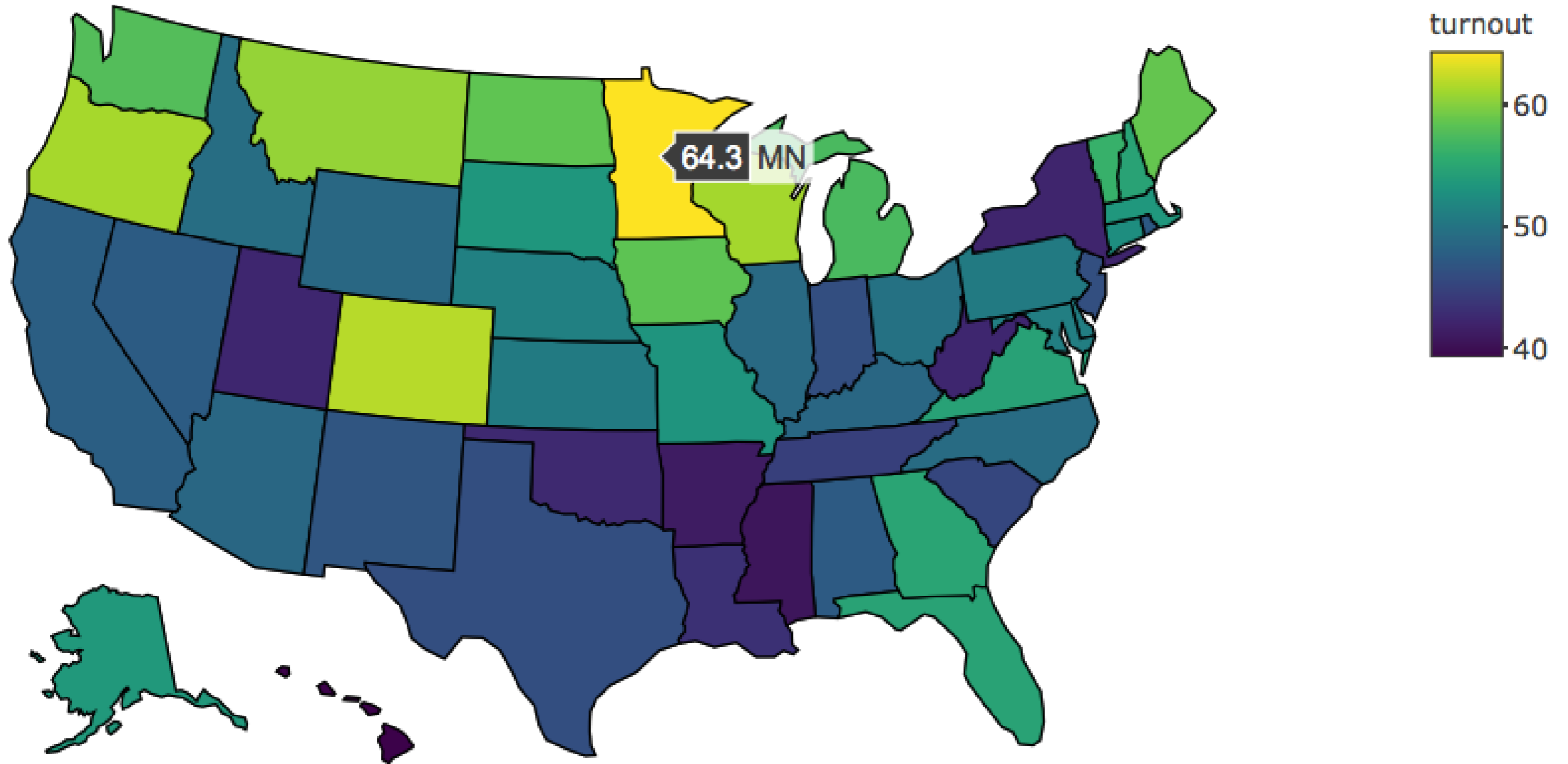
Choropleth maps

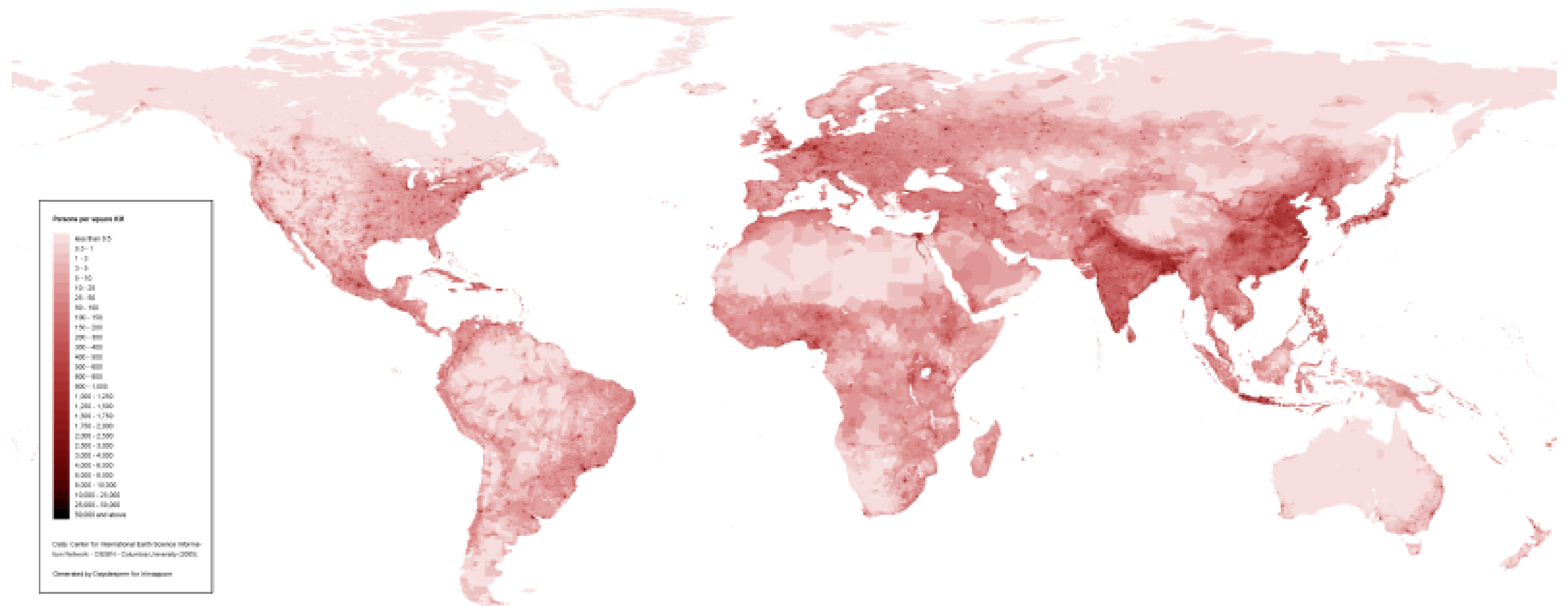
INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R



Adam Loy

Statistician, Carleton College





Choropleth maps in plotly

```
head(turnout)
```

```
   state state.abbr turnout2018 turnout2014 ballots      vep      vap
1  Alabama        AL      0.474      0.332 1725000 3641209 3802714
2   Alaska        AK      0.537      0.548  280000  521777  554426
3  Arizona        AZ      0.486      0.341 2385000 4910625 5519036
4 Arkansas        AR      0.412      0.403  895000 2171940 2319740
5 California      CA      0.478      0.307 12250000 25635139 30836229
6  Colorado      CO      0.619      0.547 2540000 4103903 4445013
```

Choropleth maps in plotly

```
turnout %>%  
  plot_geo(locationmode = 'USA-states') %>%  
  add_trace(  
    z = ~turnout,           # Sets the color values  
    locations = ~state.abbr # Matches cases to polygons  
  ) %>%  
  layout(geo = list(scope = 'usa')) # Restricts map only to USA
```

Limitation of plot_geo()

locationmode: "USA-states" | "ISO-3" | "country names"

Mapping options

```
scope = "usa"
```

- "world" | "usa" | "europe" | "asia" | "africa" | "north america" | "south america"

```
projection = list(type = "mercator")
```

- "conic conformal" | "mercator" | "robinson" | "stereographic" | and 18 more...

