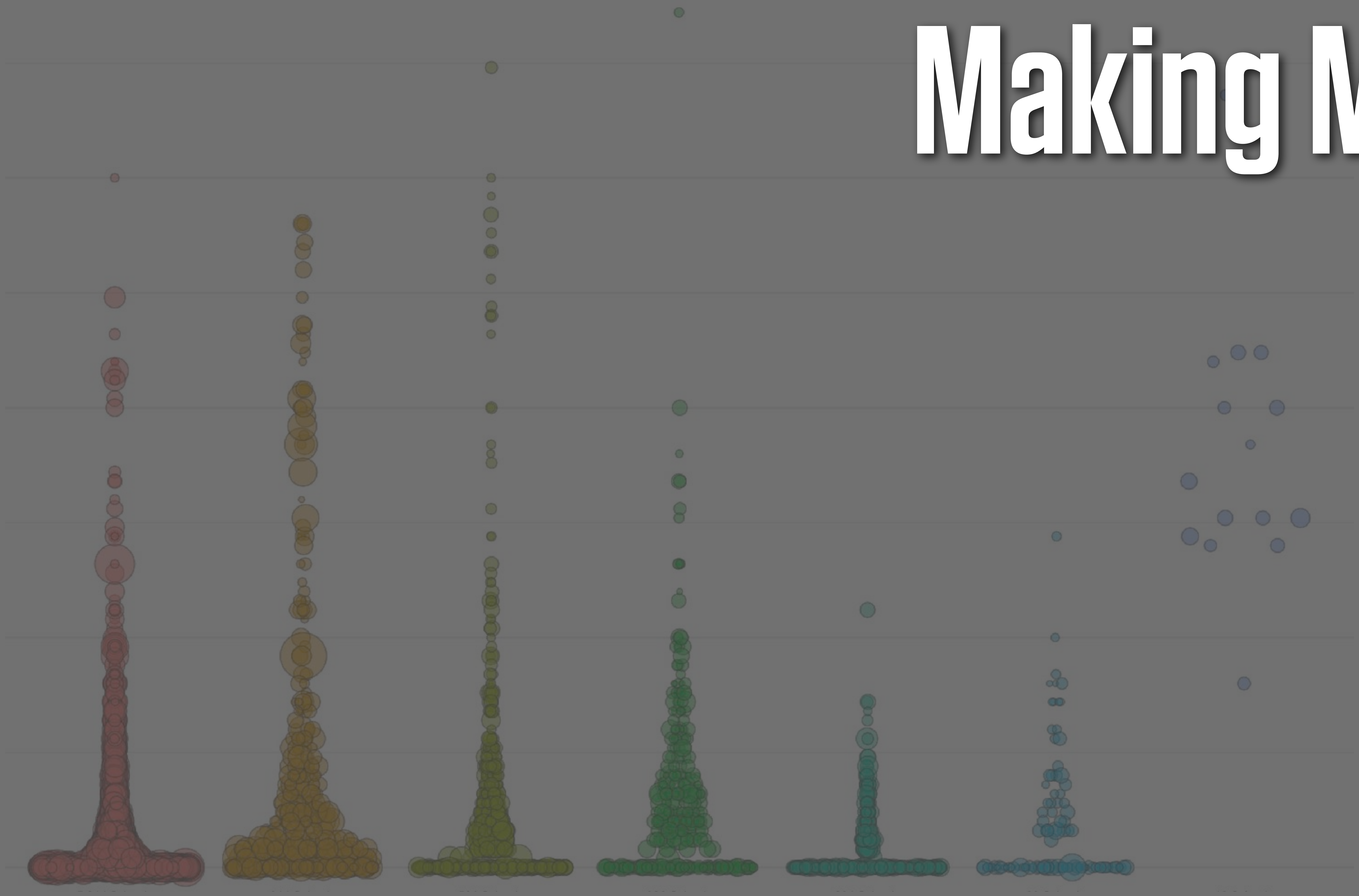
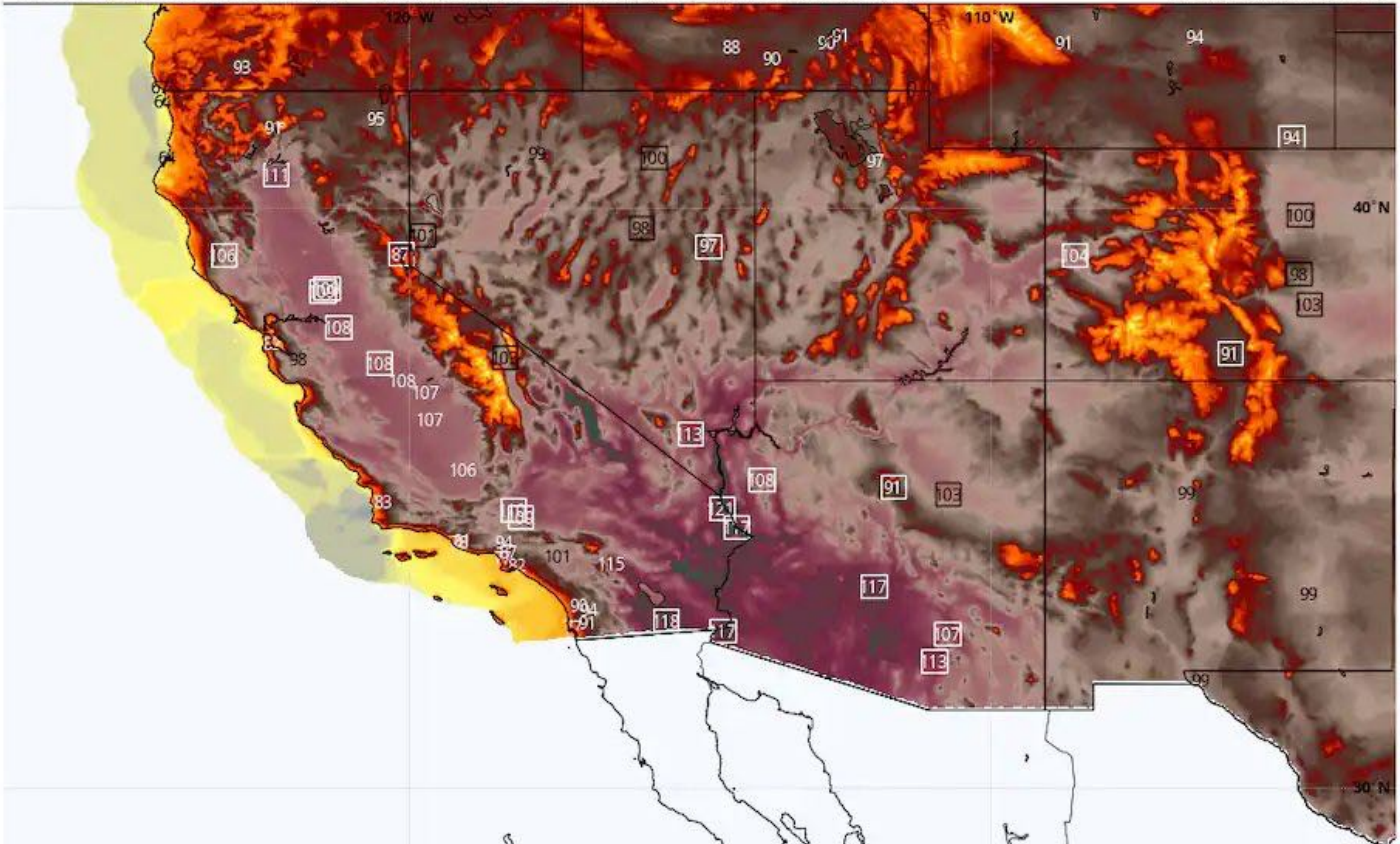
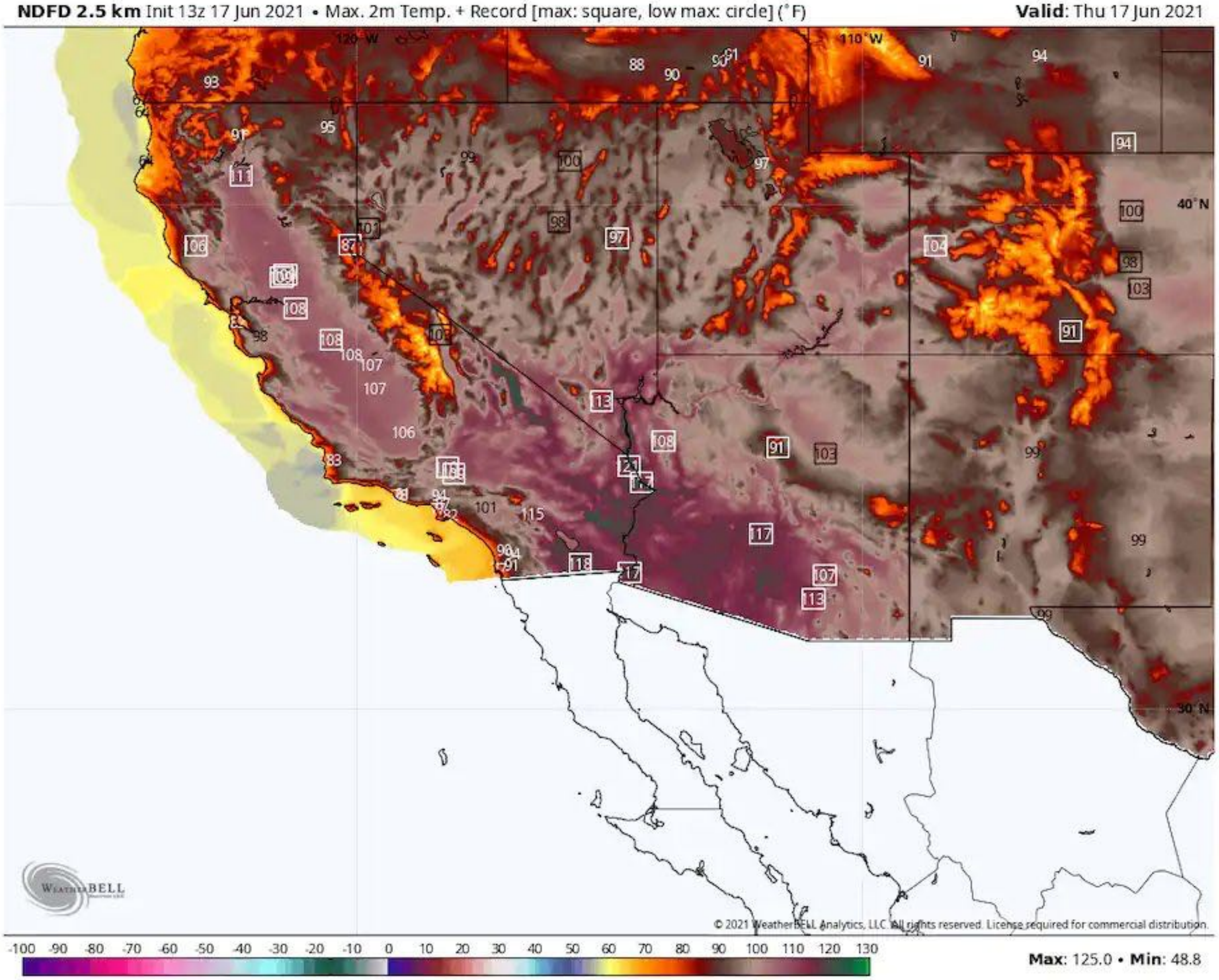

1988F5
20
40
60
80
Time since planting (days)

Making Maps

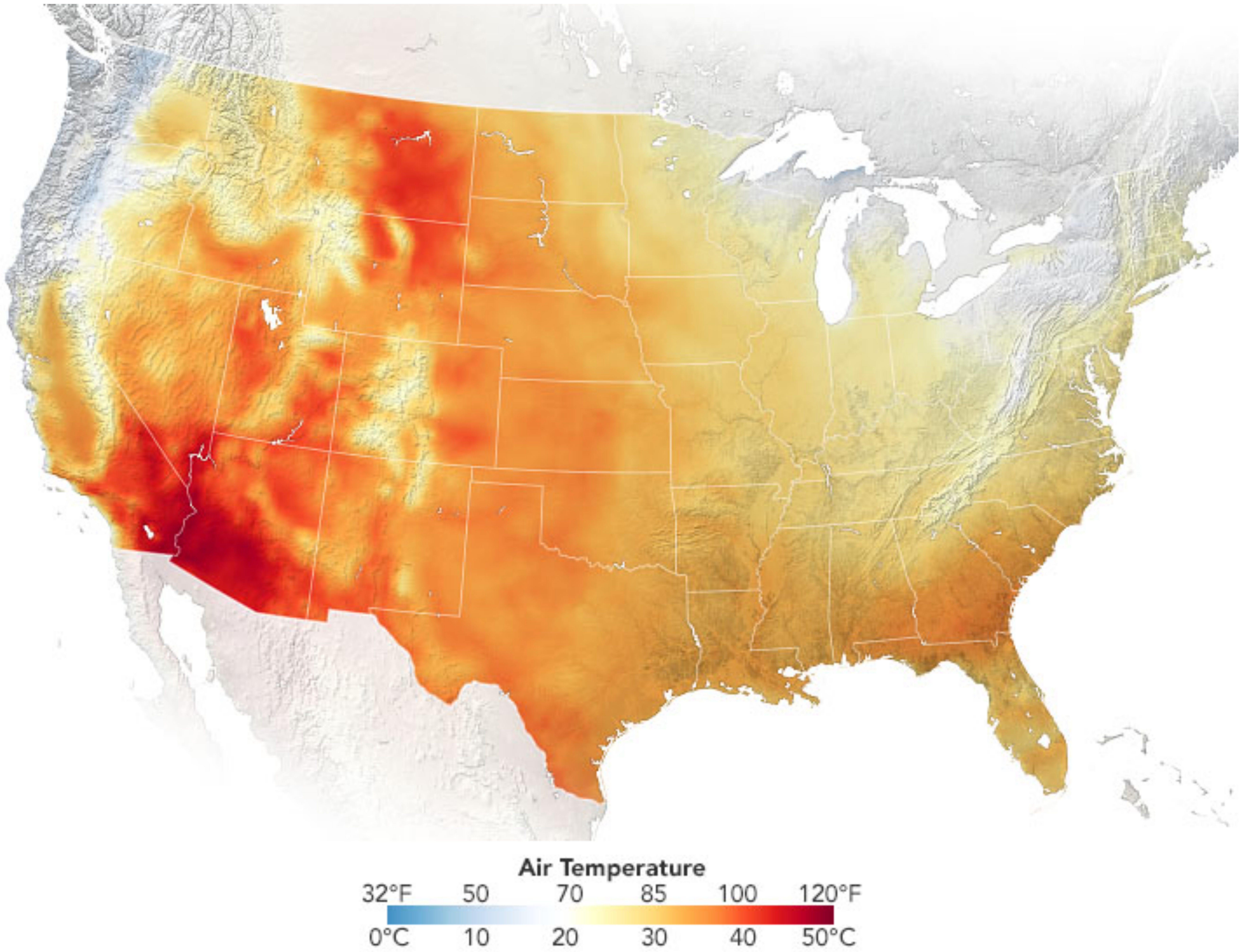


NDFD 2.5 km Init 13z 17 Jun 2021 • Max. 2m Temp. + Record [max: square, low max: circle] (°F) **Valid**: Thu 17 Jun 2021



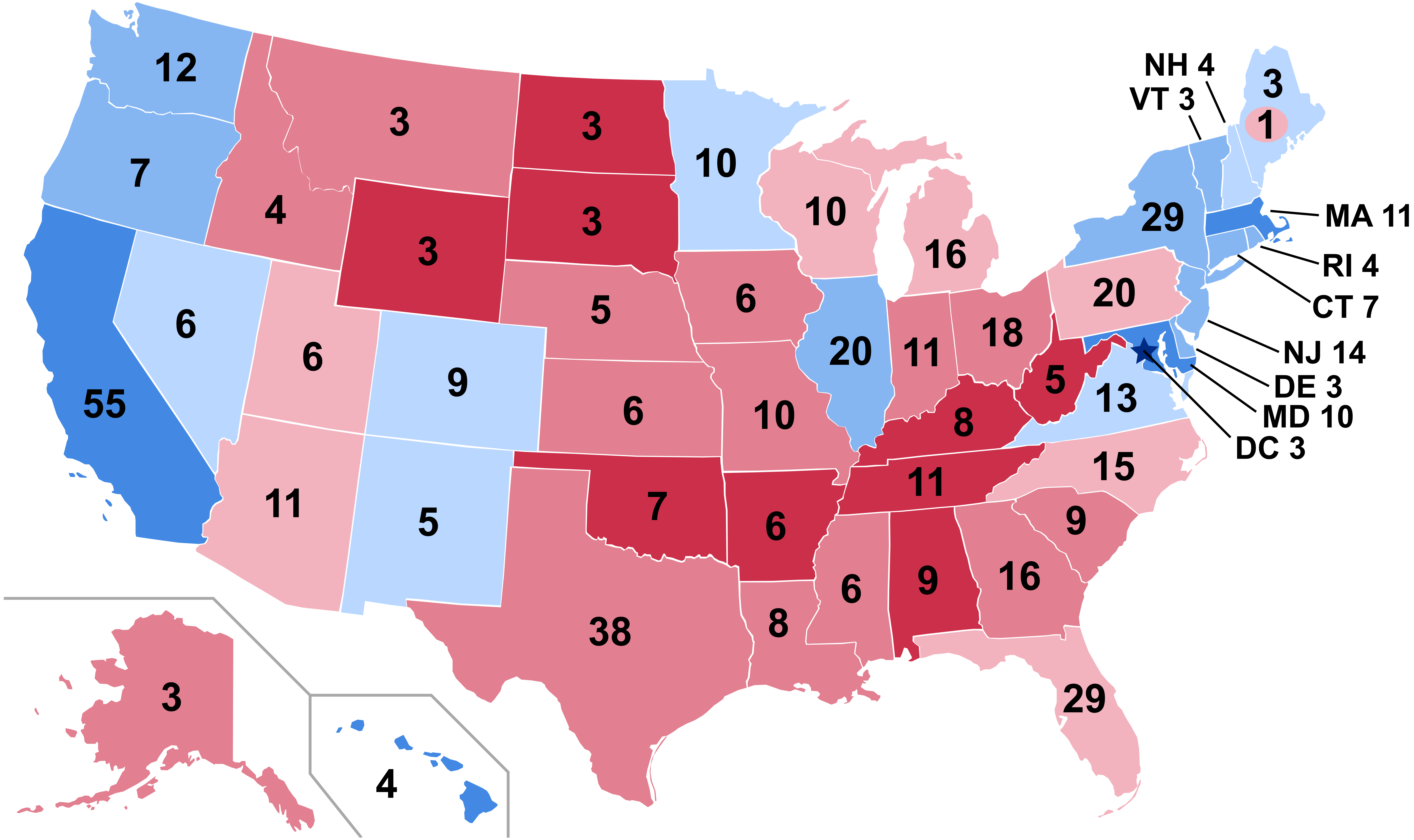


NDFD 2.5 km Init 13z 17 Jun 2021 • Max. 2m Temp. + Record [max: square, low max: circle] (°F)
Valid: Thu 17 Jun 2021
120°W
110°W
40°N
30°N
-100 -90 -80 -70 -60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 100 110 120 130
Max: 125.0 • Min: 48.8
© 2021 WeatherBELL Analytics, LLC. All rights reserved. License required for commercial distribution.
WeatherBELL