```
scale = 1
```

- Larger values = tighter zoom

```
center = list(lat = ~c.lat, lon = ~c.lon)
```

- Set `c.lat` and `c.lon` to center the map

Let's get mapping!

INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R

From polygons to maps

INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R



Adam Loy

Statistician, Carleton College

Boundaries

```
head(us_states)
```

```
   long    lat group order region
1 -87.46201 30.38968   1     1 alabama
2 -87.48493 30.37249   1     2 alabama
3 -87.52503 30.37249   1     3 alabama
4 -87.53076 30.33239   1     4 alabama
5 -87.57087 30.32665   1     5 alabama
6 -87.58806 30.32665   1     6 alabama
```


Joining data frames

```
glimpse(us_states)
```

```
Rows: 15,537
Columns: 5
$ long  <dbl> -87.46201, -87.48493, -87.52503, ...
$ lat   <dbl> 30.38968, 30.37249, 30.37249, ...
$ group <dbl> 1, 1, 1, ...
$ order <int> 1, 2, 3, ...
$ region <chr> "alabama", "alabama", "alabama", ...
```

```
glimpse(turnout)
```

```
Rows: 51
Columns: 7
$ state      <fct> Alabama, Alaska, Arizona, Ar...
$ state.abbr <fct> AL, AK, AZ, AR, ...
$ turnout2018 <dbl> 0.474, 0.537, 0.486, 0.412, ...
$ turnout2014 <dbl> 0.332, 0.548, 0.341, 0.403, ...
$ ballots    <int> 1725000, 280000, 2385000, 89...
$ vep        <int> 3641209, 521777, 4910625, 21...
$ vap        <int> 3802714, 554426, 5519036, 23...
```

Joining data frames

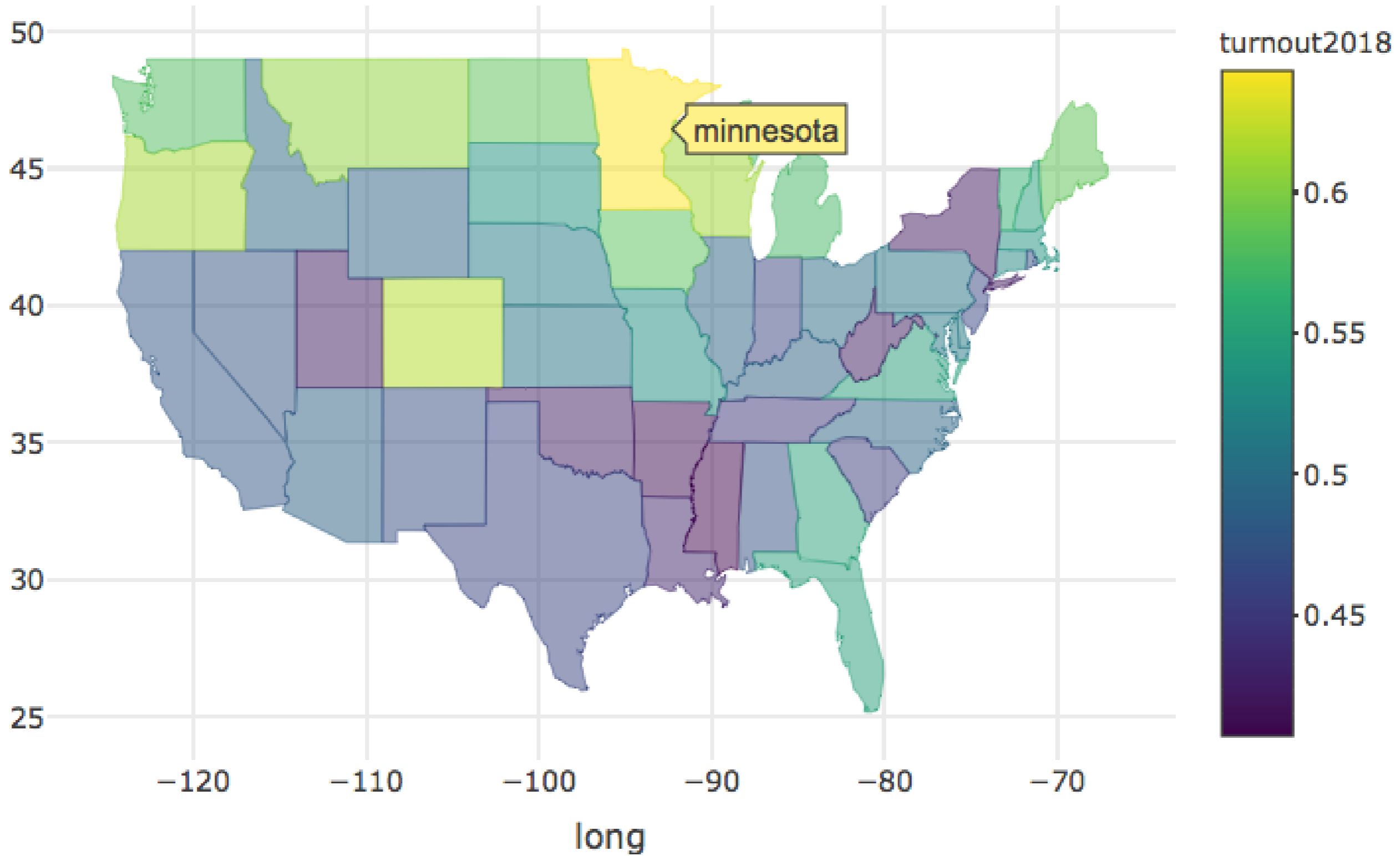
```
turnout <- turnout %>%  
  mutate(state = tolower(state)) # make state names lowercase  
states_map <- left_join(us_states, turnout, by = c("region" = "state"))
```

```
Rows: 15,537  
Columns: 11  
$ long      <dbl> -87.46201, -87.48493, -87.52503, -87.53076...  
$ lat       <dbl> 30.38968, 30.37249, 30.37249, 30.33239, 30...  
$ group     <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...  
$ order     <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13,...  
$ region    <chr> "alabama", "alabama", "alabama", "alabama"...  
$ state.abbr <fct> AL, AL, AL, AL, AL, AL, AL, AL, AL, AL, AL...  
$ turnout2018 <dbl> 0.474, 0.474, 0.474, 0.474, 0.474, 0.474, ...  
$ turnout2014 <dbl> 0.332, 0.332, 0.332, 0.332, 0.332, 0.332, ...  
...
```

Creating the map

```
states_map %>%  
  group_by(group) %>%  
  plot_ly(  
    x = ~long,  
    y = ~lat,  
    color = ~turnout2018, # variable mapped to fill color  
    split = ~region      # no more than one fill color per polygon  
  ) %>%  
  add_polygons(  
    line = list(width = 0.4),  
    showlegend = FALSE  
  )
```