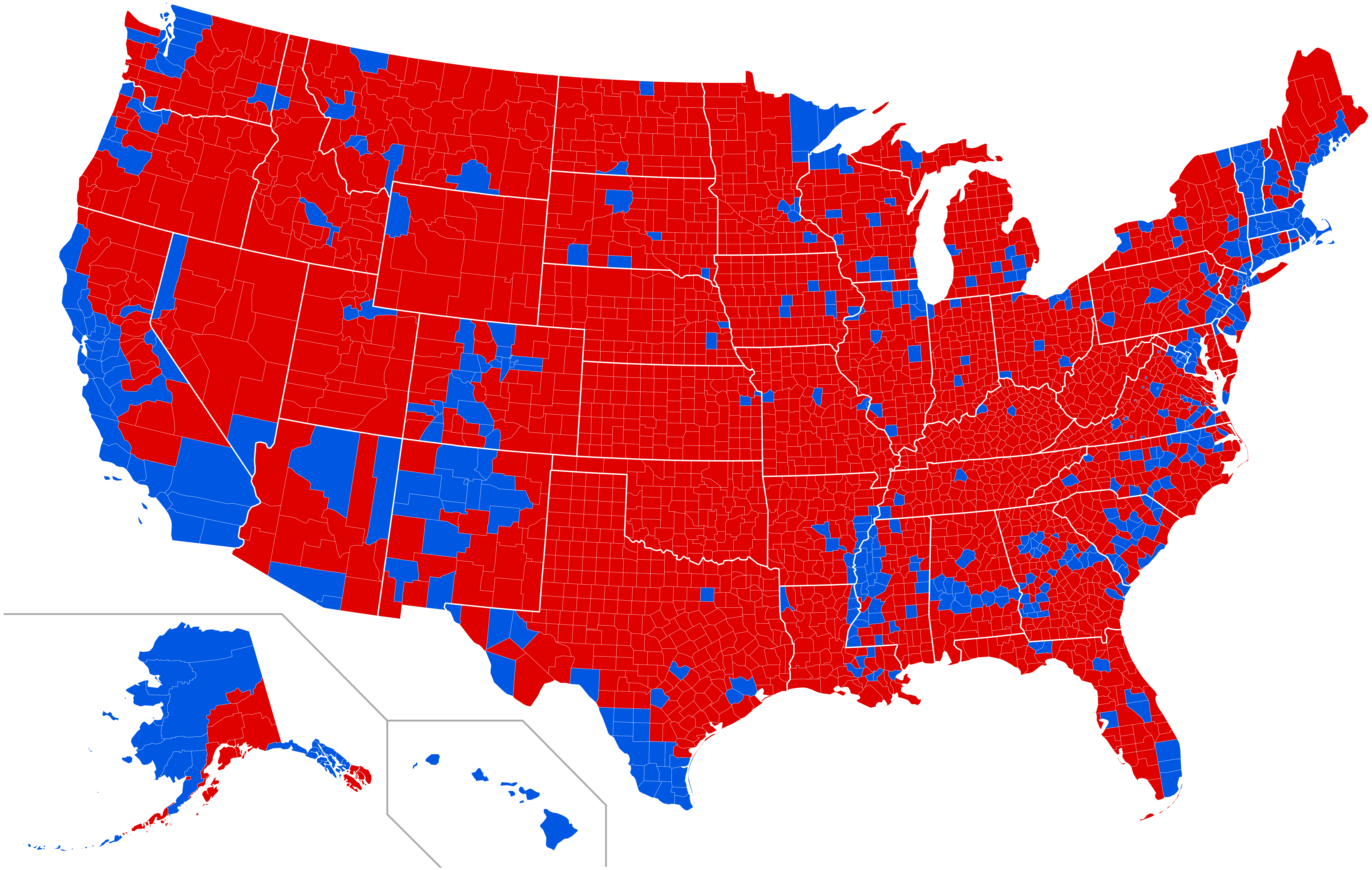


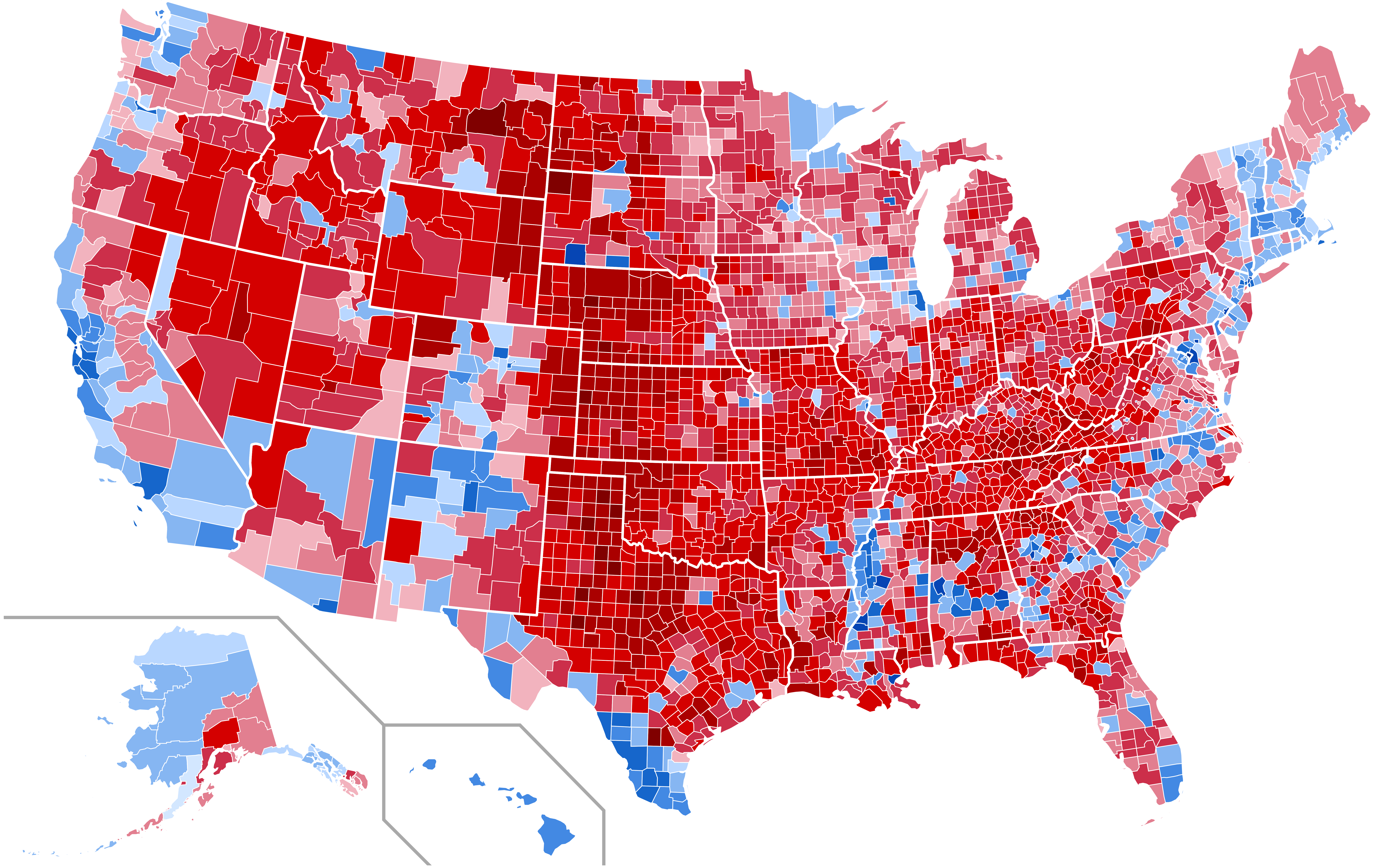
Joshua Stevens / @jscarto

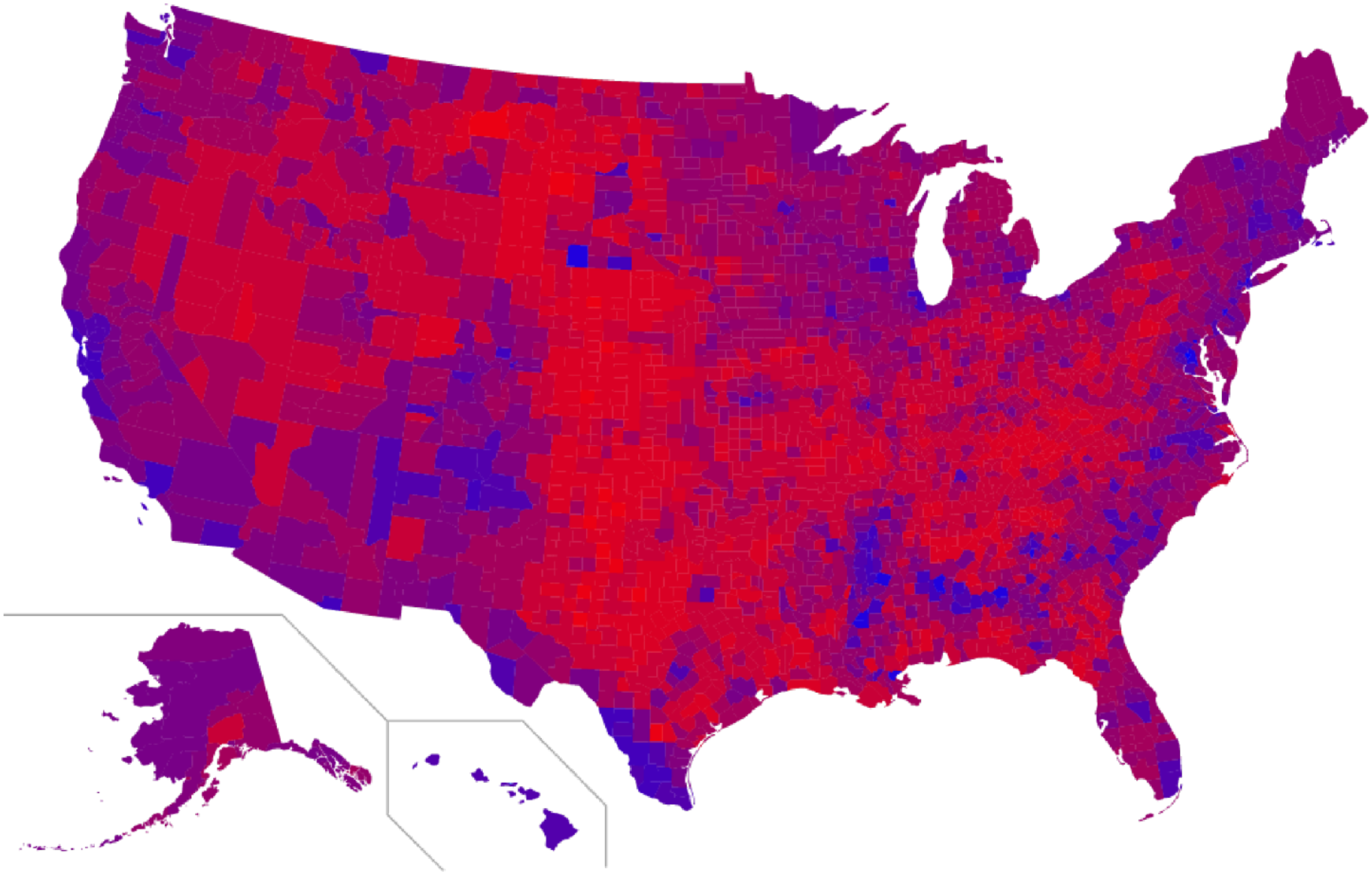
CHOROPLETHS

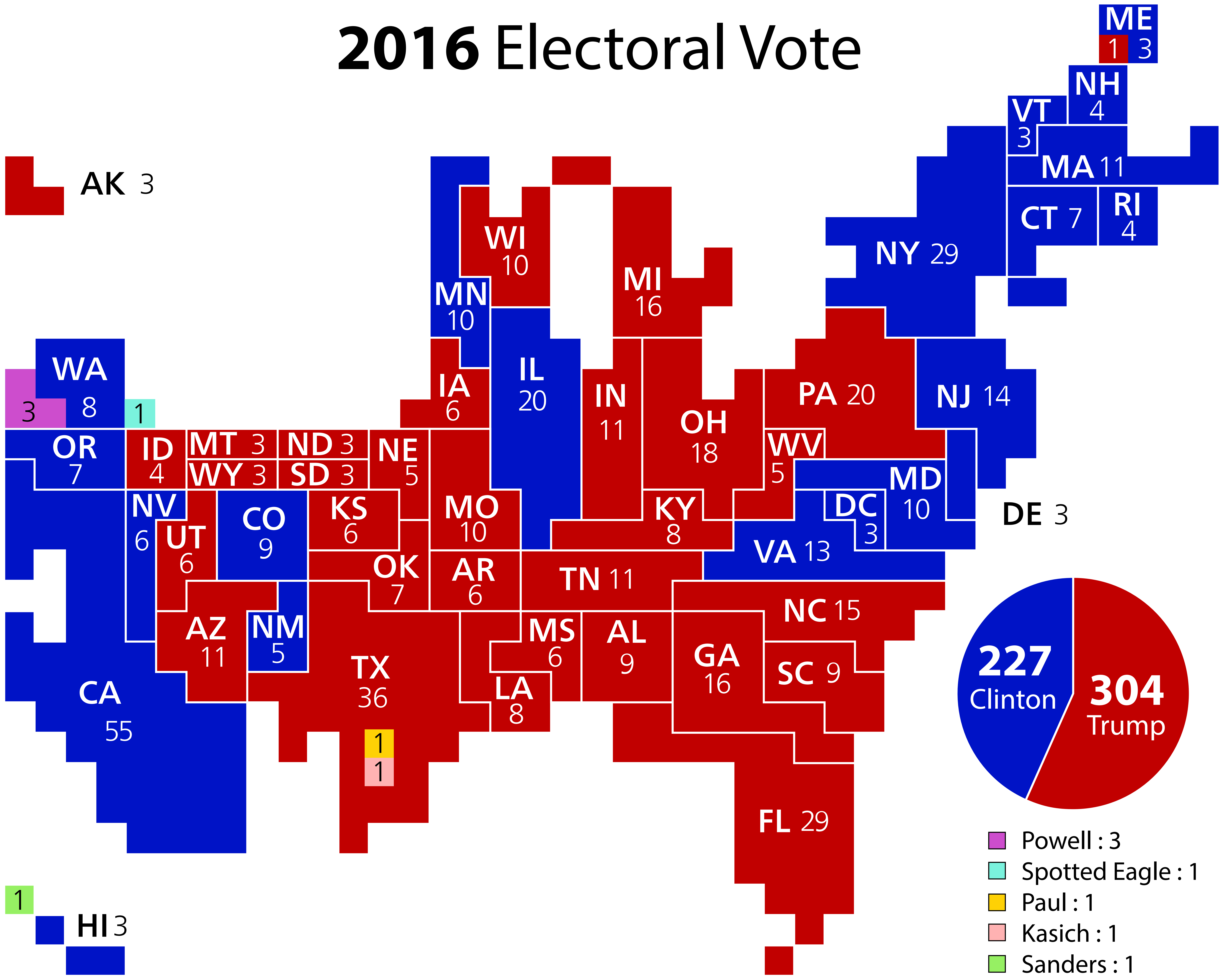


12
7
55
6
4
3
6
11
3
9
5
3
3
5
6
38
7
10
10
6
10
20
6
8
16
11
11
9
6
9
16
18
5
8
20
13
15
29
29
3
1
NH 4
VT 3
MA 11
RI 4
CT 7
NJ 14
DE 3
MD 10
DC 3
3
4

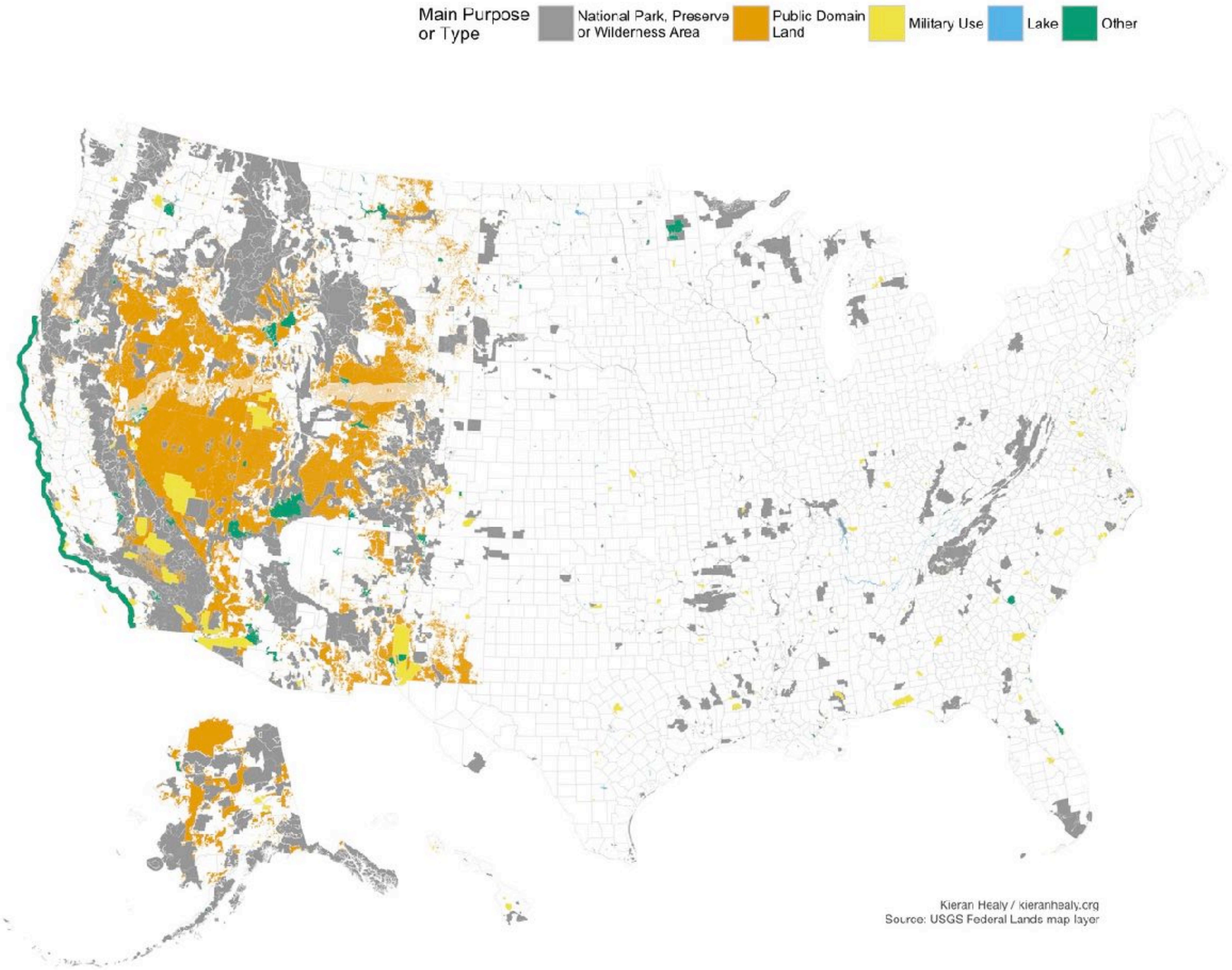




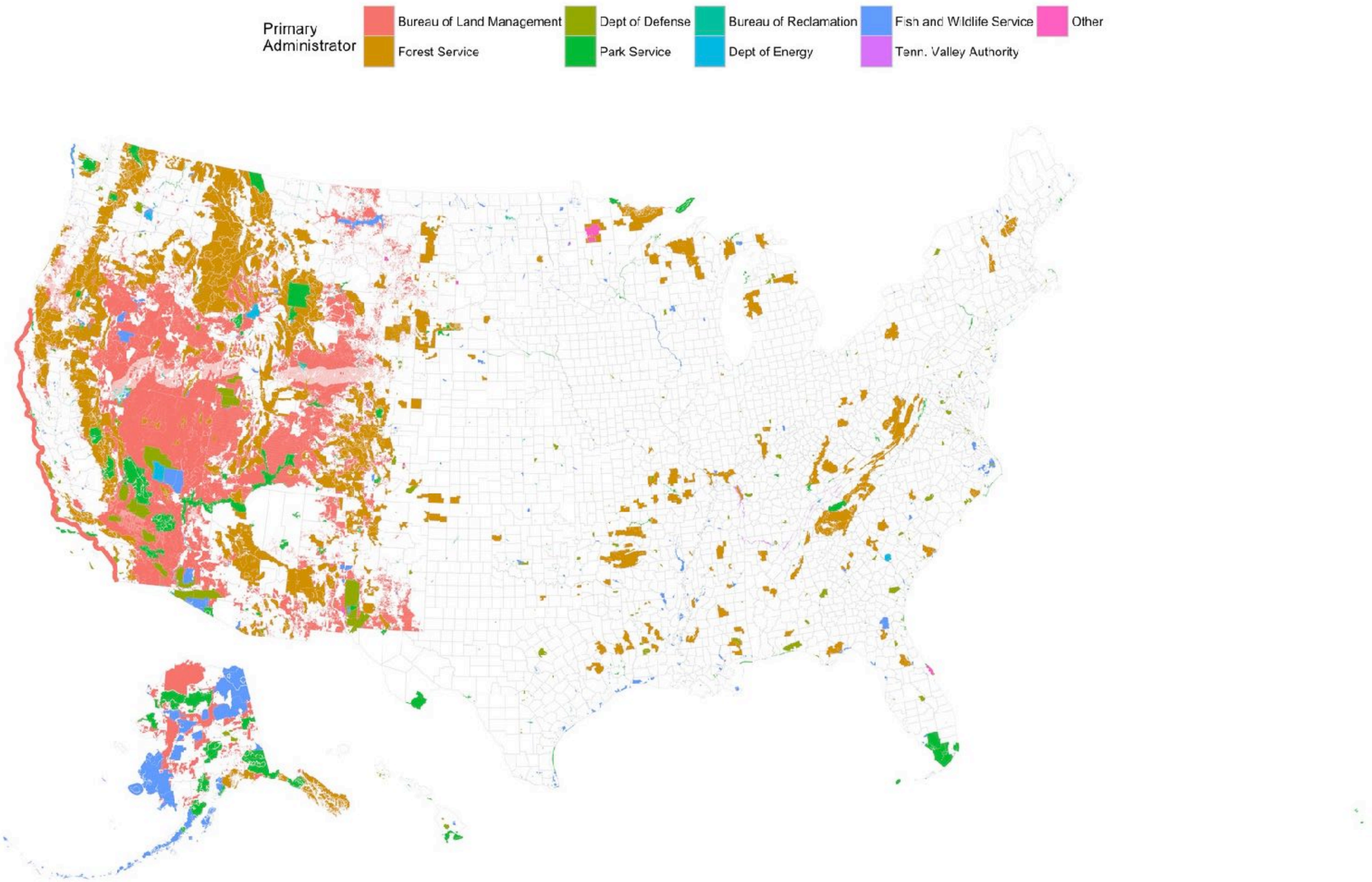



2016 Electoral Vote
ME
1
3
NH
4
VT
3
MA 11
CT 7
RI
4
NY 29
AK 3
WI
10
MN
10
MI
16
WA
8
3
1
IA
6
IL
20
IN
11
OH
18
PA 20
NJ 14
OR
7
ID
4
MT 3
ND 3
NE
5
WY 3
SD 3
WV
5
MD
10
NV
6
UT
6
CO
9
KS
6
MO
10
KY
8
DC
3
DE 3
VA 13
OK
7
AR
6
TN 11
NC 15
CA
55
AZ
11
NM
5
TX
36
MS
6
AL
9
GA
16
SC 9
LA
8
1
1
FL 29
227
Clinton
304
Trump
Powell : 3
Spotted Eagle : 1
Paul : 1
Kasich : 1
Sanders : 1
1
HI 3

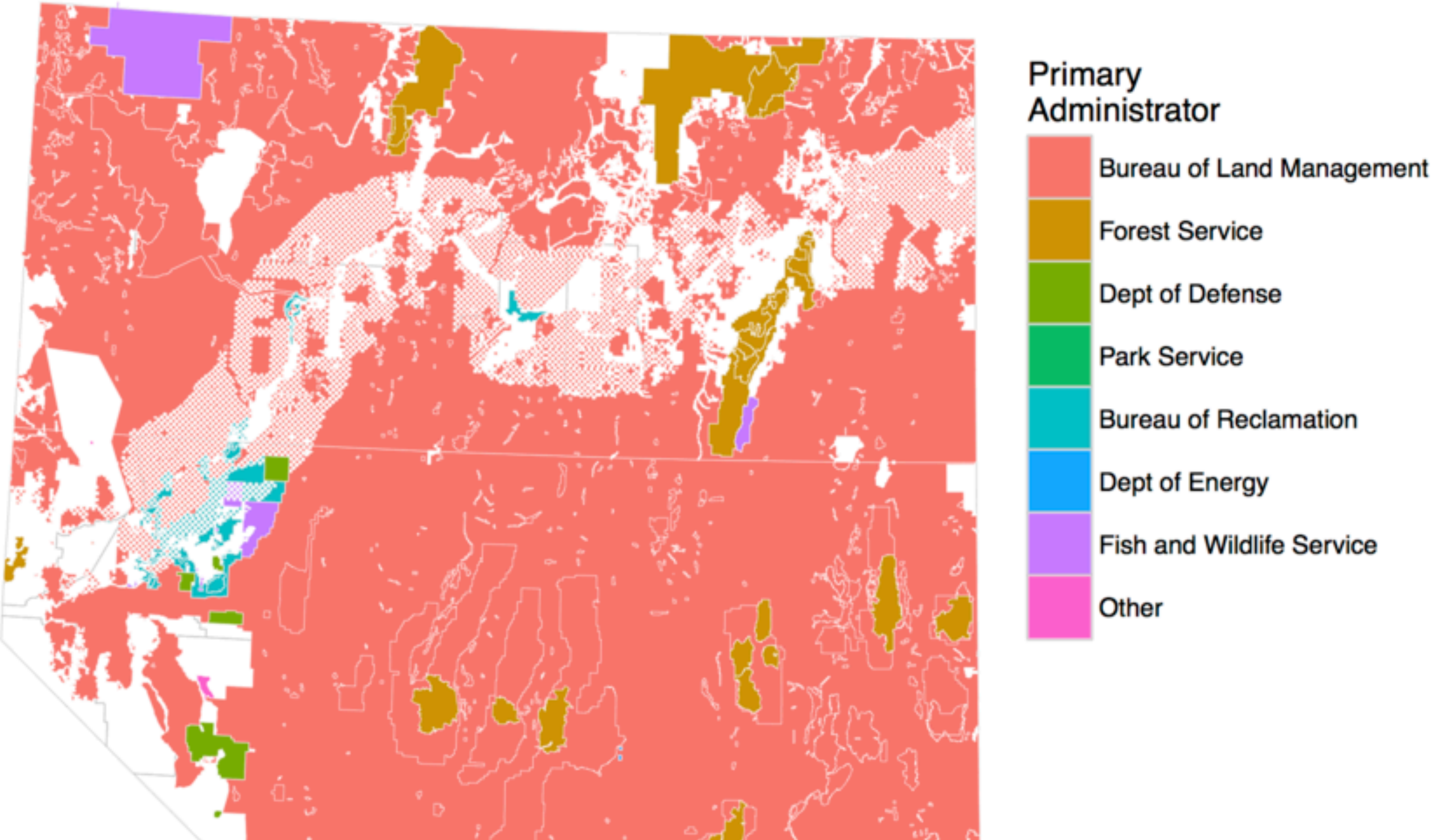
Land Owned or Administered by the US Federal Government