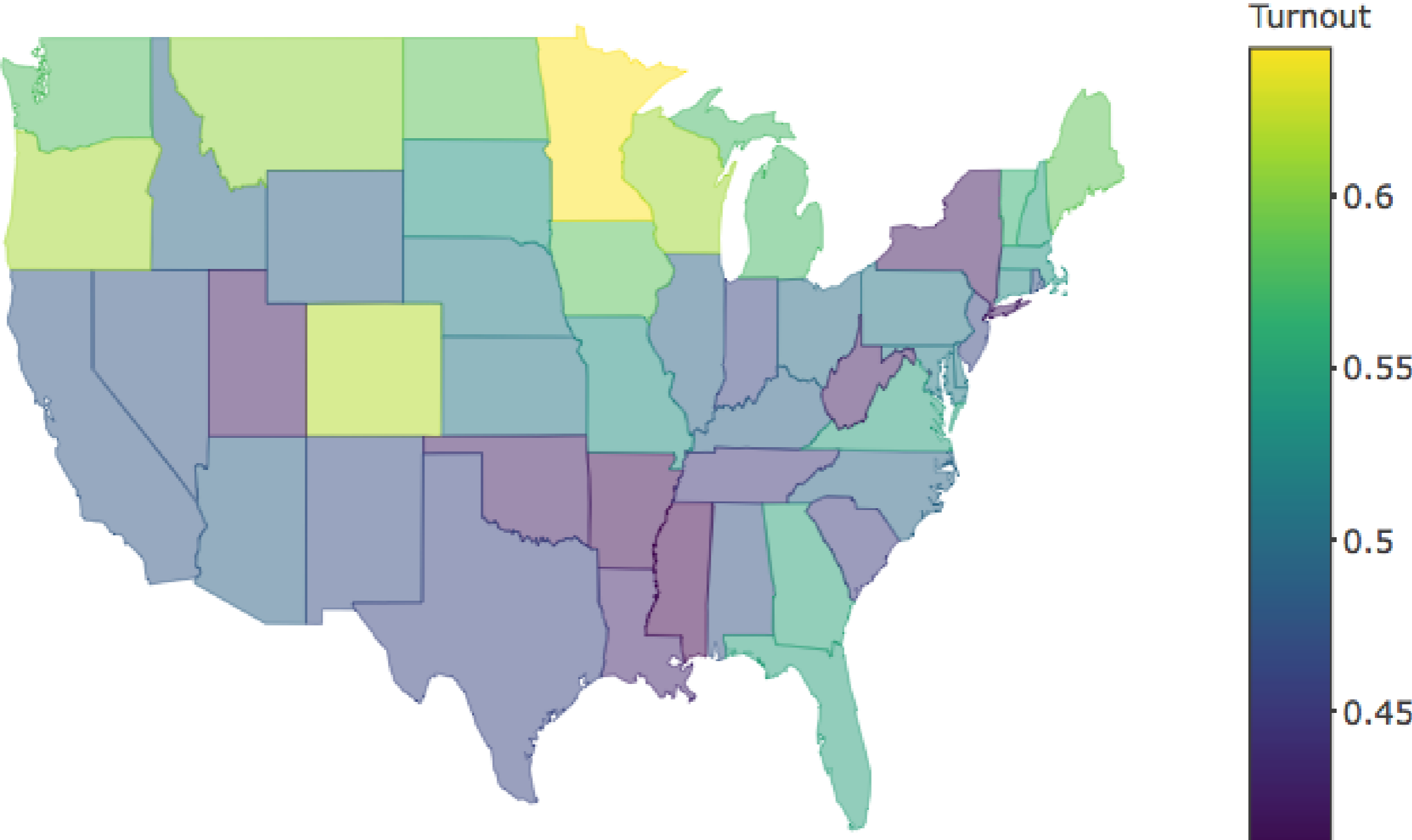
lat



Polishing your map

```
state_turnout_map %>%  
  layout(  
    title = "2018 Voter Turnout by State",  
    xaxis = list(title = "", showgrid = FALSE,  
                 zeroline = FALSE, showticklabels = FALSE),  
    yaxis = list(title = "", showgrid = FALSE,  
                 zeroline = FALSE, showticklabels = FALSE)  
  ) %>%  
  colorbar(title = "Turnout")
```

2018 Voter Turnout by State



Let's practice!

INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R

Wrap-up

INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R

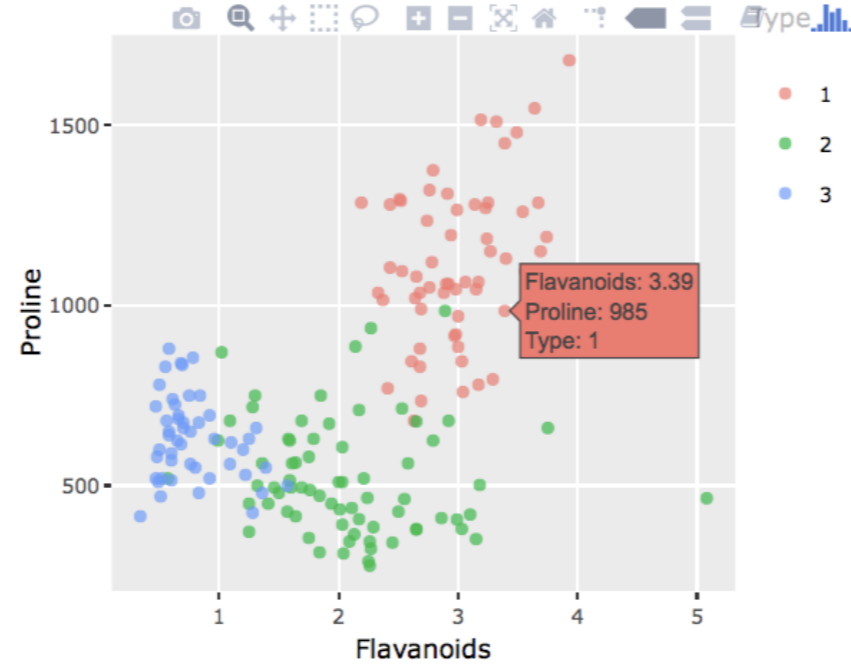


Adam Loy

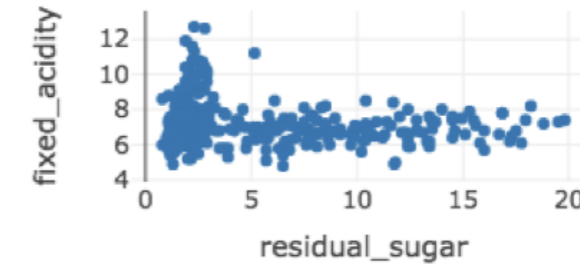
Statistician, Carleton College

Chapter 1: Displaying distributions

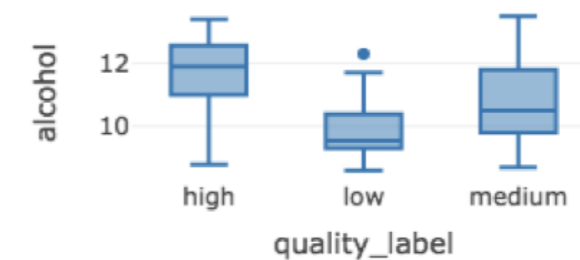
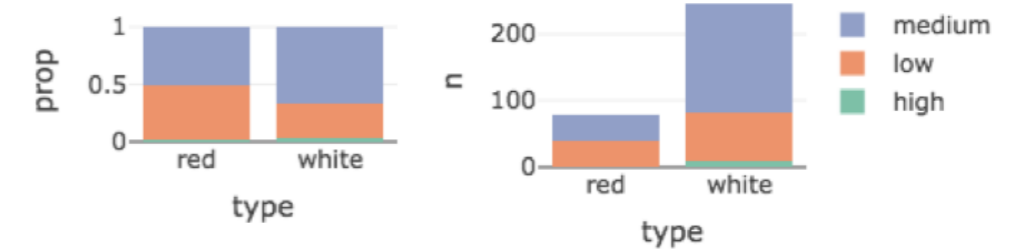
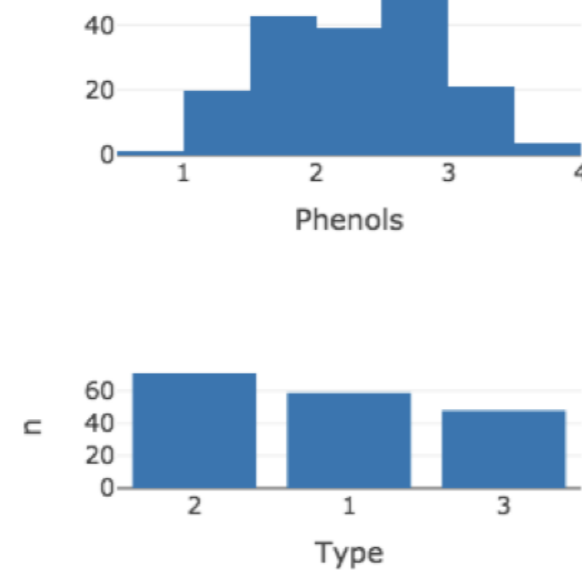
What is plotly?
Converting ggplot2
graphics