Land Owned or Administered by the US Federal Government



Land Owned or Administered by the US Federal Government

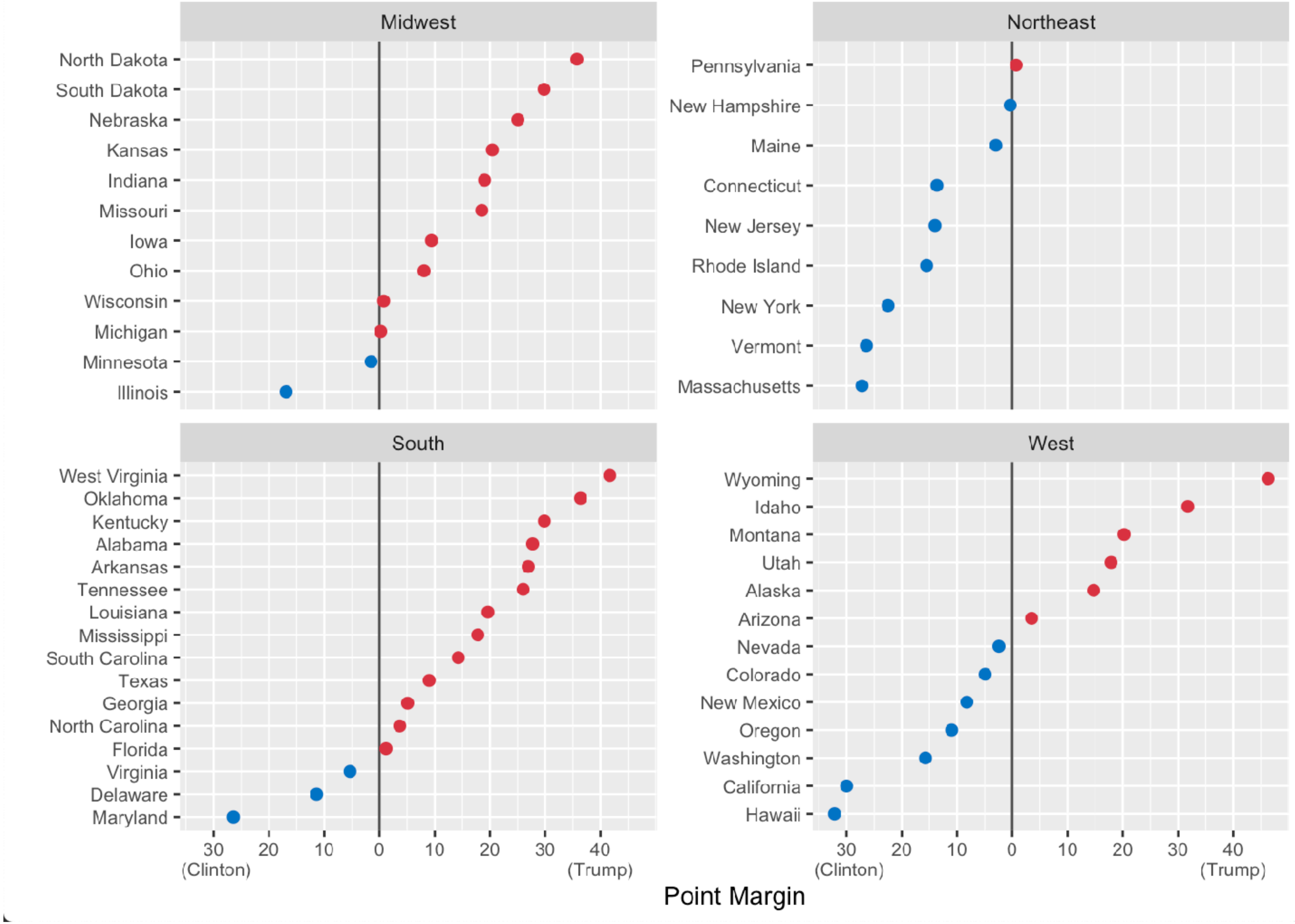



OR
ID
WY
NV
UT

US STATE-LEVEL ELECTION DATA

```r
election %>% select(state, total_vote,
                    r_points, pct_trump, party, census) %>%
    sample_n(5)

## # A tibble: 5 x 6
##          state total_vote r_points pct_trump      party    census
##          <chr>      <dbl>    <dbl>     <dbl>      <chr>     <chr>
## 1   New Jersey    3906723   -13.99      41.0   Democrat Northeast
## 2       Nevada    1125385    -2.42      45.5   Democrat      West
## 3     Virginia    3982752    -5.32      44.4   Democrat     South
## 4 Pennsylvania    6166710     0.71      48.2 Republican Northeast
## 5      Vermont     315067   -26.41      30.3   Democrat Northeast
```

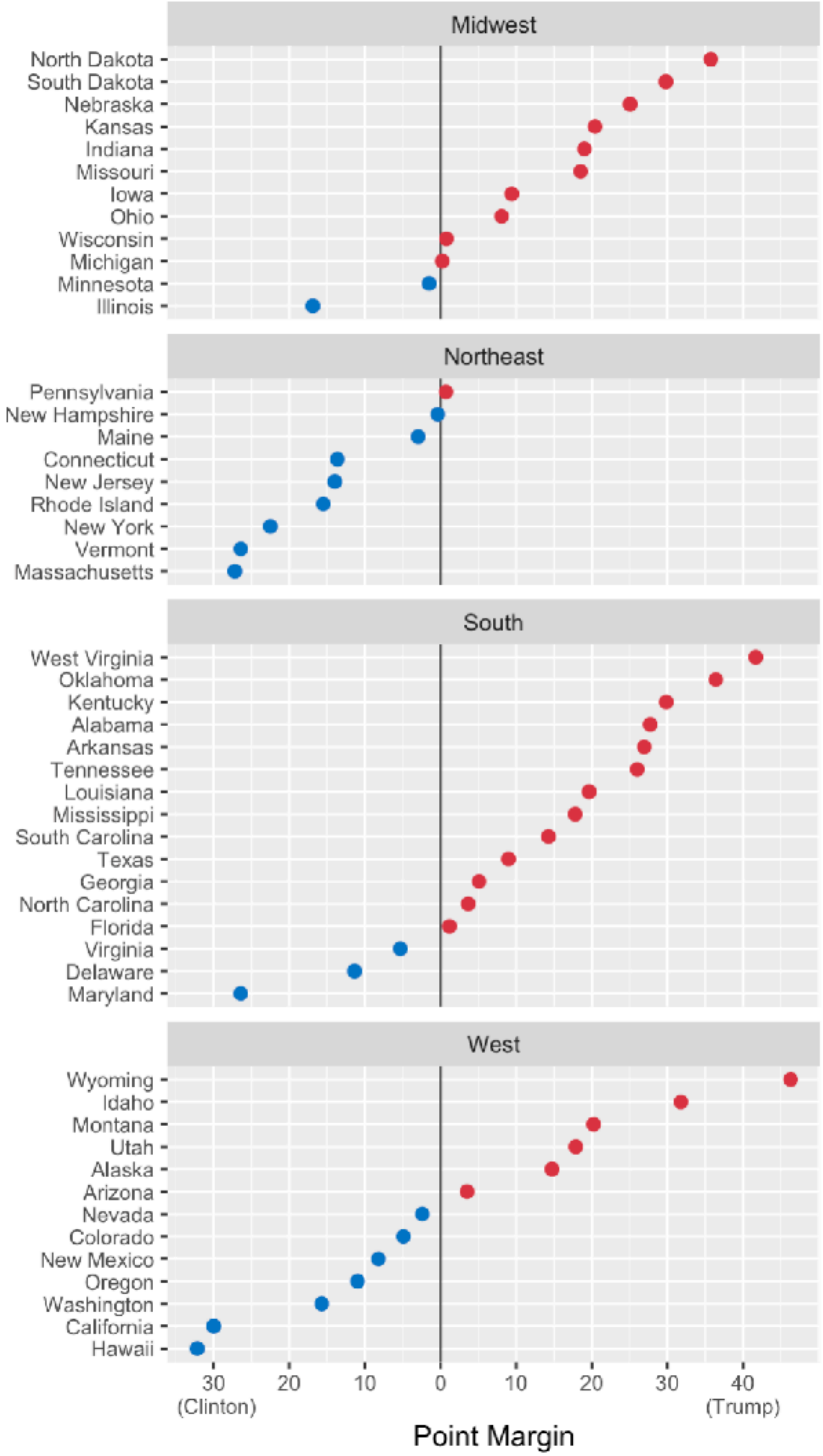
```r
## Hex color codes for Dem Blue and Rep Red
party_colors <- c("#2E74C0", "#CB454A")

p0 <- ggplot(data = subset(election, st %nin% "DC"),
             mapping = aes(x = r_points,
                           y = reorder(state, r_points),
                           color = party))

p1 <- p0 + geom_vline(xintercept = 0, color = "gray30") +
    geom_point(size = 2)

p2 <- p1 + scale_color_manual(values = party_colors)

p3 <- p2 + scale_x_continuous(breaks = c(-30, -20, -10, 0, 10, 20, 30, 40),
                              labels = c("30\n (Clinton)", "20", "10", "0",
                                         "10", "20", "30", "40\n(Trump)"))

p4 <- p3 + facet_wrap(~ census, nrow=1, scales="free_y") +
    guides(color=FALSE) + labs(x = "Point Margin", y = "") +
    theme(axis.text=element_text(size=8))
```


Midwest
North Dakota
South Dakota
Nebraska
Kansas
Indiana
Missouri
Iowa
Ohio
Wisconsin
Michigan
Minnesota
Illinois
Northeast
Pennsylvania
New Hampshire
Maine
Connecticut
New Jersey
Rhode Island
New York
Vermont
Massachusetts
South
West Virginia
Oklahoma
Kentucky
Alabama
Arkansas
Tennessee
Louisiana
Mississippi
South Carolina
Texas
Georgia
North Carolina
Florida
Virginia
Delaware
Maryland
West
Wyoming
Idaho
Montana
Utah
Alaska
Arizona
Nevada
Colorado
New Mexico
Oregon
Washington
California
Hawaii
30 20 10 0 10 20 30 40
(Clinton)
(Trump)
Point Margin

```r
library(ggforce)
```

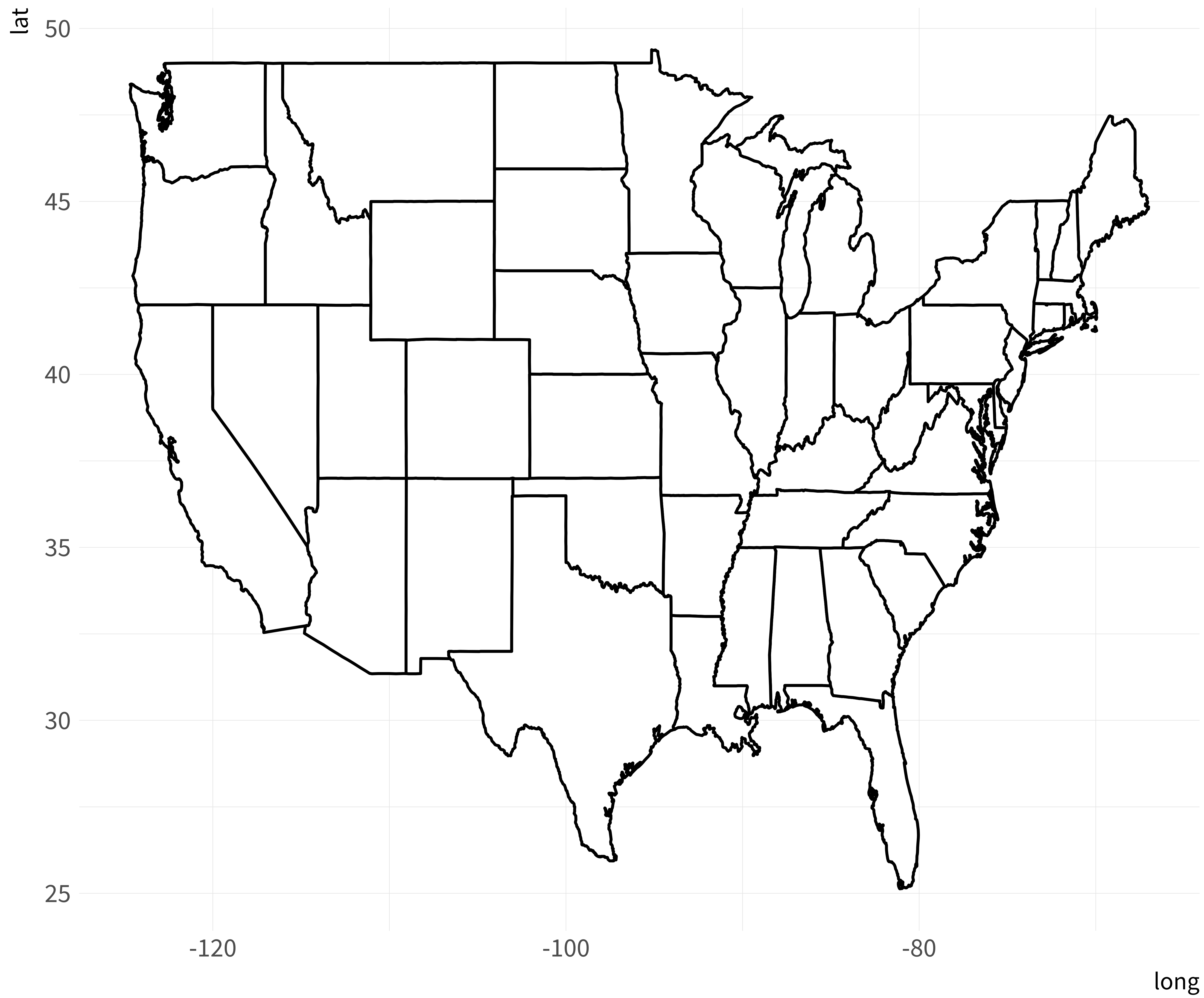
Very useful set of enhancements

```r
p3 + facet_col(~ census,
               scales = "free_y",
               space = "free") +
    guides(color=FALSE) +
    labs(x = "Point Margin", y = "") +
    theme(axis.text=element_text(size=8))
```

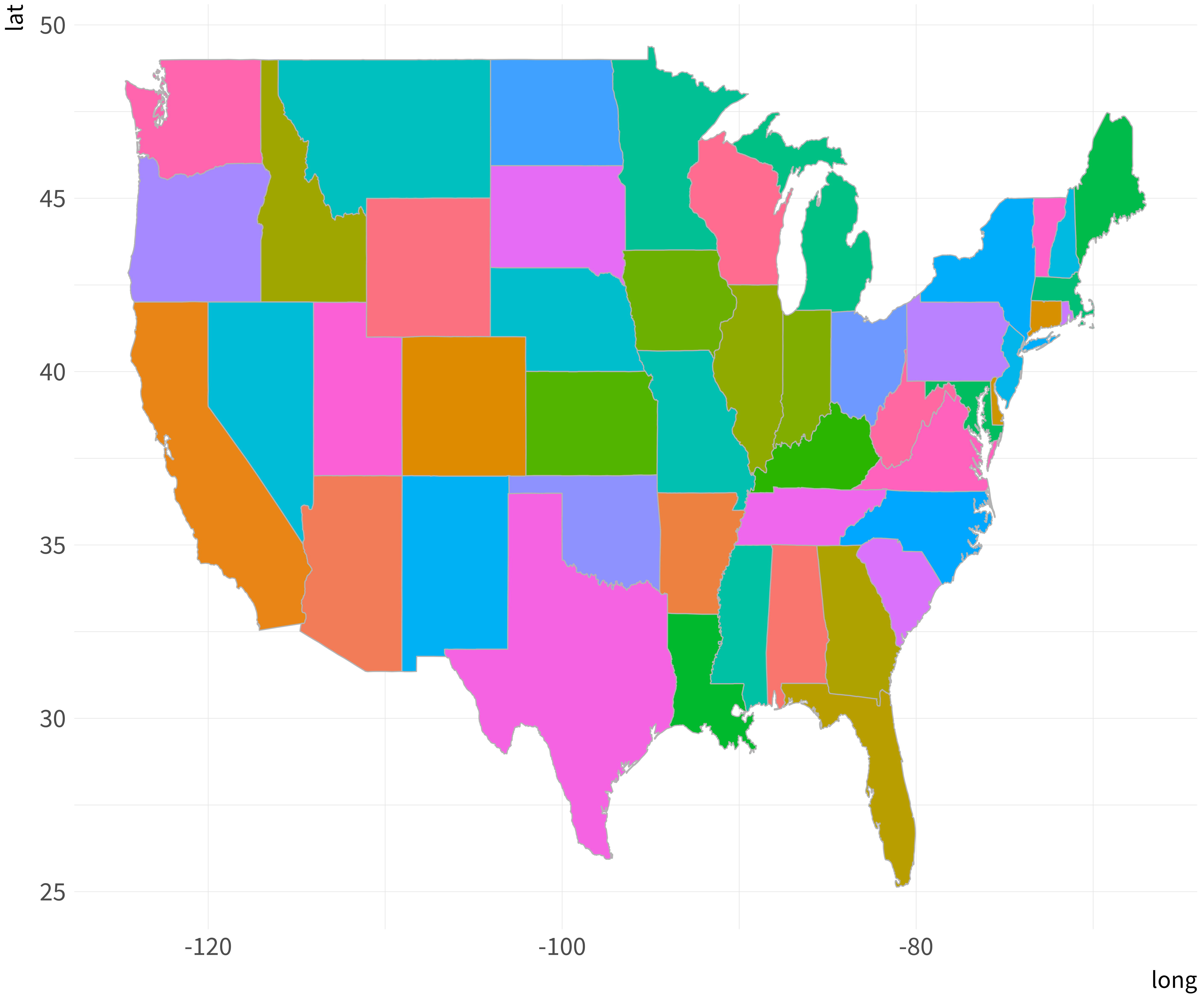


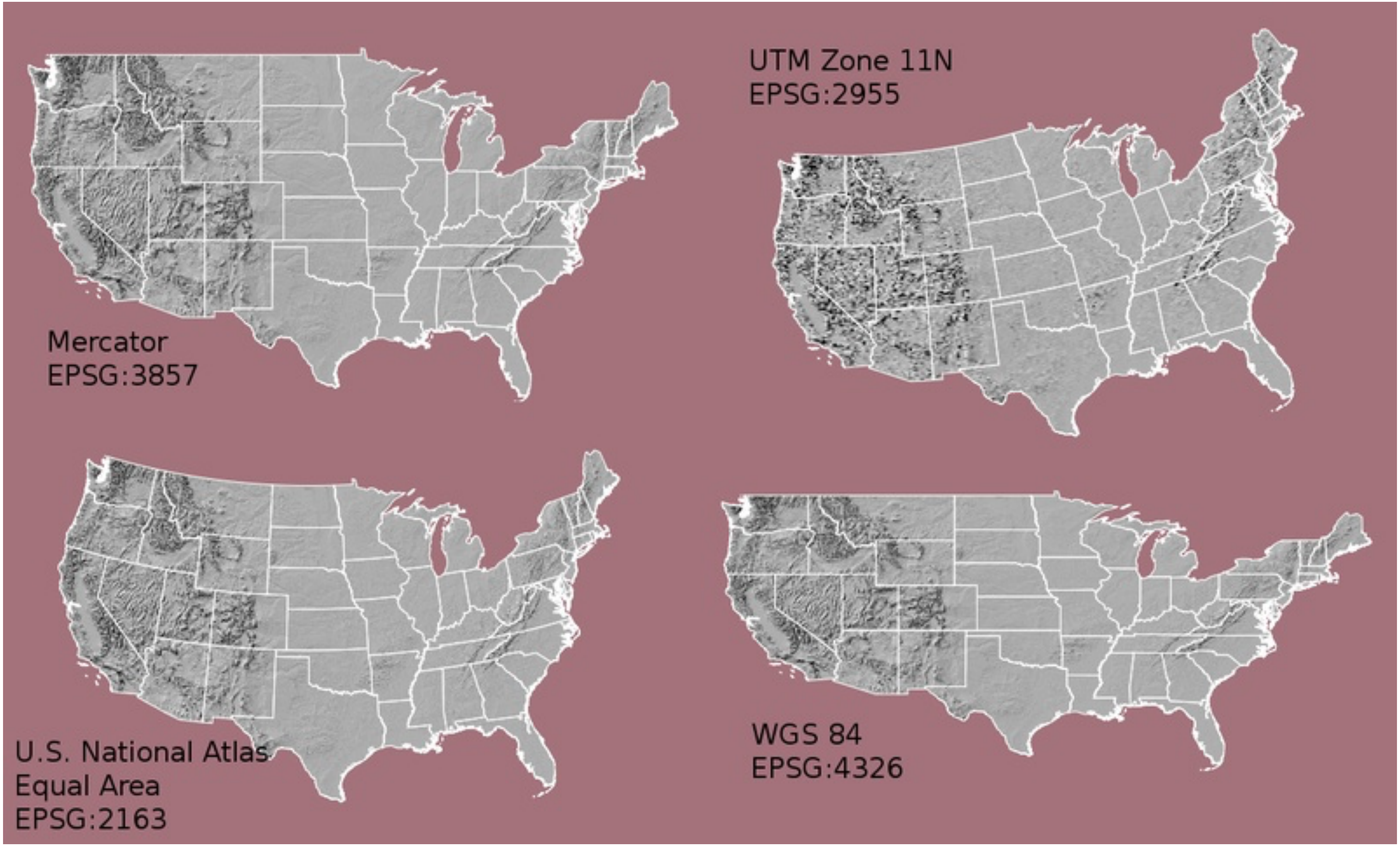
```r
library(maps)
us_states <- map_data("state")
head(us_states)

##        long      lat group order  region subregion
## 1 -87.4620 30.3897     1     1 alabama      <NA>
## 2 -87.4849 30.3725     1     2 alabama      <NA>
## 3 -87.5250 30.3725     1     3 alabama      <NA>
## 4 -87.5308 30.3324     1     4 alabama      <NA>
## 5 -87.5709 30.3267     1     5 alabama      <NA>
## 6 -87.5881 30.3267     1     6 alabama      <NA>

dim(us_states)

## [1] 15537     6
```

```r
p <- ggplot(data = us_states, aes(x = long, y = lat, group = group))

p + geom_polygon(fill = "white", color = "black")
```