Bivariate
graphics

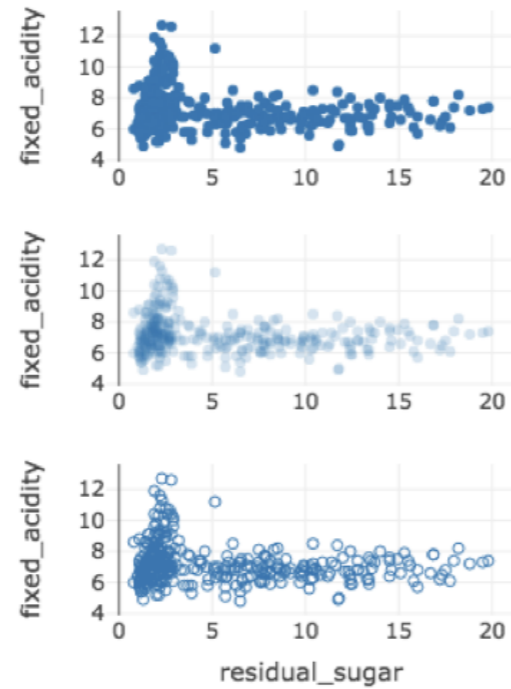


Univariate
graphics

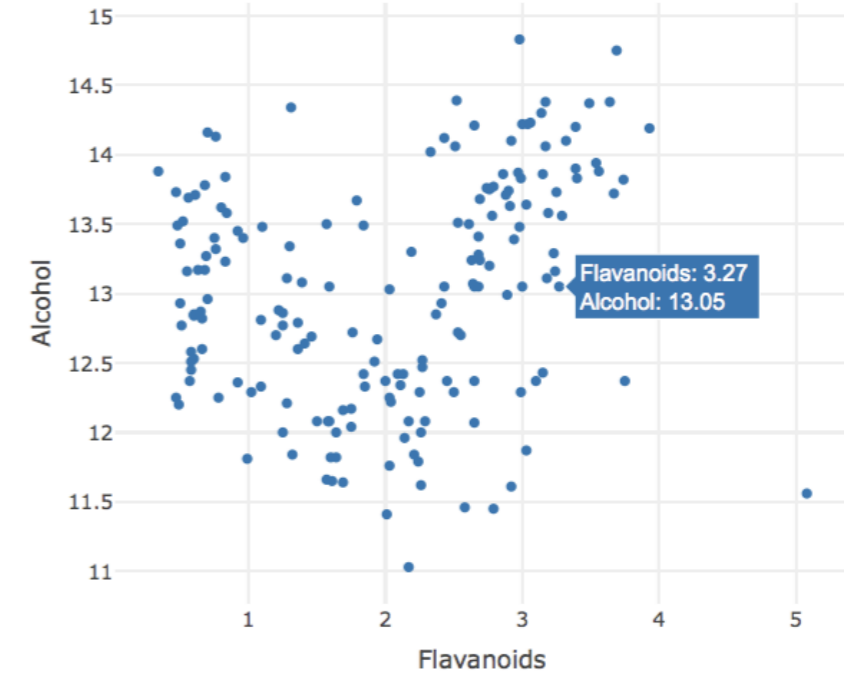


Chapter 2: Customizing your charts

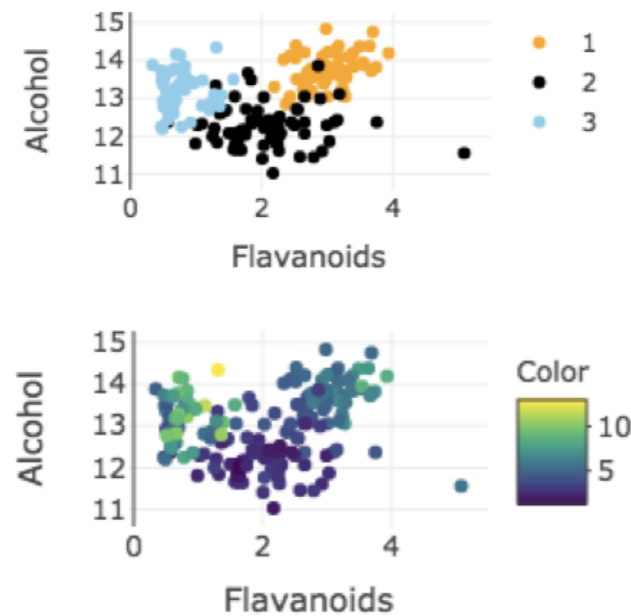
Color, opacity, symbols, and more!



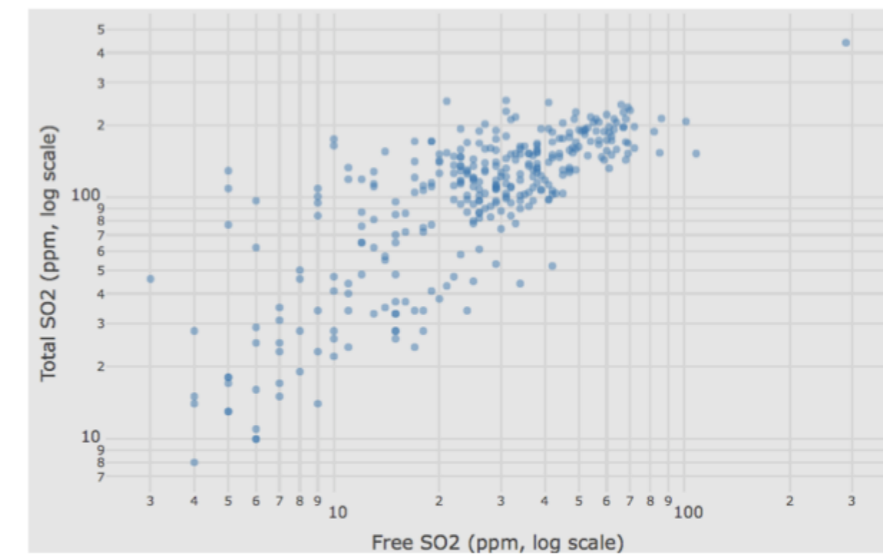
Customizing hover info



Thoughtful use of color

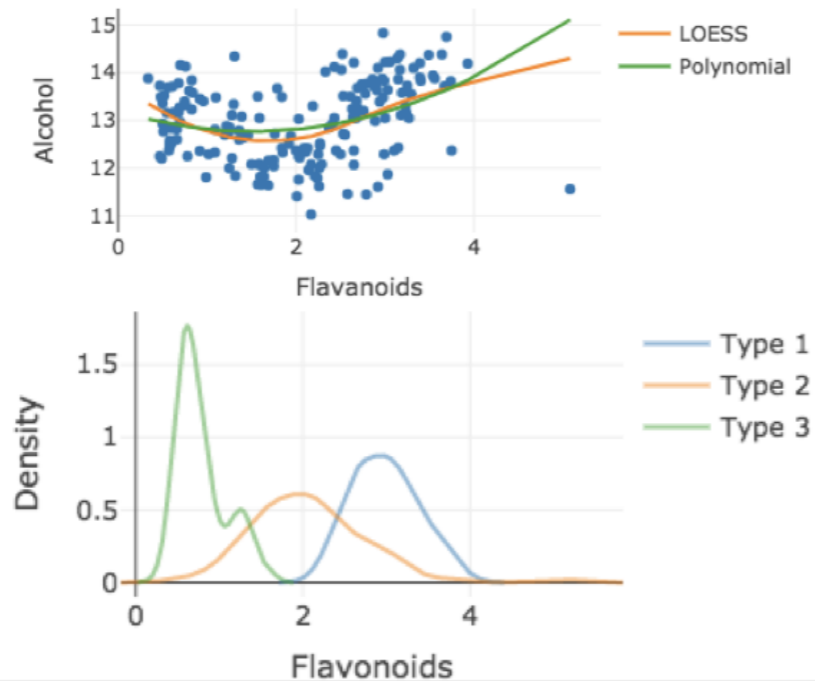


Customizing the layout

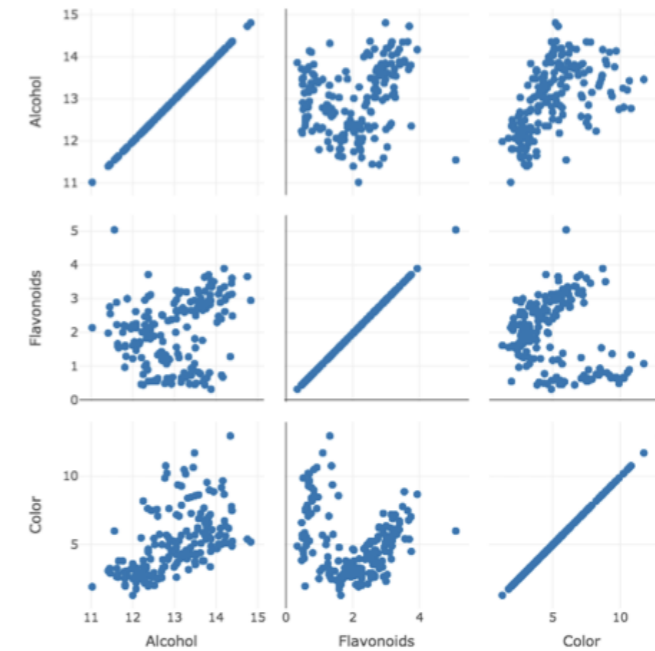


Chapter 3: Advanced charts

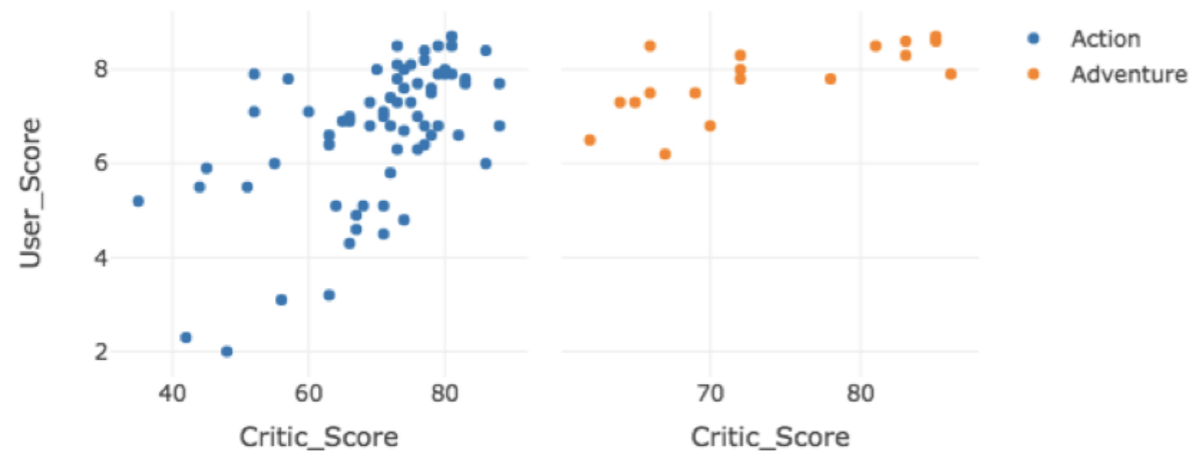
Layer traces to tell more complex stories