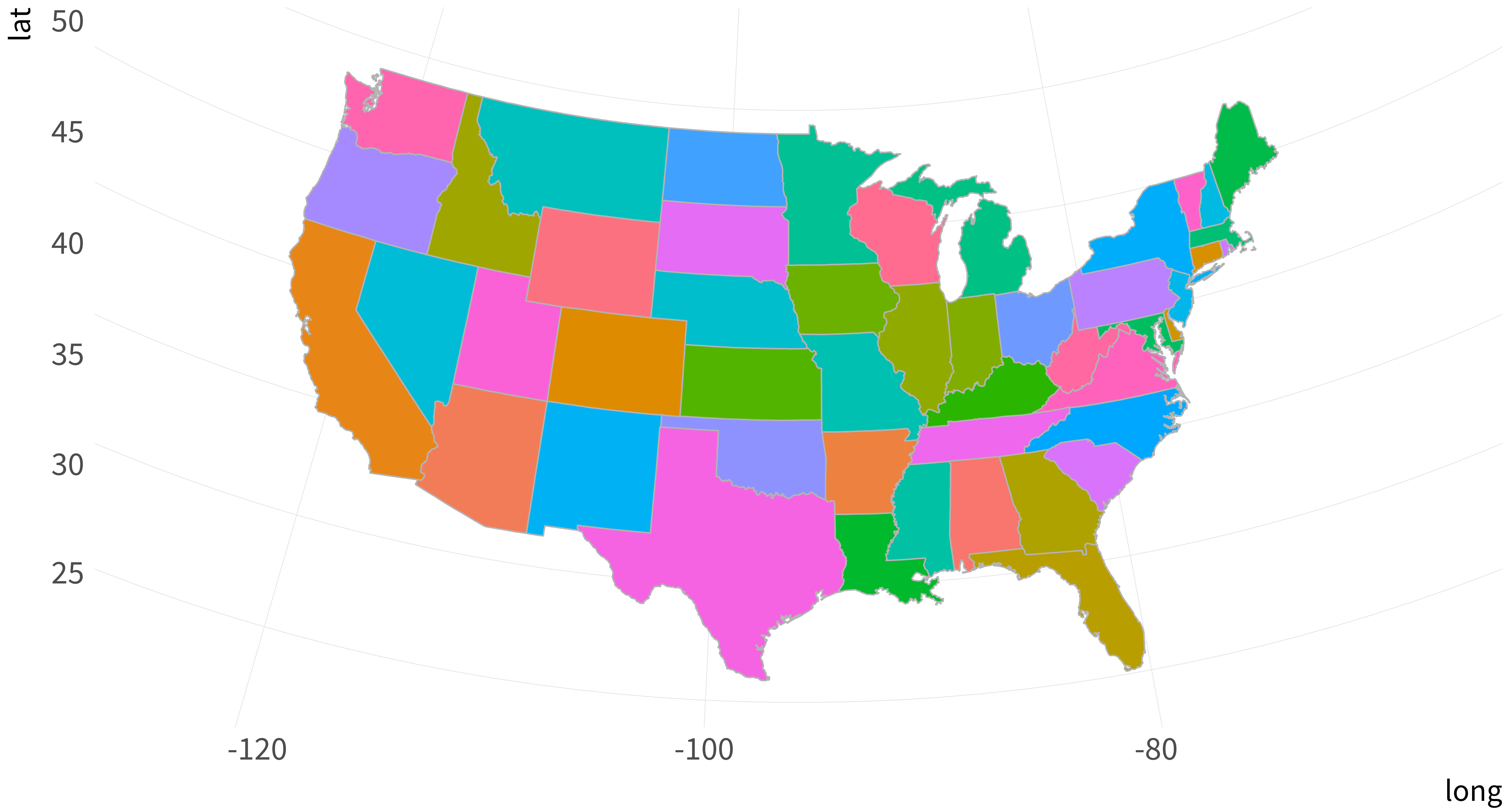
```r
p <- ggplot(data = us_states, aes(x = long, y = lat, group = group, fill = region))

p + geom_polygon(color = "gray70", size = 0.2) + guides(fill = FALSE)
```




UTM Zone 11N
EPSG:2955
Mercator
EPSG:3857
U.S. National Atlas
Equal Area
EPSG:2163
WGS 84
EPSG:4326

```r
p <- ggplot(data = us_states, aes(x = long, y = lat,
                                  group = group, fill = region))

p + geom_polygon(color = "gray70", size = 0.2) +
    coord_map(projection = "albers", lat0 = 39, lat1 = 45) +
    guides(fill = FALSE)
```

Many other projections are available


lat
50
45
40
35
30
25
-120
-100
-80
long

```r
election$region <- tolower(election$state)
us_states_elec <- left_join(us_states, election)
```

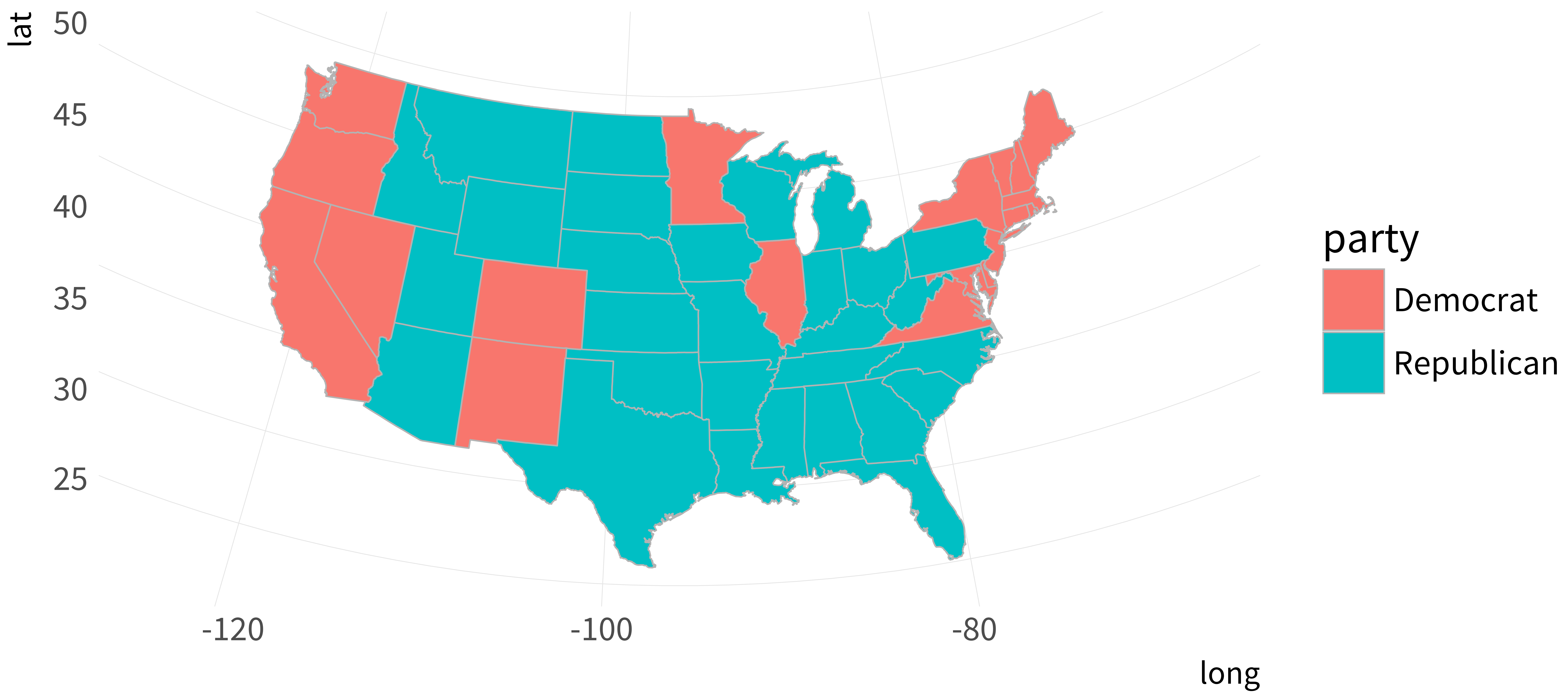
Don’t do this part blindly. Be aware of the variables/ keys you are merging/ joining the data on and why you are doing it.

https://github.com/kjhealy/fips-codes

state.name	state.abbr	long.name	fips	sumlev	region	division	state	region.name	division.name
Alabama	AL	Alabama AL	1	40	3	6	1	South	East South Central
Alaska	AK	Alaska AK	2	40	4	9	2	West	Pacific
Arizona	AZ	Arizona AZ	4	40	4	8	4	West	Mountain
Arkansas	AR	Arkansas AR	5	40	3	7	5	South	West South Central
California	CA	California CA	6	40	4	9	6	West	Pacific

Some helper CSV files

```r
p <- ggplot(data = us_states_elec,
            aes(x = long, y = lat,
                group = group, fill = party))

p + geom_polygon(color = "gray70", size = 0.2) +
    coord_map(projection = "albers", lat0 = 39, lat1 = 45)
```

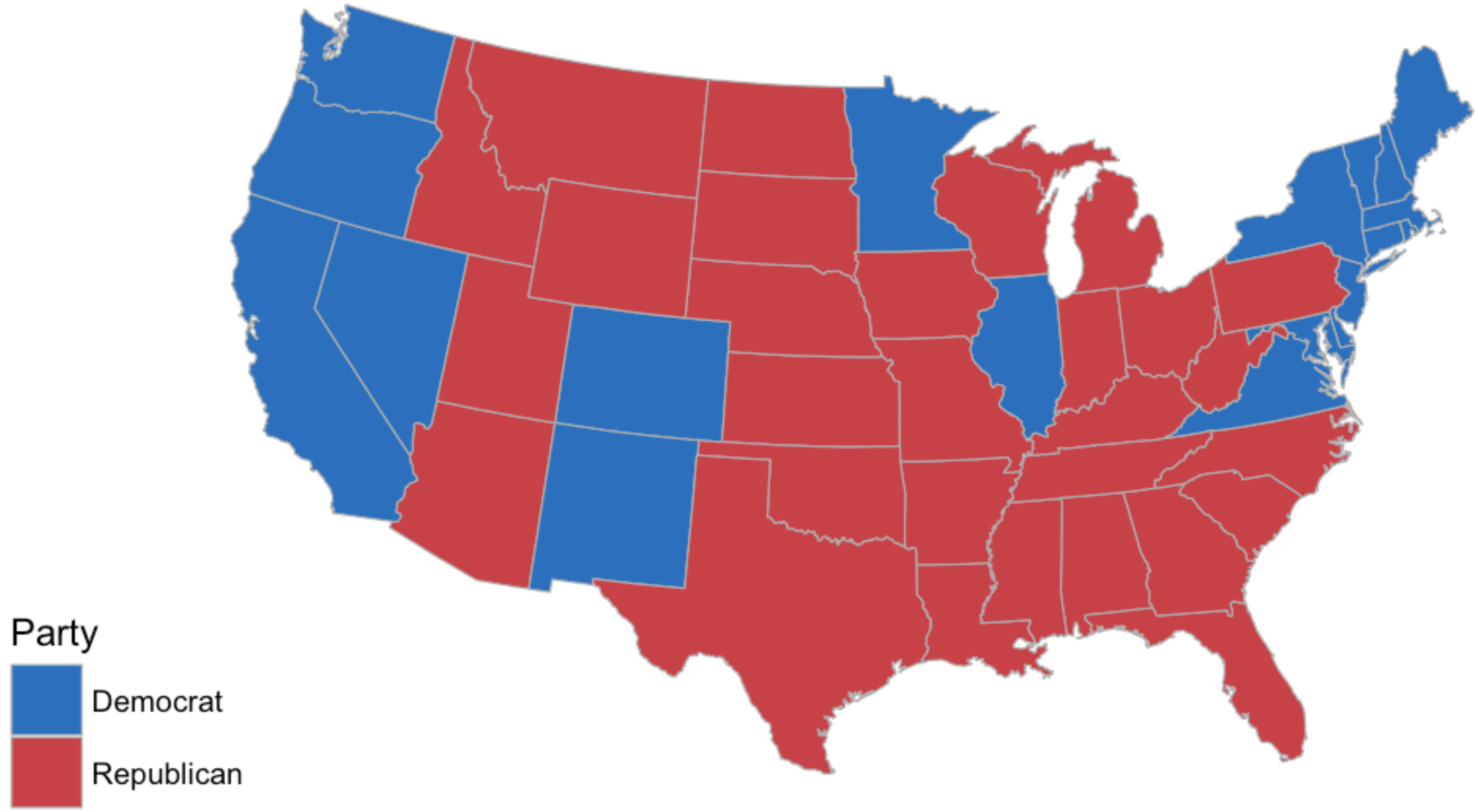

lat
50
45
40
35
30
25
-120
-100
-80
long
party
Democrat
Republican

```r
theme_map <- function(base_size=9, base_family="") {
    require(grid)
    theme_bw(base_size=base_size, base_family=base_family) %+replace%
        theme(axis.line=element_blank(),
              axis.text=element_blank(),
              axis.ticks=element_blank(),
              axis.title=element_blank(),
              panel.background=element_blank(),
              panel.border=element_blank(),
              panel.grid=element_blank(),
              panel.spacing=unit(0, "lines"),
              plot.background=element_blank(),
              legend.justification = c(0,0),
              legend.position = c(0,0)
              )
}
```

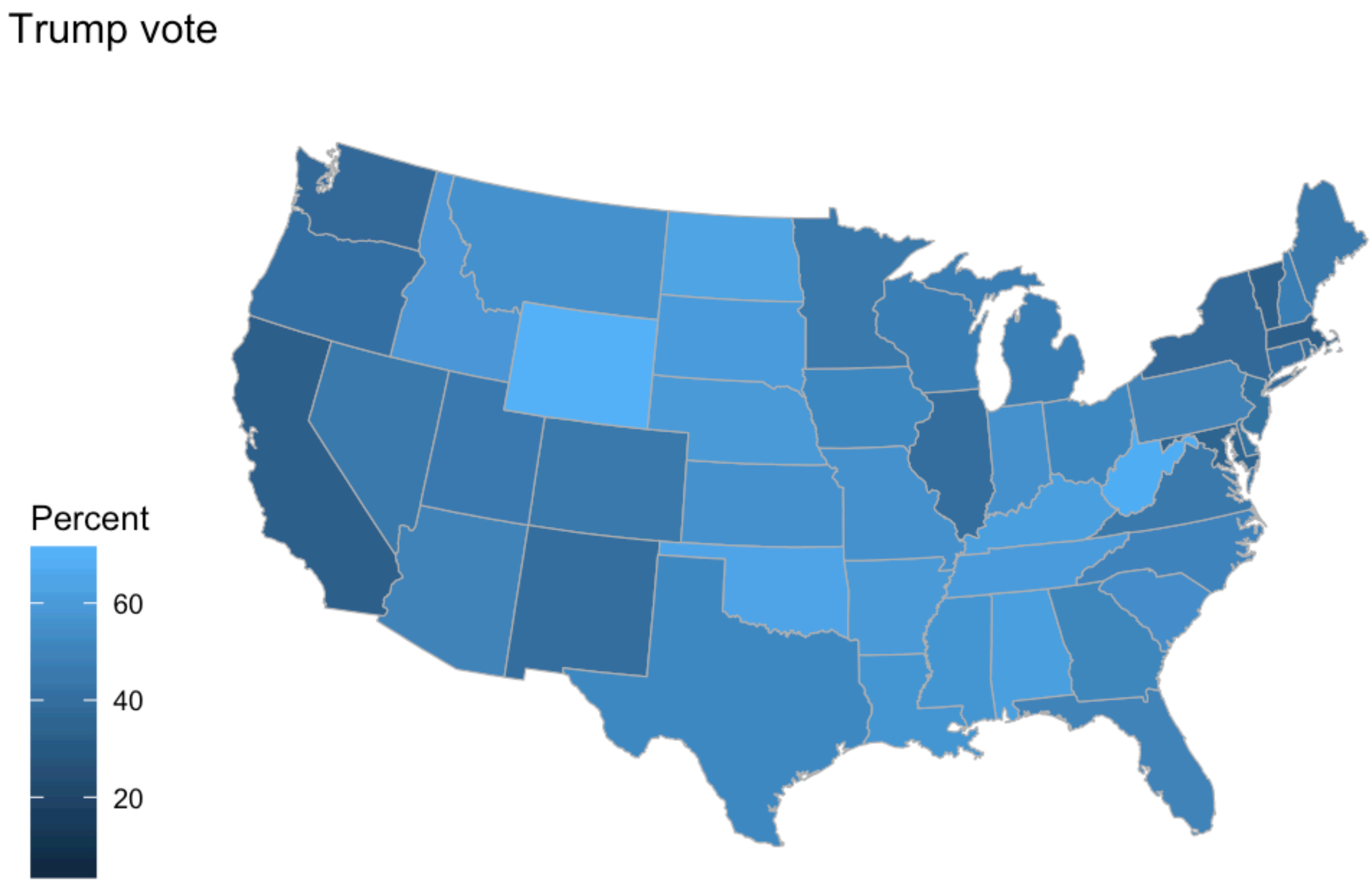
http://kieranhealy.org/sh

```r
p0 <- ggplot(data = us_states_elec, aes(x = long, y = lat,
                                        group = group, fill = party))

p1 <- p0 + geom_polygon(color = "gray70", size = 0.2) +
    coord_map(projection = "albers", lat0 = 39, lat1 = 45)

p2 <- p1 + scale_fill_manual(values = party_colors) +
    labs(title = "Election Results 2016", fill = NULL)

p2 + theme_map()
```

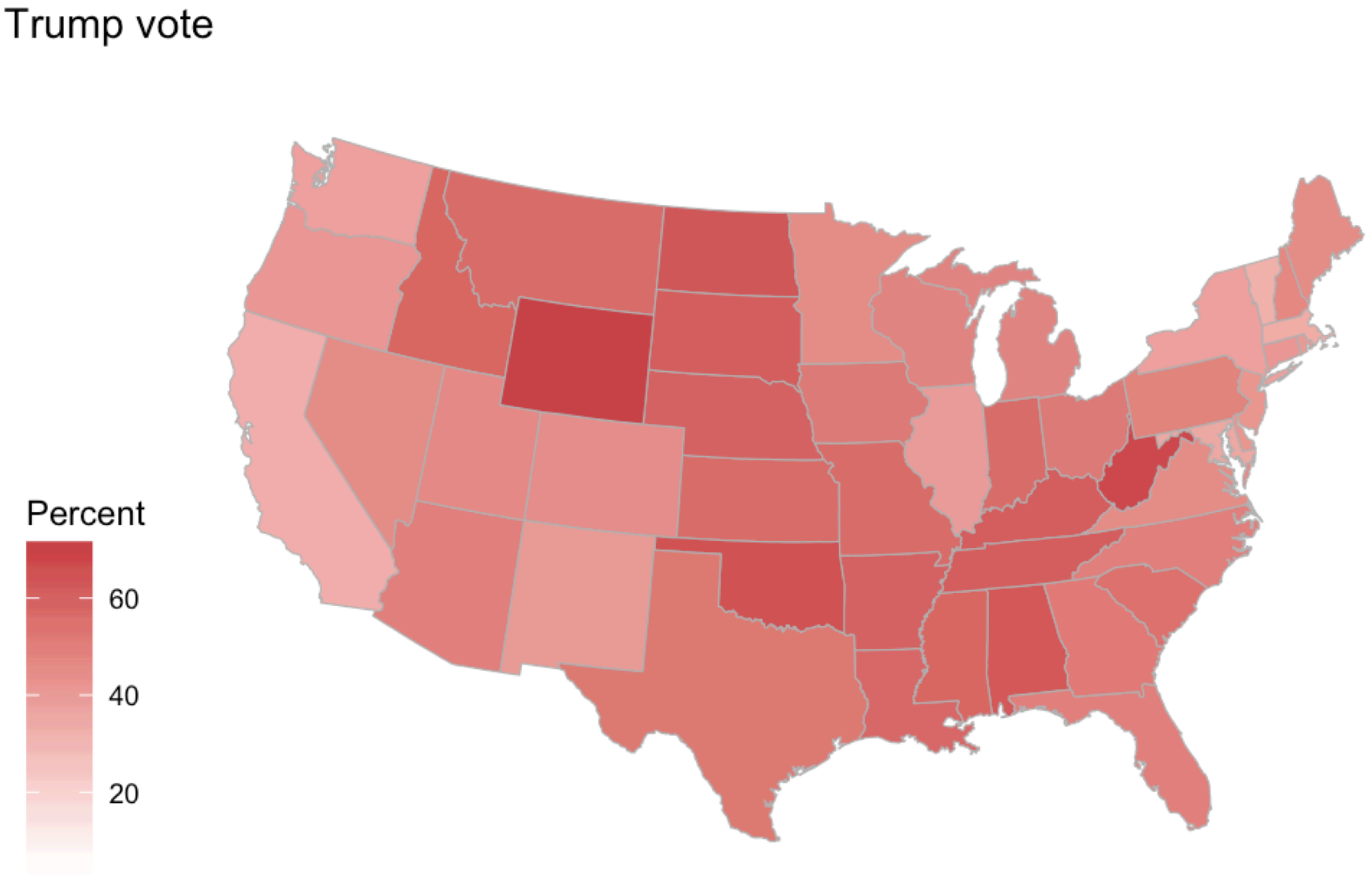
Election Results 2016



```r
p0 <- ggplot(data = us_states_elec, aes(x = long, y = lat,
                                        group = group, fill = pct_trump))

p1 <- p0 + geom_polygon(color = "gray70", size = 0.2) +
    coord_map(projection = "albers", lat0 = 39, lat1 = 45)

p2 <- p1 + labs(title = "Trump vote")

p2 + theme_map() + labs(fill = "Percent")
```

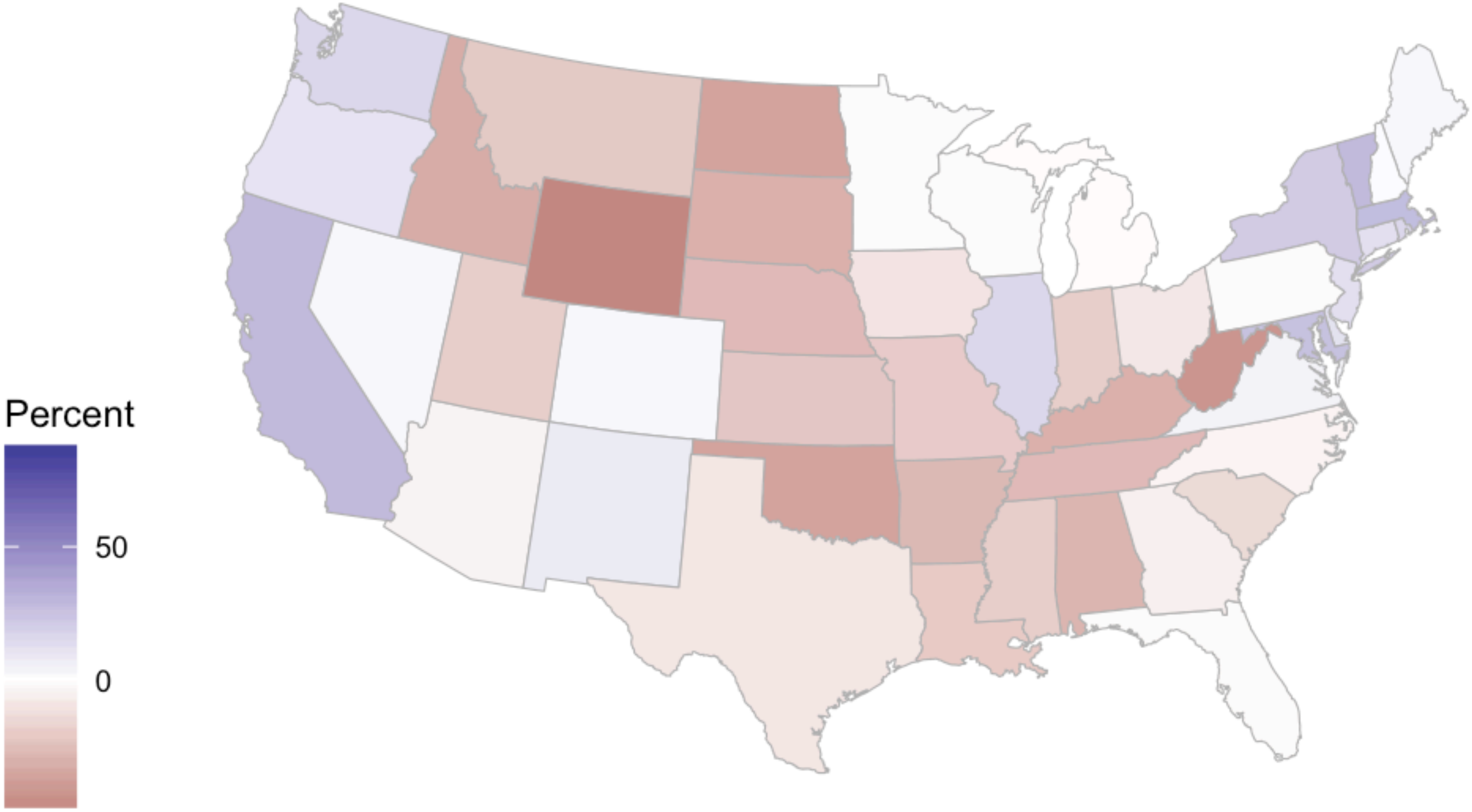

Trump vote
Percent
60
40
20

```r
p0 <- ggplot(data = us_states_elec, aes(x = long, y = lat,
                                         group = group, fill = pct_trump))

p1 <- p0 + geom_polygon(color = "gray70", size = 0.2) +
    coord_map(projection = "albers", lat0 = 39, lat1 = 45)

p2 <- p1 + scale_fill_gradient(low = "white", high = "#CB454A") +
        labs(title = "Trump vote")
p2 + theme_map() + labs(fill = "Percent")
```

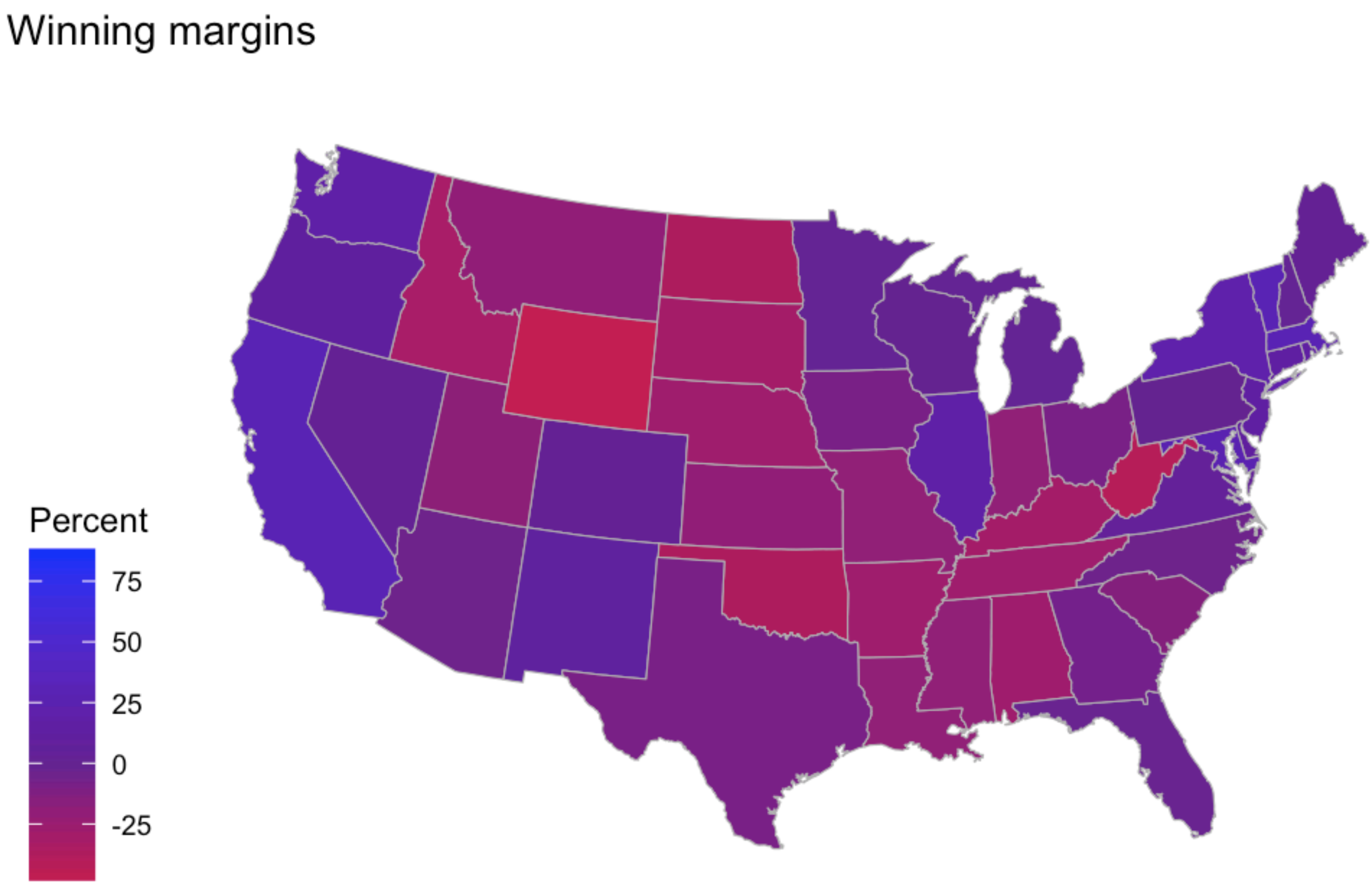

Trump vote
Percent
60
40
20

```r
p0 <- ggplot(data = us_states_elec, aes(x = long, y = lat,
                                        group = group, fill = d_points))

p1 <- p0 + geom_polygon(color = "gray70", size = 0.2) +
    coord_map(projection = "albers", lat0 = 39, lat1 = 45)

p2 <- p1 + scale_fill_gradient2() + labs(title = "Winning margins")
p2 + theme_map() + labs(fill = "Percent")
```

Winning margins

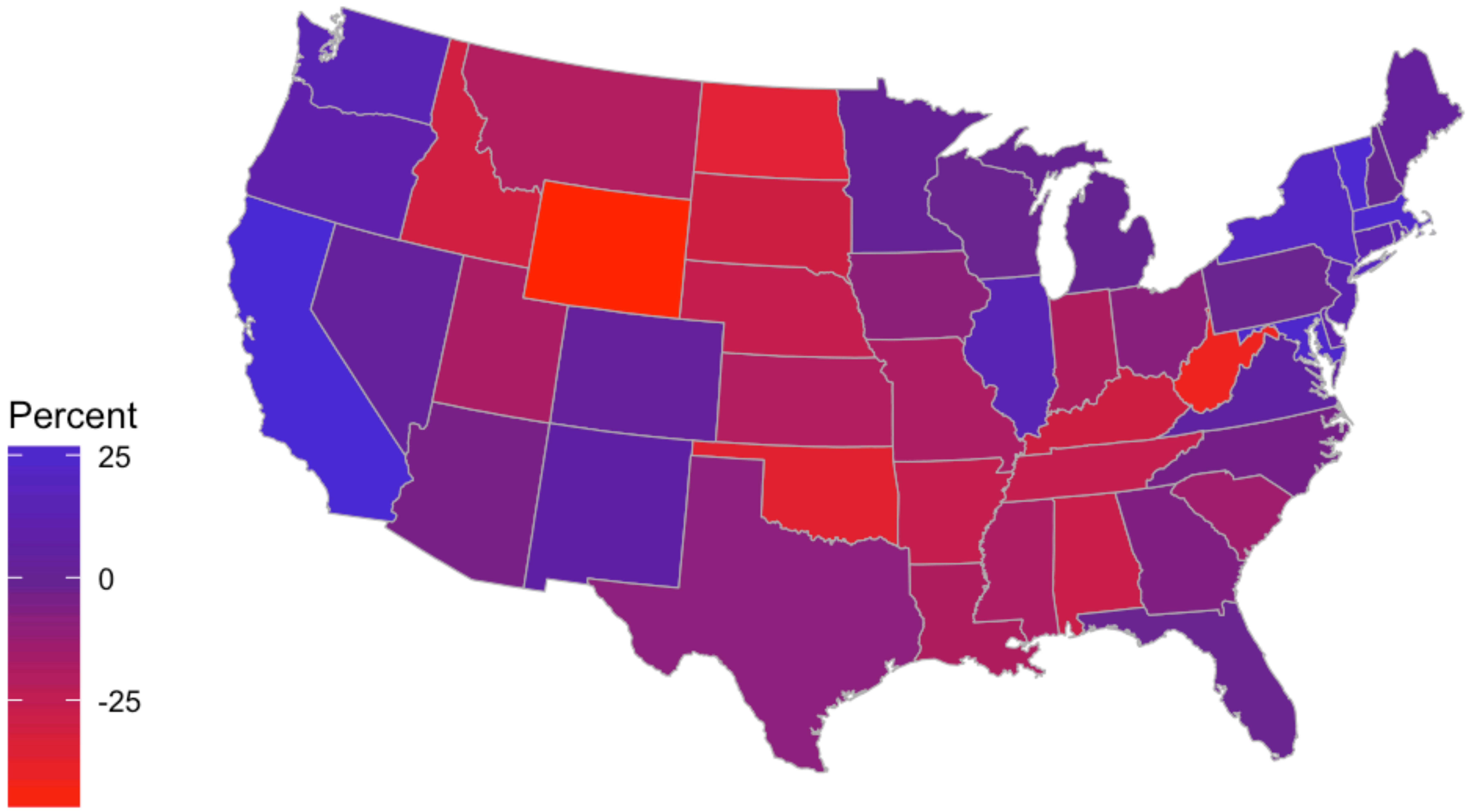


```r
p0 <- ggplot(data = us_states_elec, aes(x = long, y = lat,
                                         group = group, fill = d_points))

p1 <- p0 + geom_polygon(color = "gray70", size = 0.2) +
    coord_map(projection = "albers", lat0 = 39, lat1 = 45)

p2 <- p1 + scale_fill_gradient2(low = "red",
                                mid = scales::muted("purple"),
                                high = "blue",
                                breaks = c(-25, 0, 25, 50, 75)) +
    labs(title = "Winning margins")
p2 + theme_map() + labs(fill = "Percent")
```


Winning margins
Percent
75
50
25
0
-25

```r
p0 <- ggplot(data = subset(us_states_elec,
                           region %nin% "district of columbia"),
             aes(x = long, y = lat, group = group, fill = d_points))

p1 <- p0 + geom_polygon(color = "gray70", size = 0.2) +
    coord_map(projection = "albers", lat0 = 39, lat1 = 45)

p2 <- p1 + scale_fill_gradient2(low = "red",
                                mid = scales::muted("purple"),
                                high = "blue") +
    labs(title = "Winning margins")
p2 + theme_map() + labs(fill = "Percent")
```

Winning margins



US COUNTIES

http://eric.clst.org/Stuff/USGeoJSON

County choropleths are conceptually identical to state ones. The tables are just bigger, because there are way more lines to draw.

http://eric.clst.org/Stuff/USGeoJSON

County files in USGeoJSON format, at varying resolutions

AMERICA'S UR-CHOROPLETHS

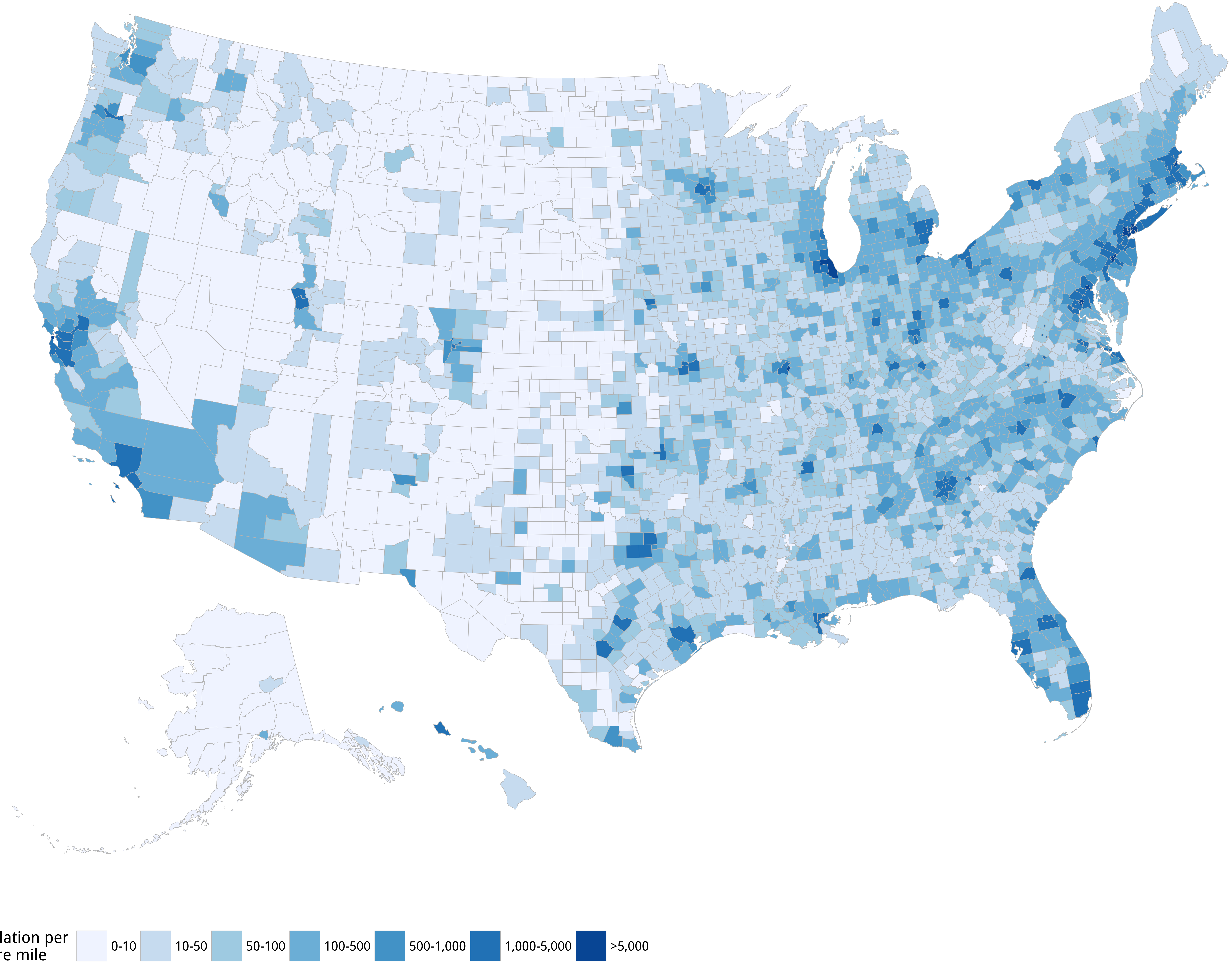
```r
county_map %>% sample_n(10)
```

```
##              long      lat  order  hole piece            group    id
## 162407   602940 -1598728 162407 FALSE     1 0500000US48351.1 48351
## 128954   262762   138691 128954 FALSE     1 0500000US38077.1 38077
## 128454   -83454   255783 128454 FALSE     1 0500000US38065.1 38065
## 987     1150231 -1429721    987 FALSE     1 0500000US01025.1 01025
## 154248   902190 -1014238 154248 FALSE     1 0500000US47157.1 47157
## 4261   -1458767 -2381708   4261 FALSE     1 0500000US02013.1 02013
## 160533   571407 -1586804 160533 FALSE     1 0500000US48241.1 48241
## 180058 -1752450   400802 180058 FALSE     1 0500000US53069.1 53069
## 183904  1758330  -497820 183904 FALSE     1 0500000US54071.1 54071
## 71658    836958 -1549129  71658 FALSE     1 0500000US22037.1 22037
```

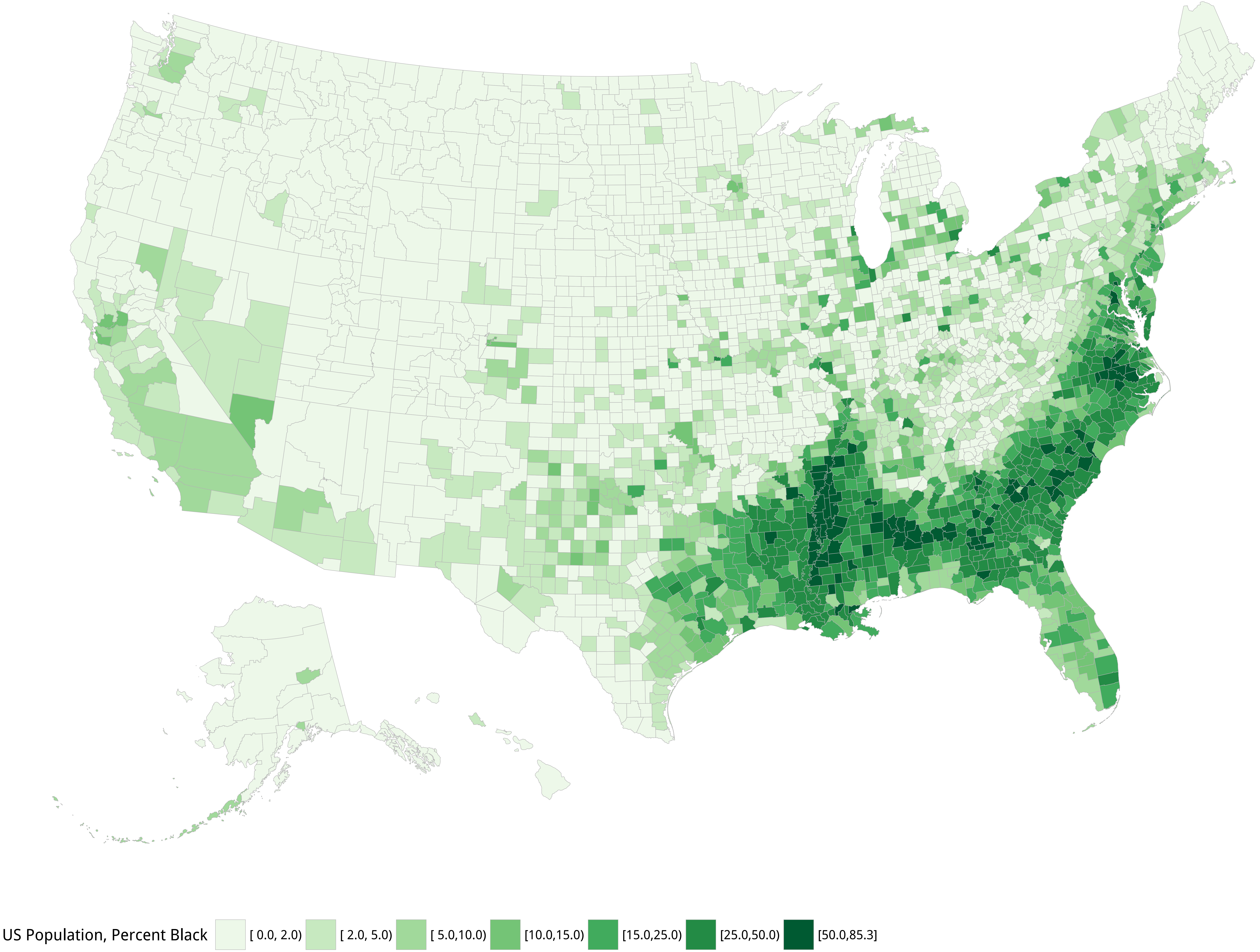
```r
county_data %>%
    select(id, name, state, pop_dens, pct_black) %>%
    sample_n(10)
```

```
##         id                    name state      pop_dens   pct_black
## 954  20095          Kingman County    KS [     0,   10) [ 0.0, 2.0)
## 1968 37085          Harnett County    NC [   100,  500) [15.0,25.0)
## 356  12051           Hendry County    FL [    10,   50) [10.0,15.0)
## 2674 48211         Hemphill County    TX [     0,   10) [ 0.0, 2.0)
## 2821 48505           Zapata County    TX [    10,   50) [ 0.0, 2.0)
## 2971 51580          Covington city    VA [  1000, 5000) [10.0,15.0)
## 2248 41001            Baker County    OR [     0,   10) [ 0.0, 2.0)
## 1644 30035          Glacier County    MT [     0,   10) [ 0.0, 2.0)
## 2844 49041           Sevier County    UT [    10,   50) [ 0.0, 2.0)
## 2932 51133 Northumberland County    VA [    50,  100) [25.0,50.0)
```

```r
county_full <- left_join(county_map, county_data, by = "id")

p <- ggplot(data = county_full,
            mapping = aes(x = long, y = lat,
                          fill = pop_dens,
                          group = group))

p1 <- p + geom_polygon(color = "gray70", size = 0.1) + coord_equal()

p2 <- p1 + scale_fill_brewer(palette="Blues",
                             labels = c("0-10", "10-50", "50-100",
                                        "100-500", "500-1,000",
                                        "1,000-5,000", ">5,000"))

p2 + labs(fill = "Population per\nsquare mile") + theme_map() +
    guides(fill = guide_legend(nrow = 1)) +
    theme(legend.position = "bottom")
```


Population per square mile
0-10
10-50
50-100
100-500
500-1,000
1,000-5,000
>5,000

```r
p <- ggplot(data = county_full,
            mapping = aes(x = long, y = lat,
                          fill = pct_black,
                          group = group))

p1 <- p + geom_polygon(color = "gray70", size = 0.1) + coord_equal()

p2 <- p1 + scale_fill_brewer(palette="Greens")

p2 + labs(fill = "US Population, Percent Black") +
    guides(fill = guide_legend(nrow = 1)) +
    theme_map() + theme(legend.position = "bottom")
```