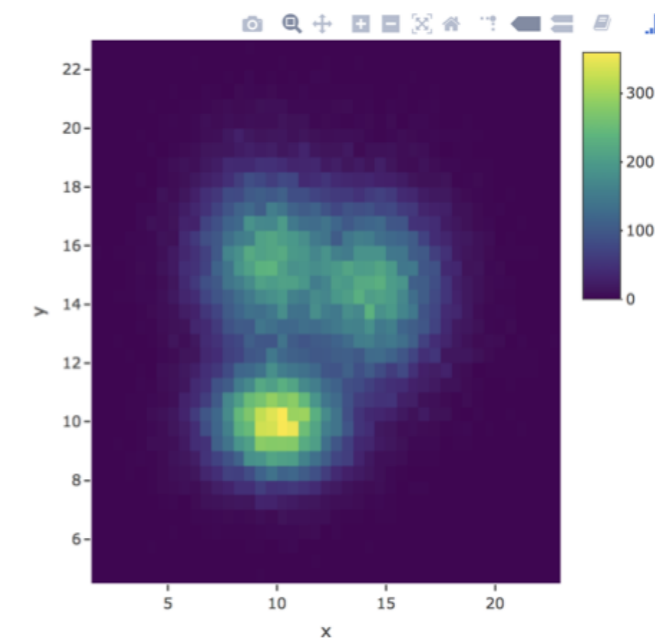
Scatterplot matrices



Subplots (i.e. faceting or small multiples)

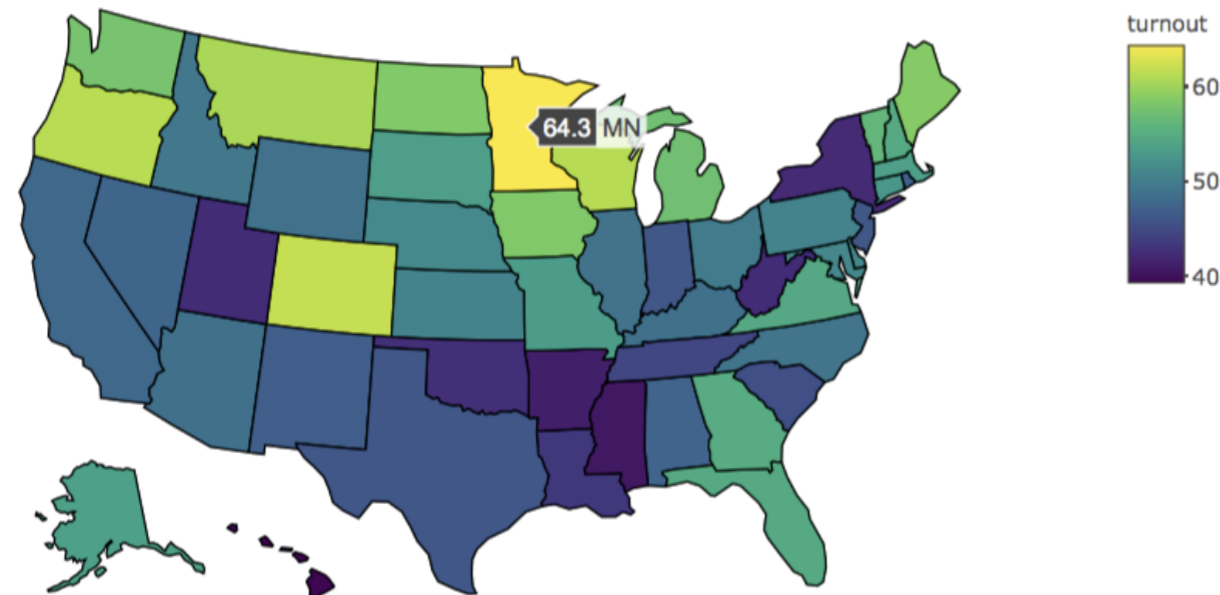


Binned scatterplots



Chapter 4: Mapping data

Choropleth maps



Points on maps



Customizing geo layouts

Going further

Books

- [Plotly for R](#) by Carson Sievert
- [Interactive Graphics for Data Analysis](#)
- [Interactive and Dynamic Graphics for Data Analysis](#)

Community

- [community.plot.ly](#) don't hesitate to ask the community a question!

Manuals

- [plot.ly/r](#): Quick examples along with in-depth documentation
- [plotly cheatsheet](#)

Forging ahead

- [Putting your charts online](#)
- Linked and/or animated graphics
- shiny + plot.ly

Thank you!

INTERACTIVE DATA VISUALIZATION WITH PLOTLY IN R