US Population, Percent Black
[0.0, 2.0)
[2.0, 5.0)
[5.0,10.0)
[10.0,15.0)
[15.0,25.0)
[25.0,50.0)
[50.0,85.3]

```r
orange_pal <- RColorBrewer::brewer.pal(n = 6, name = "Oranges")
orange_pal
```

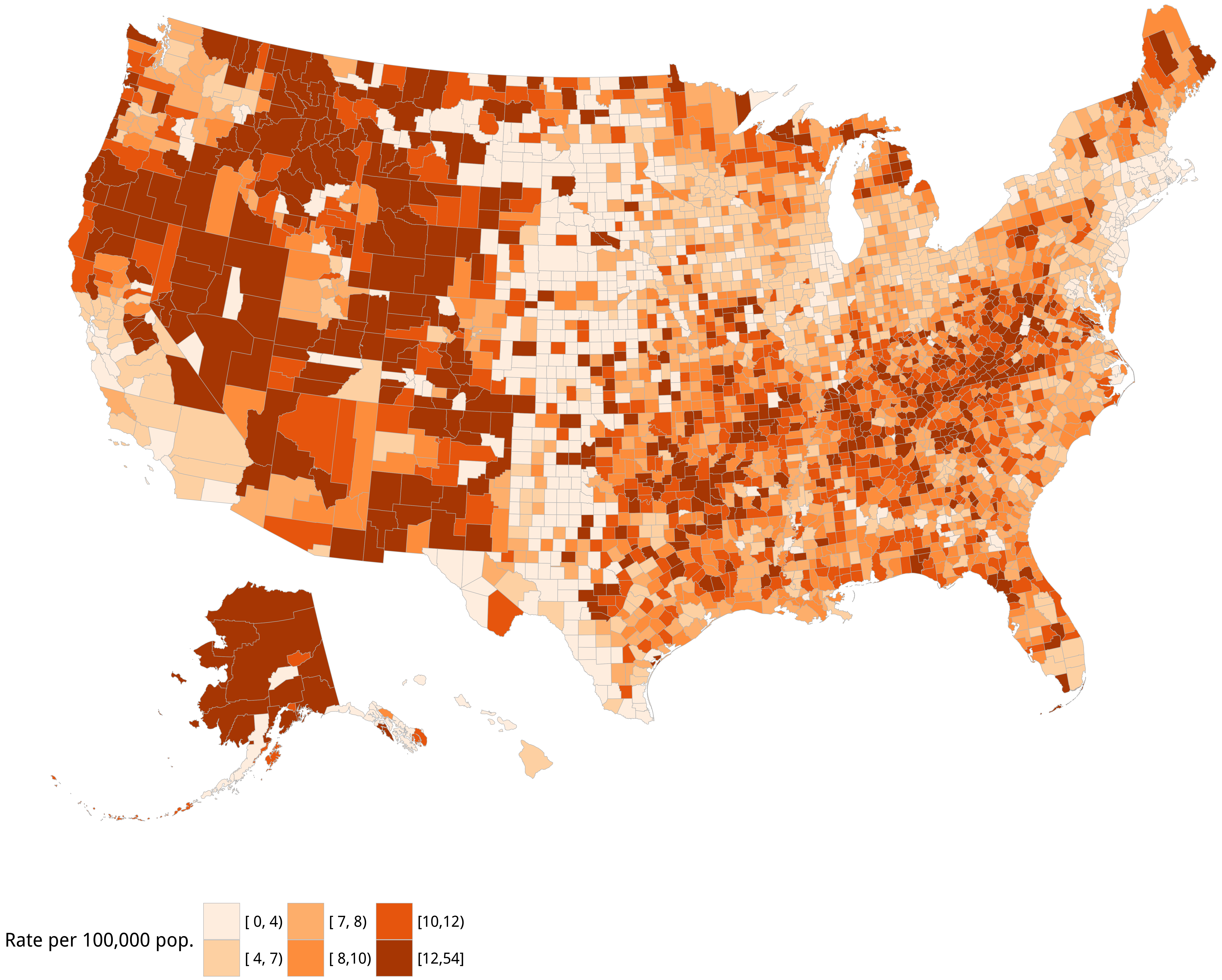
```
## [1] "#FEEDDE" "#FDD0A2" "#FDAE6B" "#FD8D3C" "#E6550D"
## [6] "#A63603"
```

```r
orange_rev <- rev(orange_pal)
orange_rev
```

```
## [1] "#A63603" "#E6550D" "#FD8D3C" "#FDAE6B" "#FDD0A2"
## [6] "#FEEDDE"
```

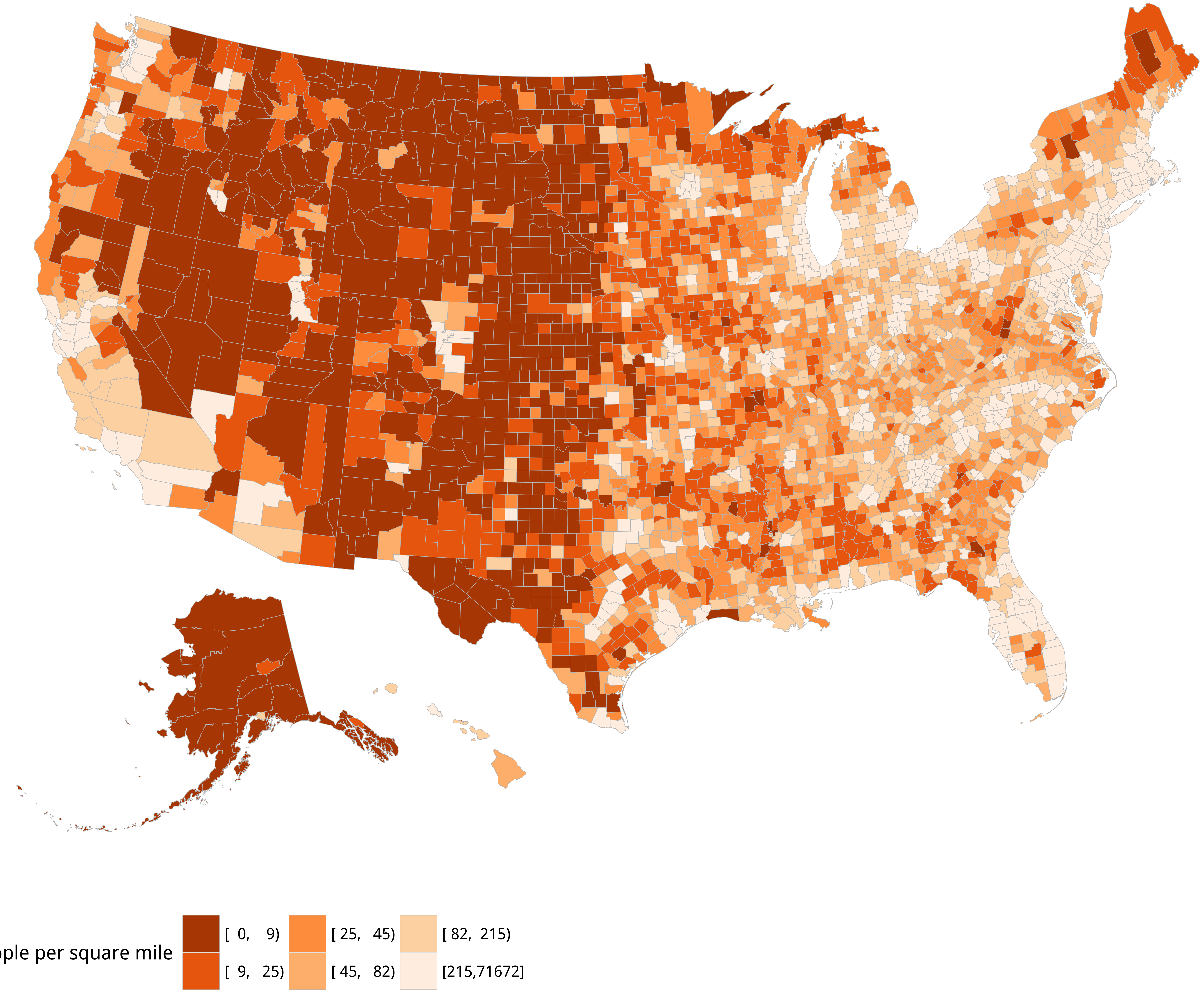
```r
gun_p <- ggplot(data = county_full,
             mapping = aes(x = long, y = lat,
                           fill = su_gun6,
                           group = group))

gun_p1 <- gun_p + geom_polygon(color = "gray70", size = 0.1) + coord_equal()

gun_p2 <- gun_p1 + scale_fill_manual(values = orange_pal)

gun_p2 + labs(title = "Gun-Related Suicides, 1999-2015",
              fill = "Rate per 100,000 pop.") +
    theme_map() +
    theme(legend.position = "bottom")
```

Gun-Related Suicides, 1999-2015

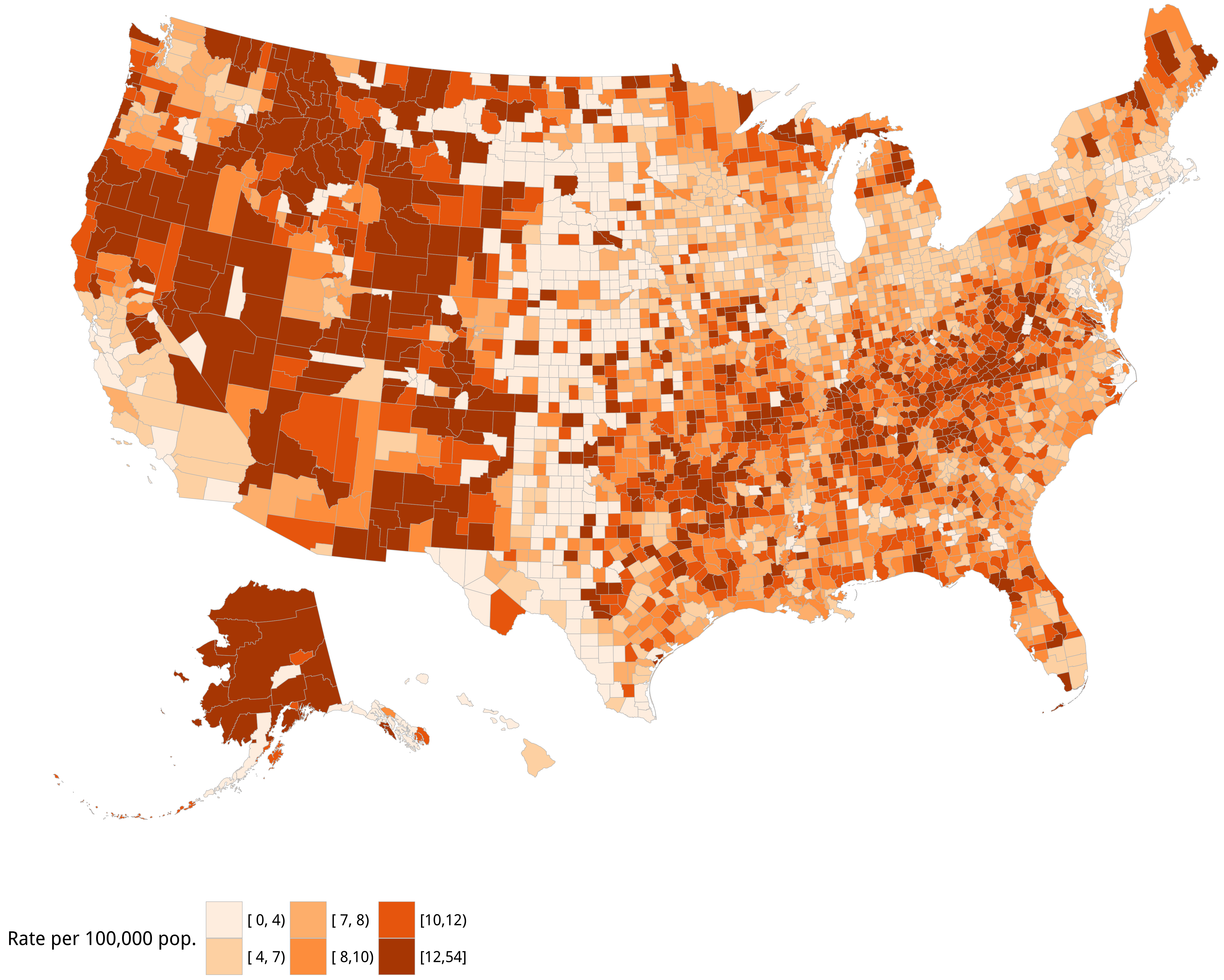


```r
pop_p <- ggplot(data = county_full,
            mapping = aes(x = long, y = lat,
                          fill = pop_dens6,
                          group = group))

pop_p1 <- pop_p + geom_polygon(color = "gray70", size = 0.1) + coord_equal()

pop_p2 <- pop_p1 + scale_fill_manual(values = orange_rev)

pop_p2 + labs(title = "Reverse-coded Population Density",
              fill = "People per square mile") +
    theme_map() +
    theme(legend.position = "bottom")
```

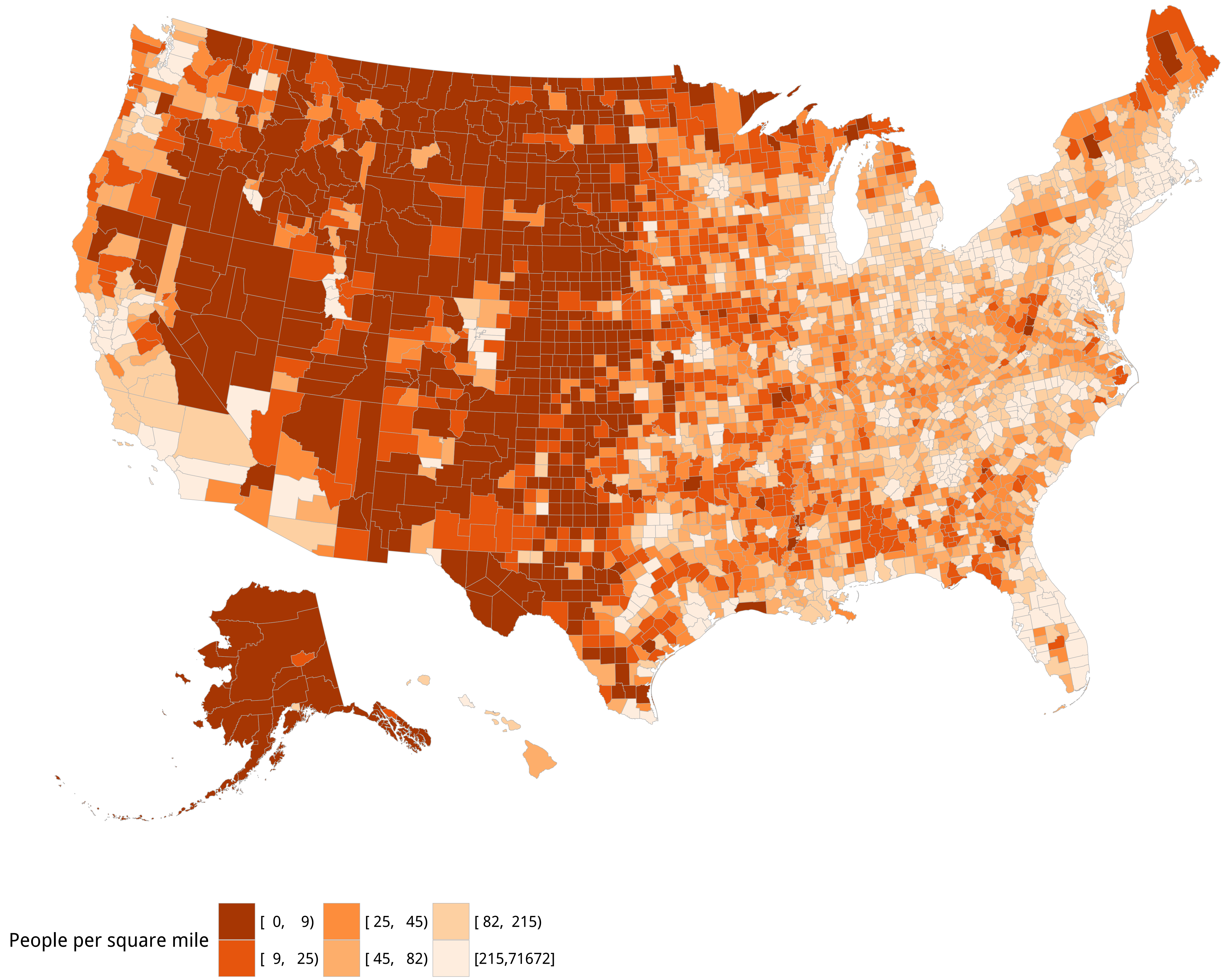
Reverse-coded Population Density

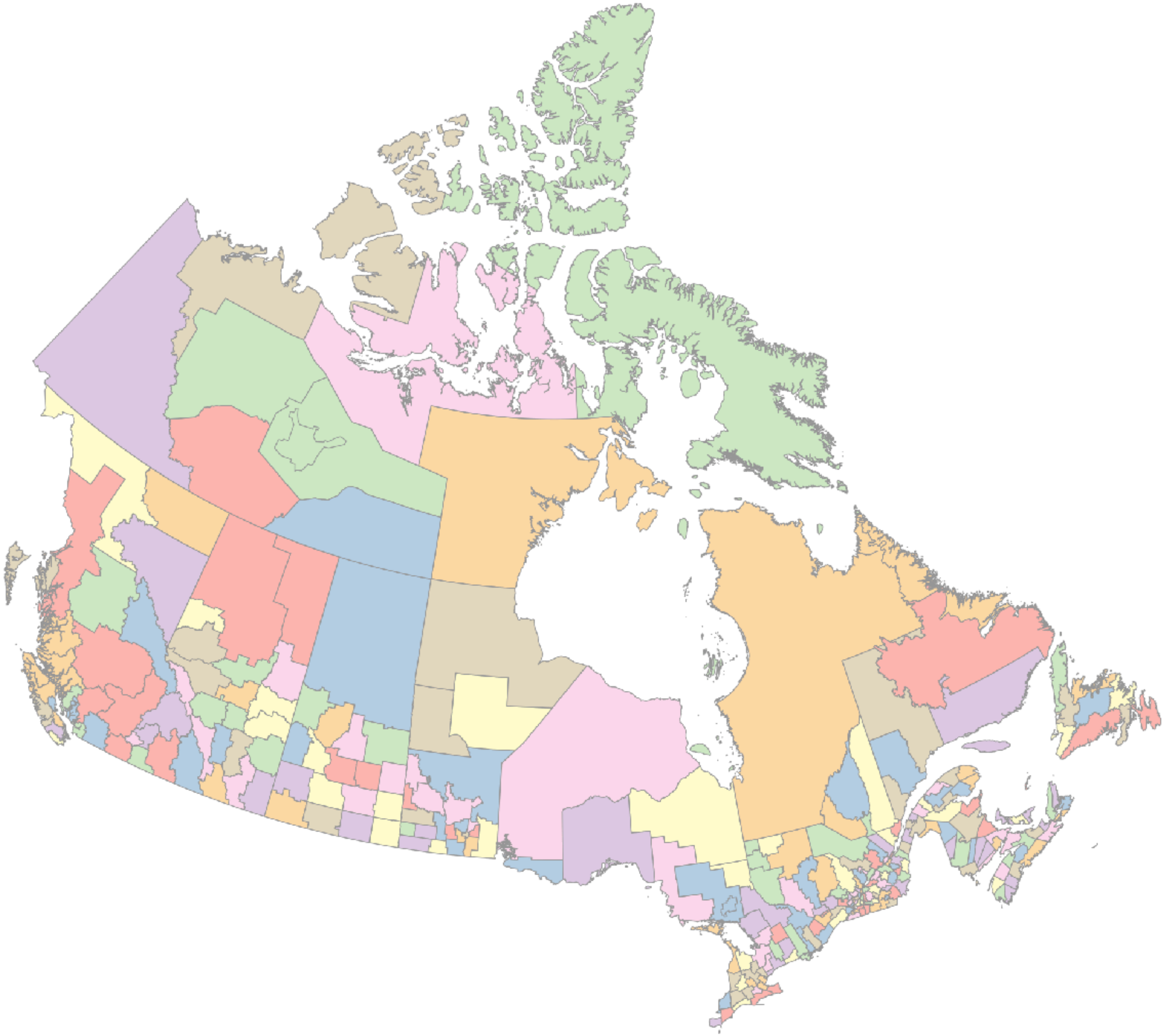


Gun-Related Suicides, 1999-2015



Reverse-coded Population Density





http://kieranhealy.org/canada.r

http://kieranhealy.org/gcd_000b11a_e.zip

SMALL MULTIPLES FOR MAPS

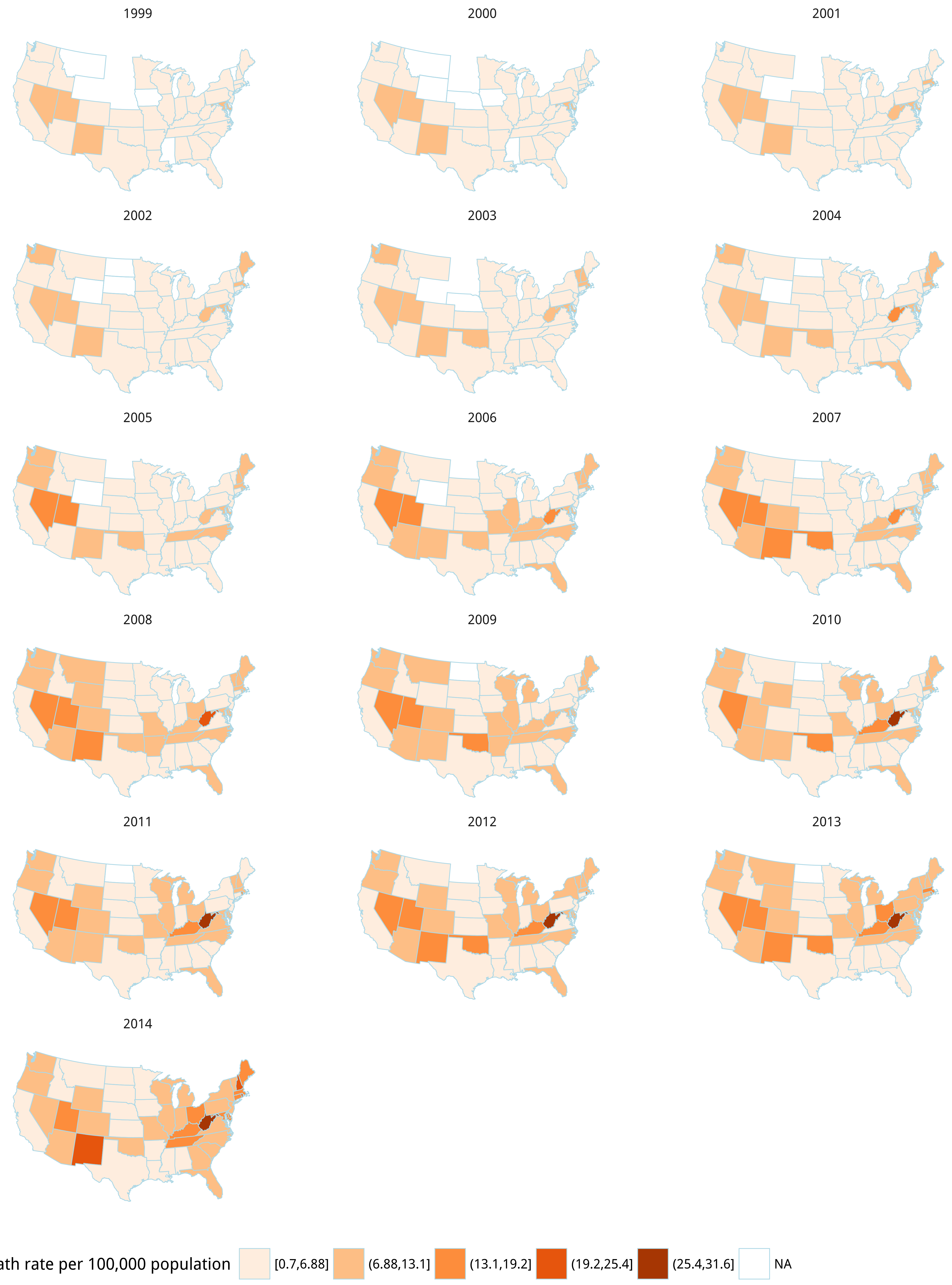
```r
head(opiates)
```

```
## # A tibble: 6 x 10
##    Year      State  FIPS Deaths Population Crude Adjusted
##   <int>      <chr> <int>  <int>      <int> <dbl>    <dbl>
## 1  1999    Alabama     1     37    4430141   0.8      0.8
## 2  1999     Alaska     2     27     624779   4.3      4.0
## 3  1999    Arizona     4    229    5023823   4.6      4.7
## 4  1999   Arkansas     5     28    2651860   1.1      1.1
## 5  1999 California     6   1474   33499204   4.4      4.5
## 6  1999   Colorado     8    164    4226018   3.9      3.7
## # ... with 3 more variables: Adjusted.se <dbl>,
## #   Region <ord>, Abbr <chr>
```

```r
opiates$region <- tolower(opiates$State)

opiates_map <- left_join(us_states, opiates)
```

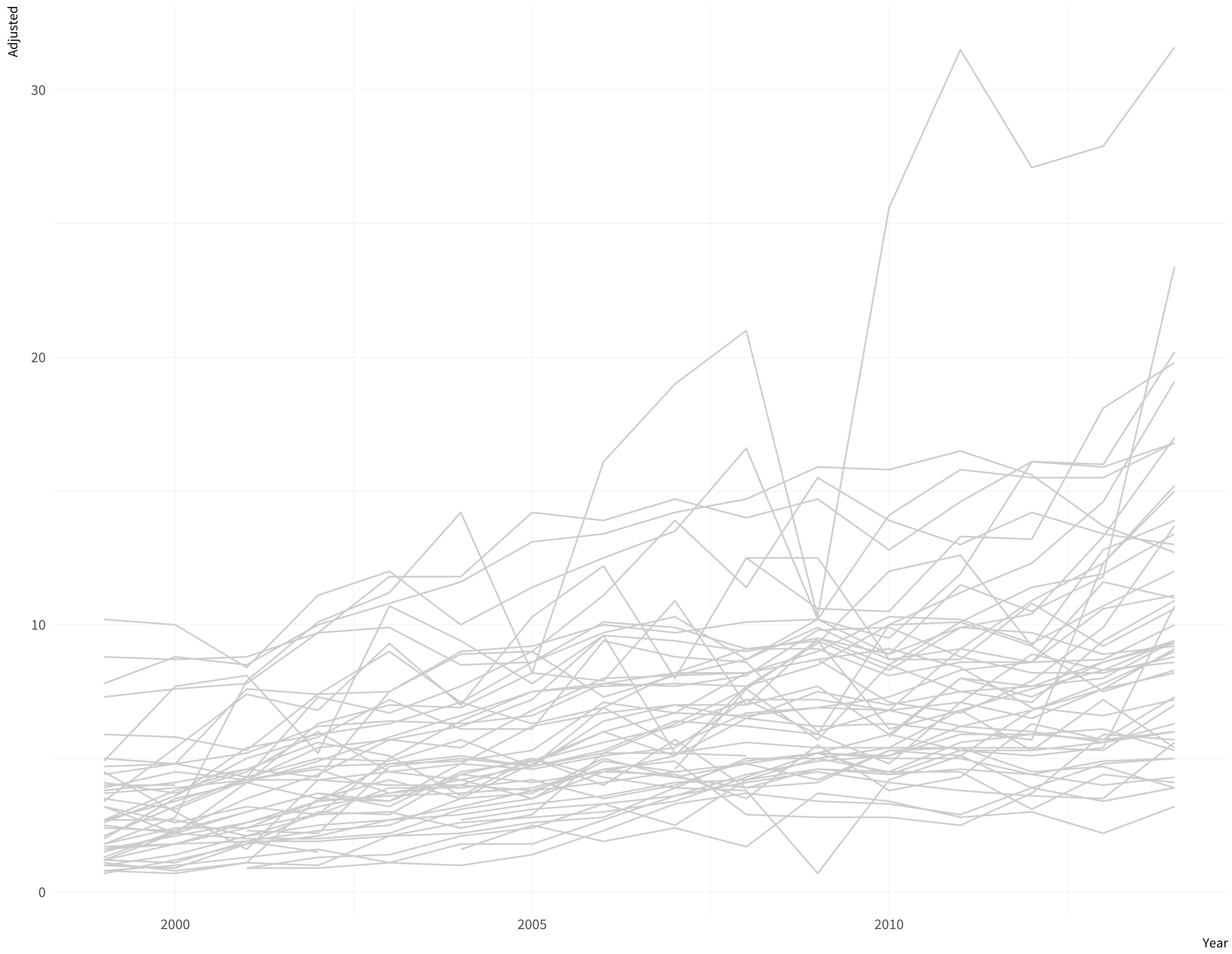
```r
p0 <- ggplot(data = opiates_map,
             aes(x = long, y = lat,
                 group = group,
                 fill = cut_interval(Adjusted, n = 5)))

p1 <- p0 + geom_polygon(color = "lightblue", size = 0.2) +
    coord_map(projection = "albers", lat0 = 39, lat1 = 45)

p2 <- p1 + scale_fill_brewer(type = "seq", palette = "Oranges")

p2 + theme_map() + facet_wrap(~ Year, ncol = 3) +
    guides(fill = guide_legend(nrow = 1)) +
    theme(legend.position = "bottom",
          strip.background = element_blank()) +
    labs(fill = "Death rate per 100,000 population",
         title = "Opiate Related Deaths by State, 1999-2014")
```

Opiate Related Deaths by State, 1999-2014



Is your data really spatial?

```r
p0 <- ggplot(data = opiates,
             mapping = aes(x = Year, y = Adjusted))

p1 <- p0 + geom_line(color = "gray85",
             mapping = aes(group = State))
```

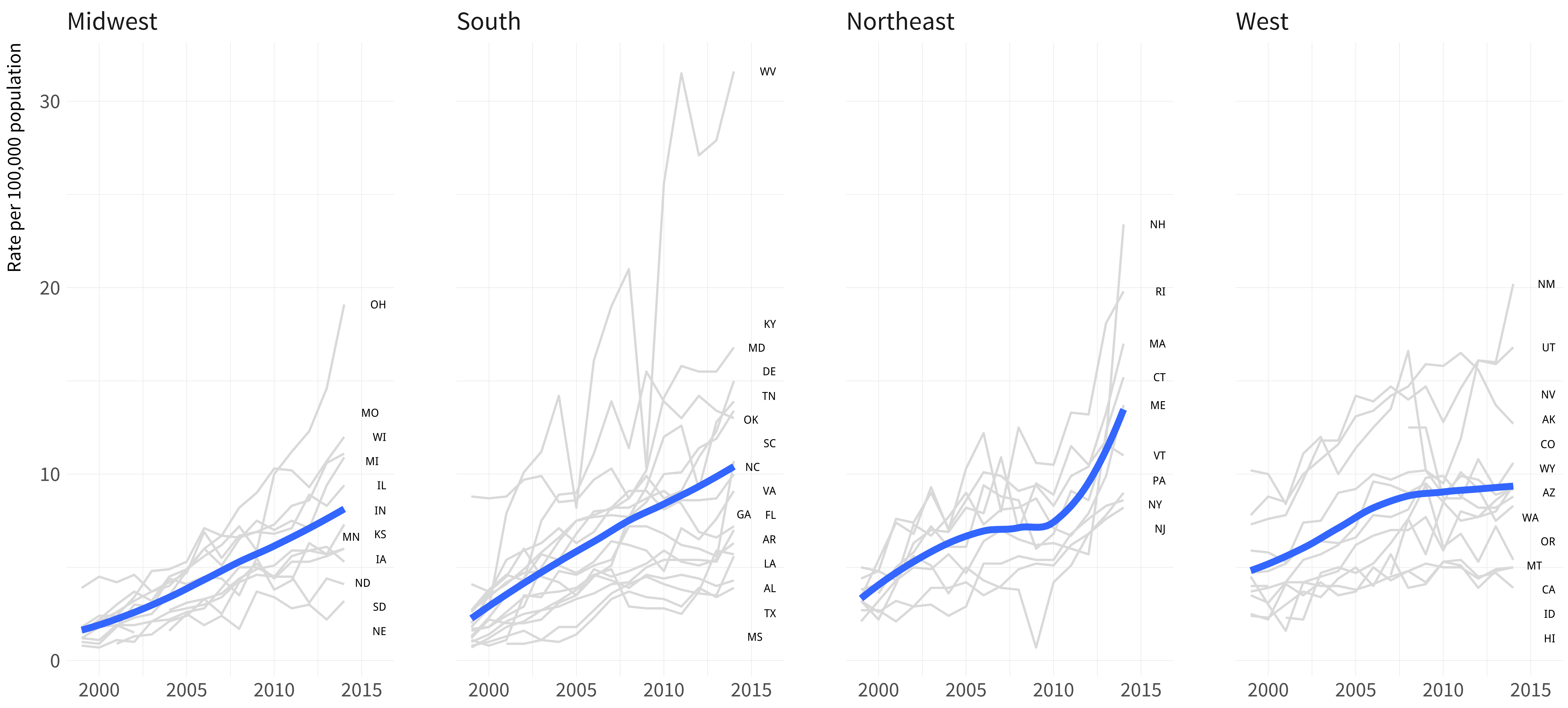

Adjusted
30
20
10
0
2000
2005
2010
Year

```r
p2 <- p1 + geom_smooth(mapping = aes(group = Region),
                       se = FALSE)
```

```r
p3 <- p2 + geom_text_repel(data = subset(opiates,
                                         Year == max(Year) & Abbr !="DC"),
                     mapping = aes(x = Year, y = Adjusted, label = Abbr),
                     size = 1.8, segment.color = NA, nudge_x = 30) +
         coord_cartesian(c(min(opiates$Year),
                     max(opiates$Year)))
```

```
p3 + labs(x = "", y = "Rate per 100,000 population",
          title = "State-Level Opiate Death Rates by Region, 1999-2014") +
    facet_wrap(~ reorder(Region, Adjusted, na.rm = TRUE), nrow  = 1)
```

State-Level Opiate Death Rates by Region, 1999-2014



SIMPLE FEATURES AND geom_sf()

r-spatial.org

r-spatial.github.io/sf/

walkerke.github.io/tidycensus

Tidy data + Simple features spec + Direct support in ggplot

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

install.packages("drat")

drat::addRepo("kjhealy")

install.packages("nycdogs")

library(sf)

library(nycdogs)
```

nycdogs 0.1.0 | Reference

nycdogs

The `nycdogs` package contains three datasets, `nyc_license`, `nyc_bites`, and `nyc_zips`. They contain, respectively, data on all licensed dogs in New York city, data on reported dog bites in New York city, and geographical data for New York city at the zip code level.

Installation

`nycdogs` is a data package, bundling several datasets into a convenient format. The relatively large size of the data in the package means it is not suitable for hosting on CRAN, the core R package repository. There are two ways to install it.

Install direct from GitHub

You can install the beta version of nycdogs from GitHub with:

```r
devtools::install_github("kjhealy/nycdogs")
```

Installation using `drat`

While using `install_github()` works just fine, it would be nicer to be able to just type `install.packages("nycdogs")` or `update.packages("nycdogs")` in the ordinary way. We can do this using Dirk Eddelbuettel's drat package. Drat provides a convenient way to make R aware of package repositories other than CRAN.

```r
devtools::install_github("kjhealy/nycdogs")
```

Dogs of New York

```
nyc_license
#> # A tibble: 122,203 x 14
#>    animal_name animal_gender animal_birth_mo… breed_rc borough zip_code
#>    <chr>       <chr>         <date>           <chr>    <chr>      <int>
#>  1 Shadow      M             2000-01-01       Beagle   Brookl…    11236
#>  2 Rocco       M             2011-10-01       Boxer    Brookl…    11210
#>  3 Luigi       M             2005-09-01       Maltese  Bronx      10464
#>  4 Petunia     F             2013-08-01       Pug      Brookl…    11221
#>  5 Romeo       M             2008-10-01       Maltese  Bronx      10451
#>  6 Brandy      M             2004-01-01       Unknown  Brookl…    11225
#>  7 Sam         M             2011-05-01       Pug      Manhat…    10021
#>  8 May         F             2004-05-01       Unknown  Staten…    10305
#>  9 Ruby        F             2010-04-01       Boxer    Brookl…    11220
#> 10 Leo         M             2014-01-01       Beagle   Bronx      10468
#> # … with 122,193 more rows, and 8 more variables:
#> #   community_district <dbl>, census_tract2010 <dbl>, nta <chr>,
#> #   city_council_district <dbl>, congressional_district <dbl>,
#> #   state_senatorial_district <dbl>, license_issued_date <date>,
#> #   license_expired_date <date>
```

Zip Codes of New York

```
> nyc_zips
Simple feature collection with 262 features and 11 fields
geometry type:  POLYGON
dimension:      XY
bbox:           xmin: -74.25576 ymin: 40.49584 xmax: -73.6996 ymax: 40.91517
epsg (SRID):    4326
proj4string:    +proj=longlat +datum=WGS84 +no_defs
# A tibble: 262 x 12
   objectid zip_code po_name state borough st_fips cty_fips bld_gpostal_code
      <int>    <int> <chr>   <chr> <chr>   <chr>   <chr>               <int>
 1        1    11372 Jackso… NY    Queens  36      081                     0
 2        2    11004 Glen O… NY    Queens  36      081                     0
 3        3    11040 New Hy… NY    Queens  36      081                     0
 4        4    11426 Beller… NY    Queens  36      081                     0
 5        5    11365 Fresh … NY    Queens  36      081                     0
 6        6    11373 Elmhur… NY    Queens  36      081                     0
 7        7    11001 Floral… NY    Queens  36      081                     0
 8        8    11375 Forest… NY    Queens  36      081                     0
 9        9    11427 Queens… NY    Queens  36      081                     0
10       10    11374 Rego P… NY    Queens  36      081                     0
# ... with 252 more rows, and 4 more variables: shape_leng <dbl>,
#   shape_area <dbl>, id <chr>, geometry <POLYGON [°]>
```

Zip Codes of New York

```
> nyc_zips
Simple feature collection with 262 features and 11 fields
geometry type:  POLYGON
dimension:      XY
bbox:           xmin: -74.25576 ymin: 40.49584 xmax: -73.6996 ymax: 40.91517
epsg (SRID):    4326
proj4string:    +proj=longlat +datum=WGS84 +no_defs
# A tibble: 262 x 12
   objectid zip_code po_name state borough st_fips cty_fips bld_gpostal_code
      <int>    <int> <chr>   <chr> <chr>   <chr>   <chr>               <int>
 1        1    11372 Jackso… NY    Queens  36      081                     0
 2        2    11004 Glen O… NY    Queens  36      081                     0
 3        3    11040 New Hy… NY    Queens  36      081                     0
 4        4    11426 Beller… NY    Queens  36      081                     0
 5        5    11365 Fresh … NY    Queens  36      081                     0
 6        6    11373 Elmhur… NY    Queens  36      081                     0
 7        7    11001 Floral… NY    Queens  36      081                     0
 8        8    11375 Forest… NY    Queens  36      081                     0
 9        9    11427 Queens… NY    Queens  36      081                     0
10       10    11374 Rego P… NY    Queens  36      081                     0
# ... with 252 more rows, and 4 more variables: shape_leng <dbl>,
#   shape_area <dbl>, id <chr>, geometry <POLYGON [°]>
```

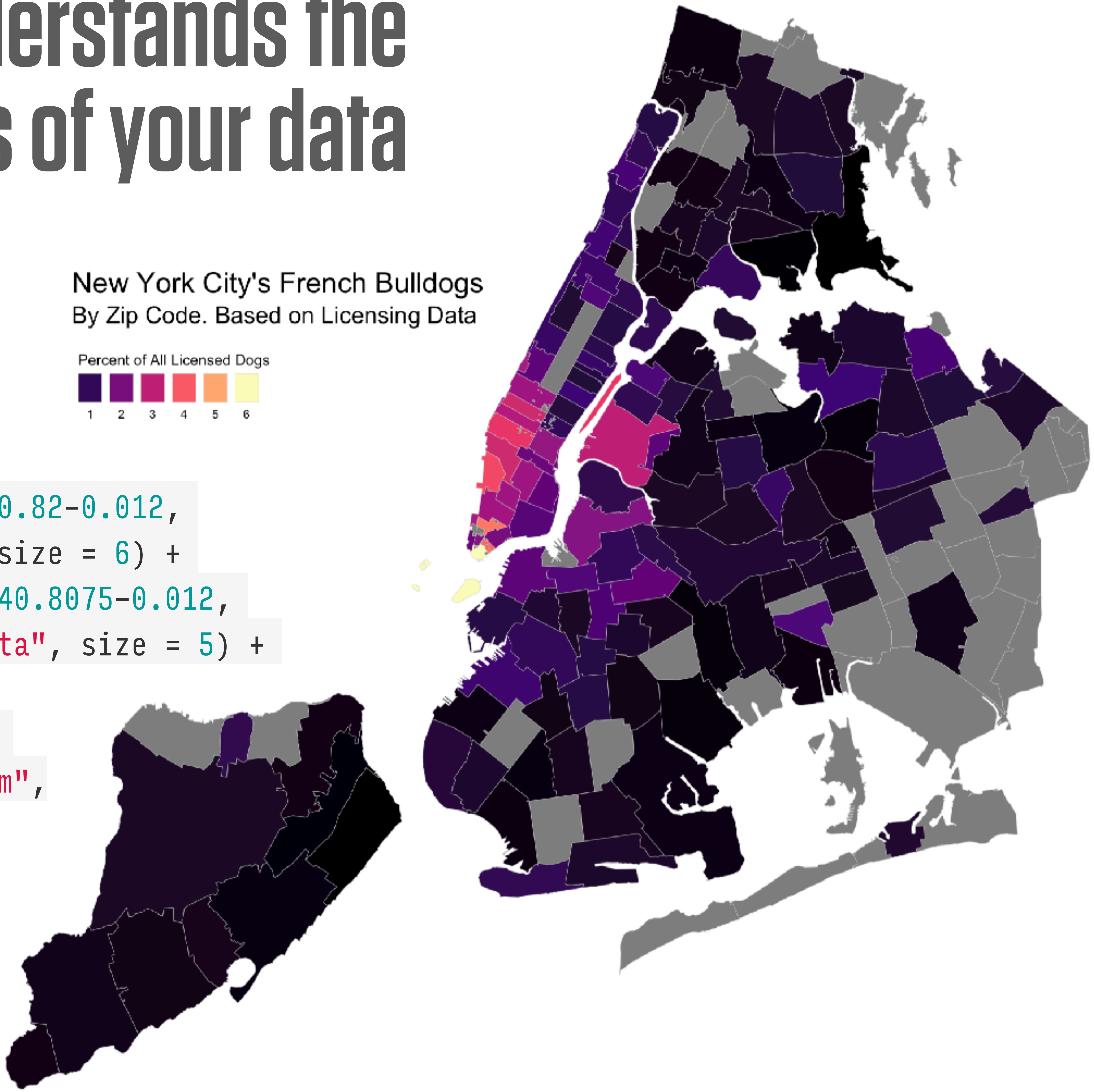
The Relative Distribution of French Bulldogs in New York City, within Zip Code

```
nyc_fb <- nyc_license %>%
    group_by(zip_code, breed_rc) %>%
    tally() %>%
    mutate(freq = n / sum(n),
           pct = round(freq*100, 2)) %>%
    filter(breed_rc == "French Bulldog")

nyc_fb
#> # A tibble: 148 x 5
#> # Groups:   zip_code [148]
#>    zip_code breed_rc           n   freq   pct
#>       <int> <chr>          <int>  <dbl> <dbl>
#>  1     7030 French Bulldog     1 0.5    50
#>  2    10001 French Bulldog    32 0.0354  3.54
#>  3    10002 French Bulldog    18 0.0159  1.59
#>  4    10003 French Bulldog    40 0.0259  2.59
#>  5    10004 French Bulldog     8 0.0602  6.02
#>  6    10005 French Bulldog    12 0.0415  4.15
#>  7    10006 French Bulldog     3 0.0221  2.21
#>  8    10007 French Bulldog     9 0.0437  4.37
#>  9    10009 French Bulldog    37 0.0175  1.75
#> 10    10010 French Bulldog    22 0.0191  1.91
#> # â€¦ with 138 more rows

fb_map <- left_join(nyc_zips, nyc_fb)
#> Joining, by = "zip_code"
```

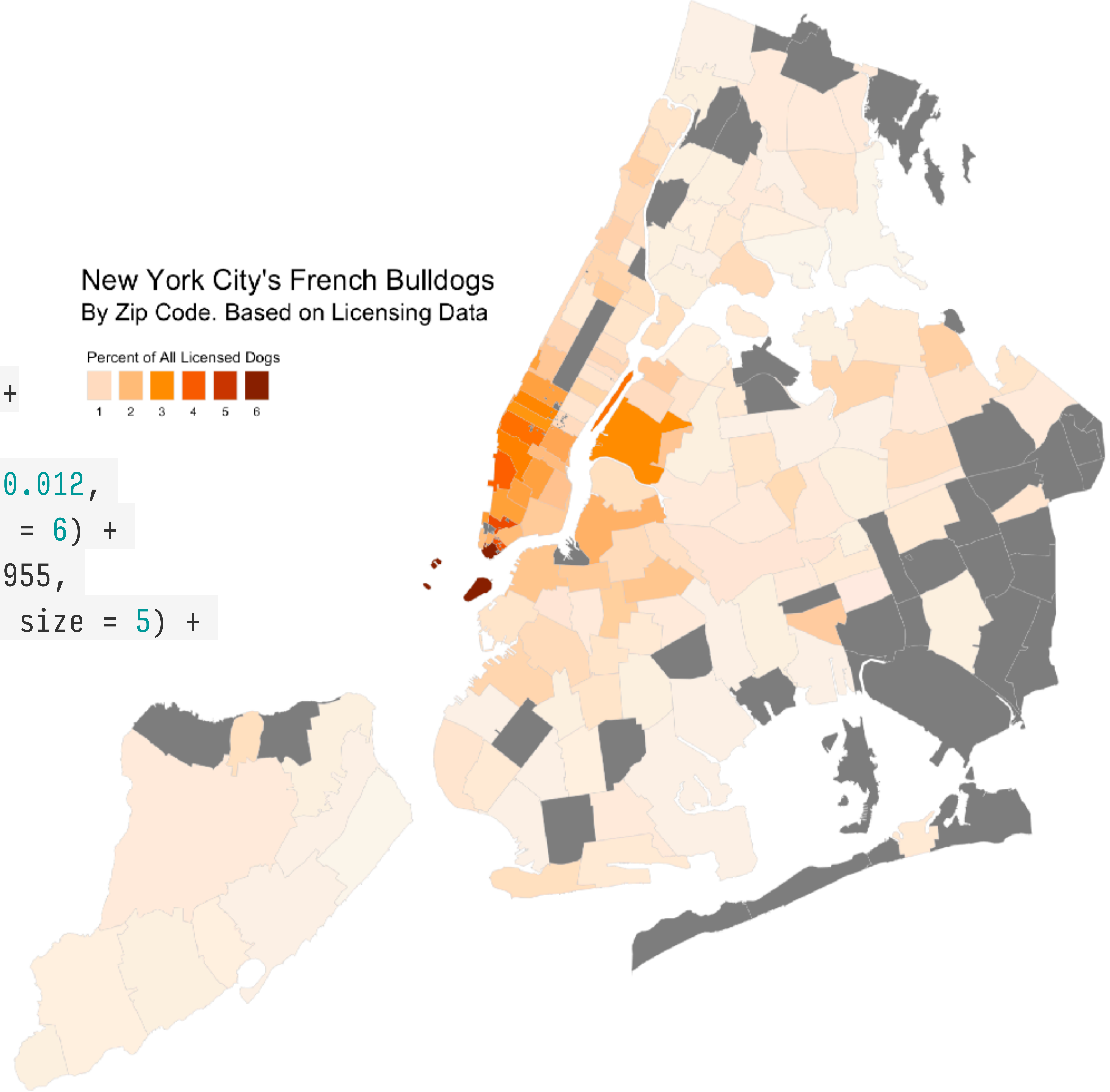
```r
## NYC map theme

theme_nymap <- function(base_size=9, base_family="") {
    require(grid)
    theme_bw(base_size=base_size, base_family=base_family) %+replace%
        theme(axis.line=element_blank(),
              axis.text=element_blank(),
              axis.ticks=element_blank(),
              axis.title=element_blank(),
              panel.background=element_blank(),
              panel.border=element_blank(),
              panel.grid=element_blank(),
              panel.spacing=unit(0, "lines"),
              plot.background=element_blank(),
              legend.justification = c(0,0),
              legend.position = c(0.1, 0.6),
              legend.direction = "horizontal"
        )
}
```

geom_sf() automatically understands the spatial features of your data

```
fb_map %>% ggplot(mapping = aes(fill = pct)) +
    geom_sf(color = "gray80", size = 0.1) +
    scale_fill_viridis_c(option = "A") +
    labs(fill = "Percent of All Licensed Dogs") +
    annotate(geom = "text", x = -74.145 + 0.029, y = 40.82-0.012,
           label = "New York City's French Bulldogs", size = 6) +
    annotate(geom = "text", x = -74.1468 + 0.029, y = 40.8075-0.012,
           label = "By Zip Code. Based on Licensing Data", size = 5) +
    theme_nymap() +
    guides(fill = guide_legend(title.position = "top",
                               label.position = "bottom",
                             keywidth = 1, nrow = 1))
```



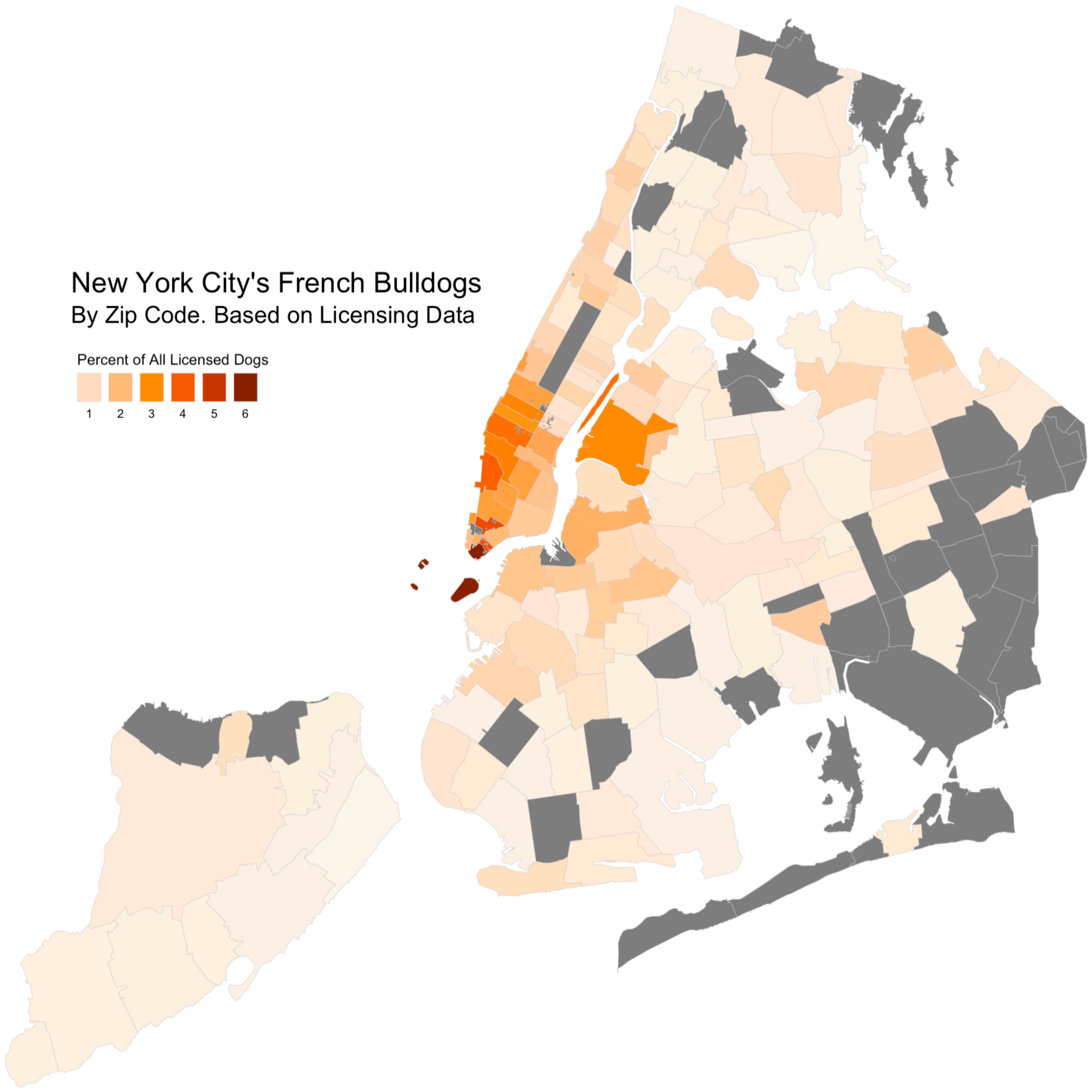
```r
library(colorspace)

fb_map %>% ggplot(mapping = aes(fill = pct)) +
    geom_sf(color = "gray80", size = 0.1) +
    scale_fill_continuous_sequential(palette = "Oranges") +
    labs(fill = "Percent of All Licensed Dogs") +
   annotate(geom = "text", x = -74.145 + 0.029, y = 40.82-0.012,
           label = "New York City's French Bulldogs", size = 6) +
    annotate(geom = "text", x = -74.1468 + 0.029, y = 40.7955,
           label = "By Zip Code. Based on Licensing Data", size = 5) +
    theme_nymap() +
    guides(fill = guide_legend(title.position = "top",
                              label.position = "bottom",
                            keywidth = 1, nrow = 1))
```




New York City's French Bulldogs
By Zip Code. Based on Licensing Data
Percent of All Licensed Dogs
1
2
3
4
5
6

```r
nyc_fb <- nyc_license %>%
    group_by(zip_code, breed_rc) %>%
    tally() %>%
    mutate(freq = n / sum(n),
           pct = round(freq*100, 2)) %>%
    filter(breed_rc == "French Bulldog")
```

```
> nyc_fb
# A tibble: 148 x 5
# Groups:   zip_code [148]
   zip_code breed_rc           n   freq   pct
      <int> <chr>          <int>  <dbl> <dbl>
 1     7030 French Bulldog     1 0.5    50
 2    10001 French Bulldog    32 0.0354  3.54
 3    10002 French Bulldog    18 0.0159  1.59
 4    10003 French Bulldog    40 0.0259  2.59
 5    10004 French Bulldog     8 0.0602  6.02
 6    10005 French Bulldog    12 0.0415  4.15
 7    10006 French Bulldog     3 0.0221  2.21
 8    10007 French Bulldog     9 0.0437  4.37
 9    10009 French Bulldog    37 0.0175  1.75
10    10010 French Bulldog    22 0.0191  1.91
# … with 138 more rows
```

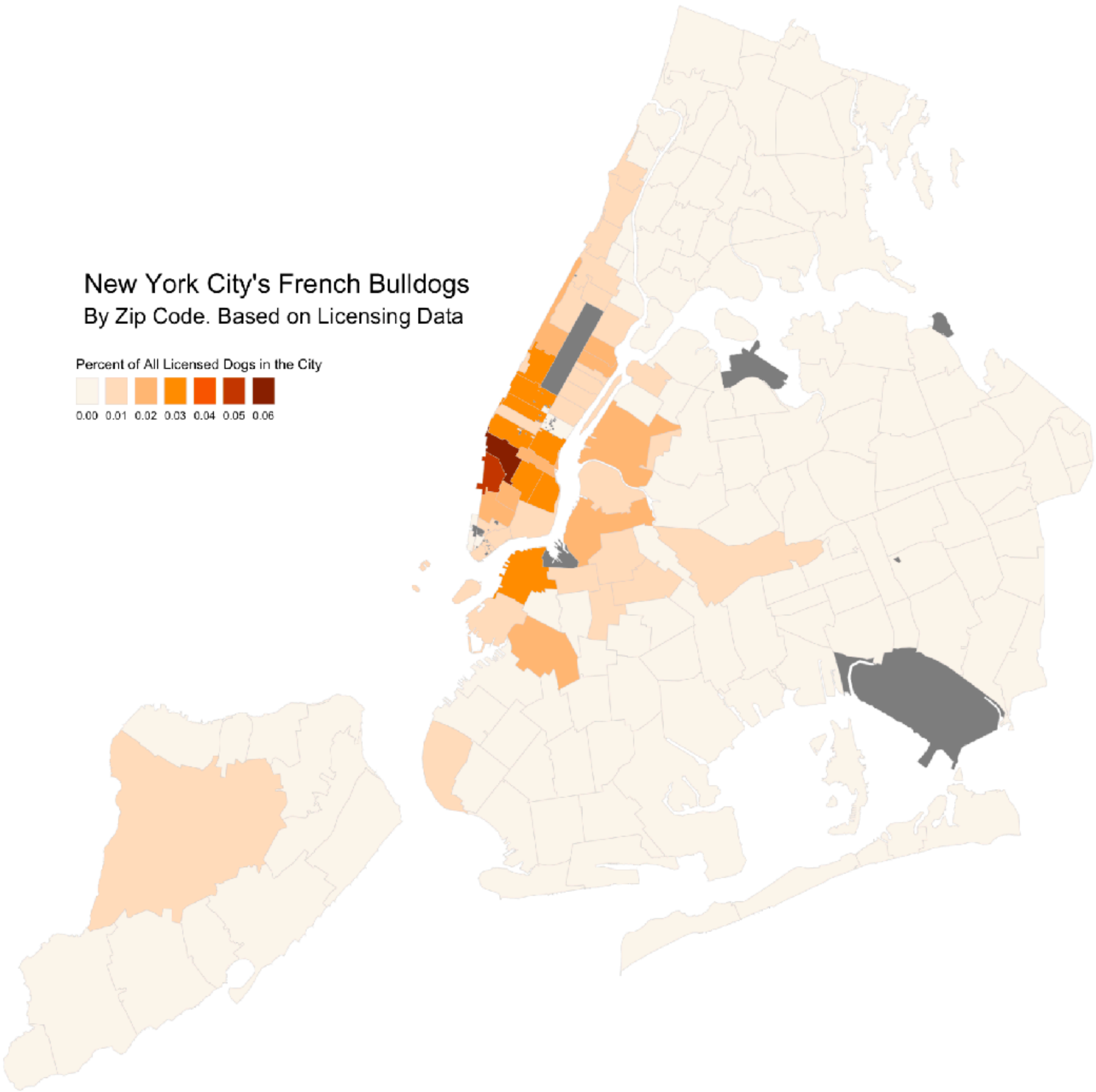

New York City's French Bulldogs
By Zip Code. Based on Licensing Data
Percent of All Licensed Dogs
1
2
3
4
5
6

The Relative Distribution of French Bulldogs in New York City, by Zip Code

```r
library(colorspace)

nyc_fb <- nyc_license %>%
    group_by(zip_code, breed_rc) %>%
    tally() %>%
    ungroup() %>%
    complete(zip_code, breed_rc,
             fill = list(n = 0)) %>%
    mutate(freq = n / sum(n),
           pct = round(freq*100, 2)) %>%
    filter(breed_rc == "French Bulldog")
```

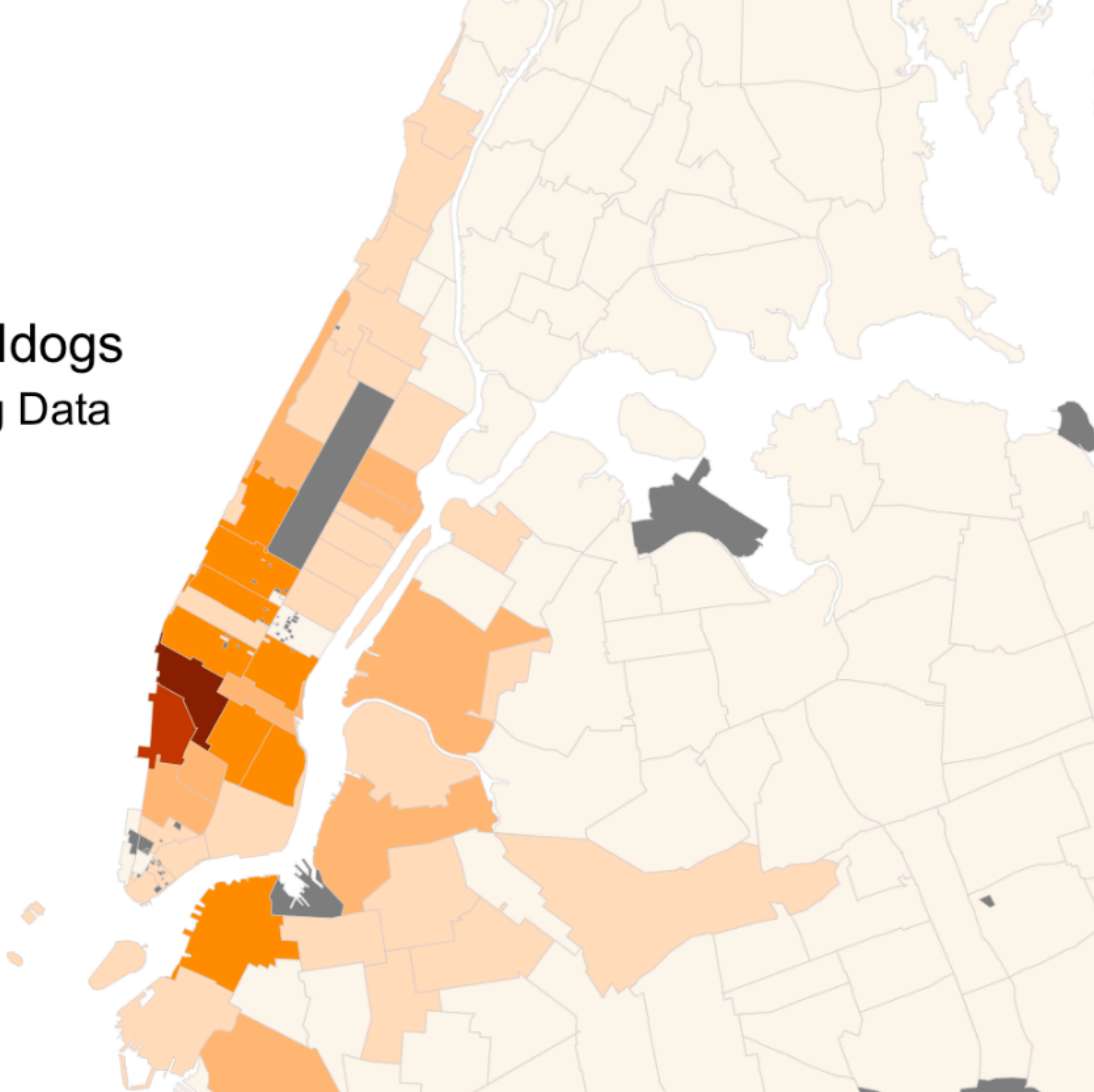
```r
nyc_fb
#> # A tibble: 358 x 5
#>    zip_code breed_rc           n       freq   pct
#>       <int> <chr>          <dbl>      <dbl> <dbl>
#>  1      121 French Bulldog     0 0              0
#>  2     1003 French Bulldog     0 0              0
#>  3     1175 French Bulldog     0 0              0
#>  4     7013 French Bulldog     0 0              0
#>  5     7030 French Bulldog     1 0.00000818     0
#>  6     7093 French Bulldog     0 0              0
#>  7     7302 French Bulldog     0 0              0
#>  8     7624 French Bulldog     0 0              0
#>  9     7675 French Bulldog     0 0              0
#> 10     8110 French Bulldog     0 0              0
#> # … with 348 more rows
```

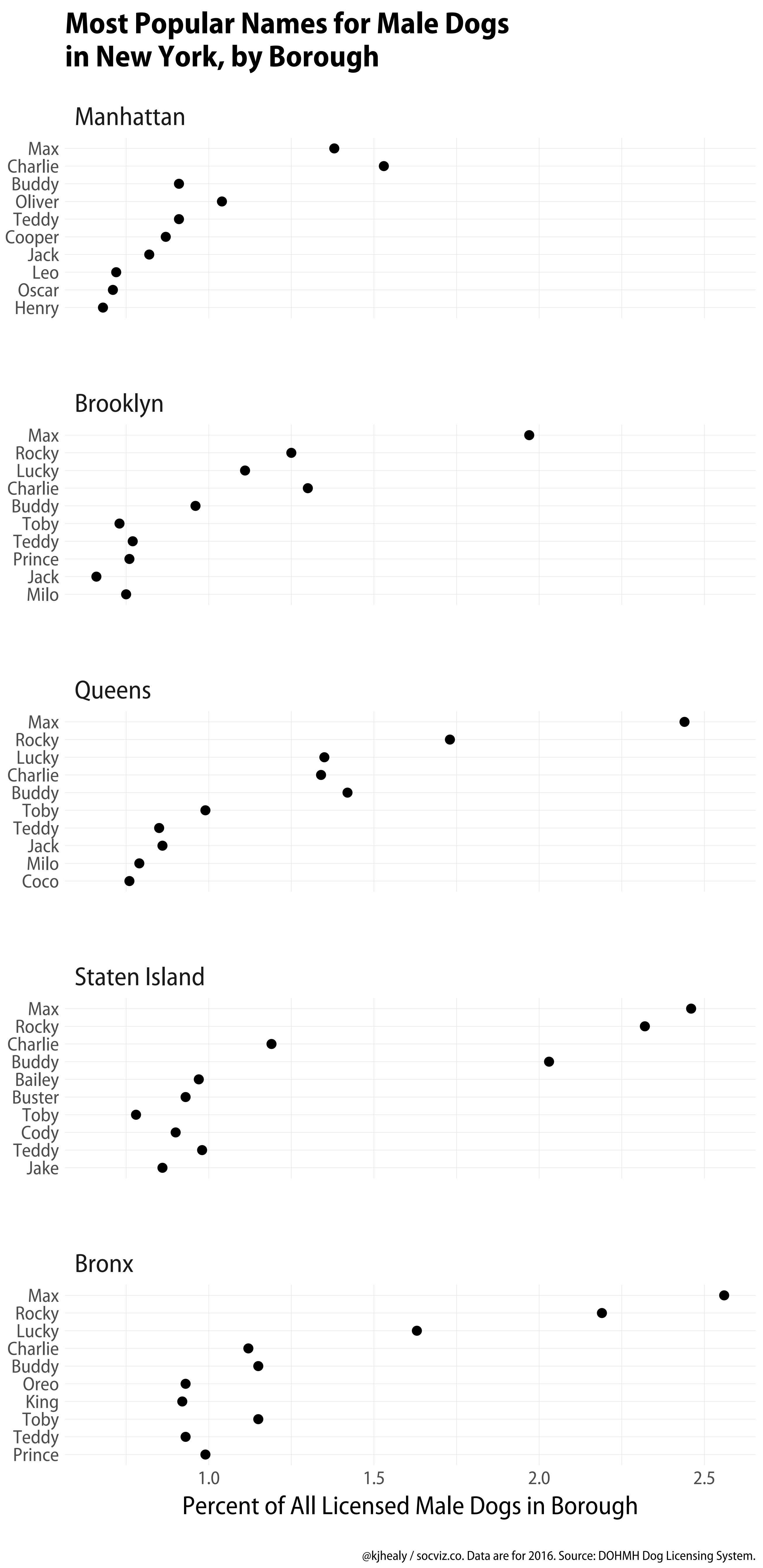
```r
library(colorspace)

fb_map <- left_join(nyc_zips, nyc_fb)
#> Joining, by = "zip_code"

fb_map %>% ggplot(mapping = aes(fill = pct)) +
    geom_sf(color = "gray80", size = 0.1) +
    scale_fill_continuous_sequential(palette = "Oranges") +
   labs(fill = "Percent of All Licensed Dogs in the City") +
  annotate(geom = "text", x = -74.145 + 0.029, y = 40.82-0.012,
           label = "New York City's French Bulldogs", size = 6) +
  annotate(geom = "text", x = -74.1468 + 0.029, y = 40.8075-0.012,
           label = "By Zip Code. Based on Licensing Data", size = 5) +
    theme_nymap() +
    guides(fill = guide_legend(title.position = "top",
                               label.position = "bottom",
                             keywidth = 1, nrow = 1))
```

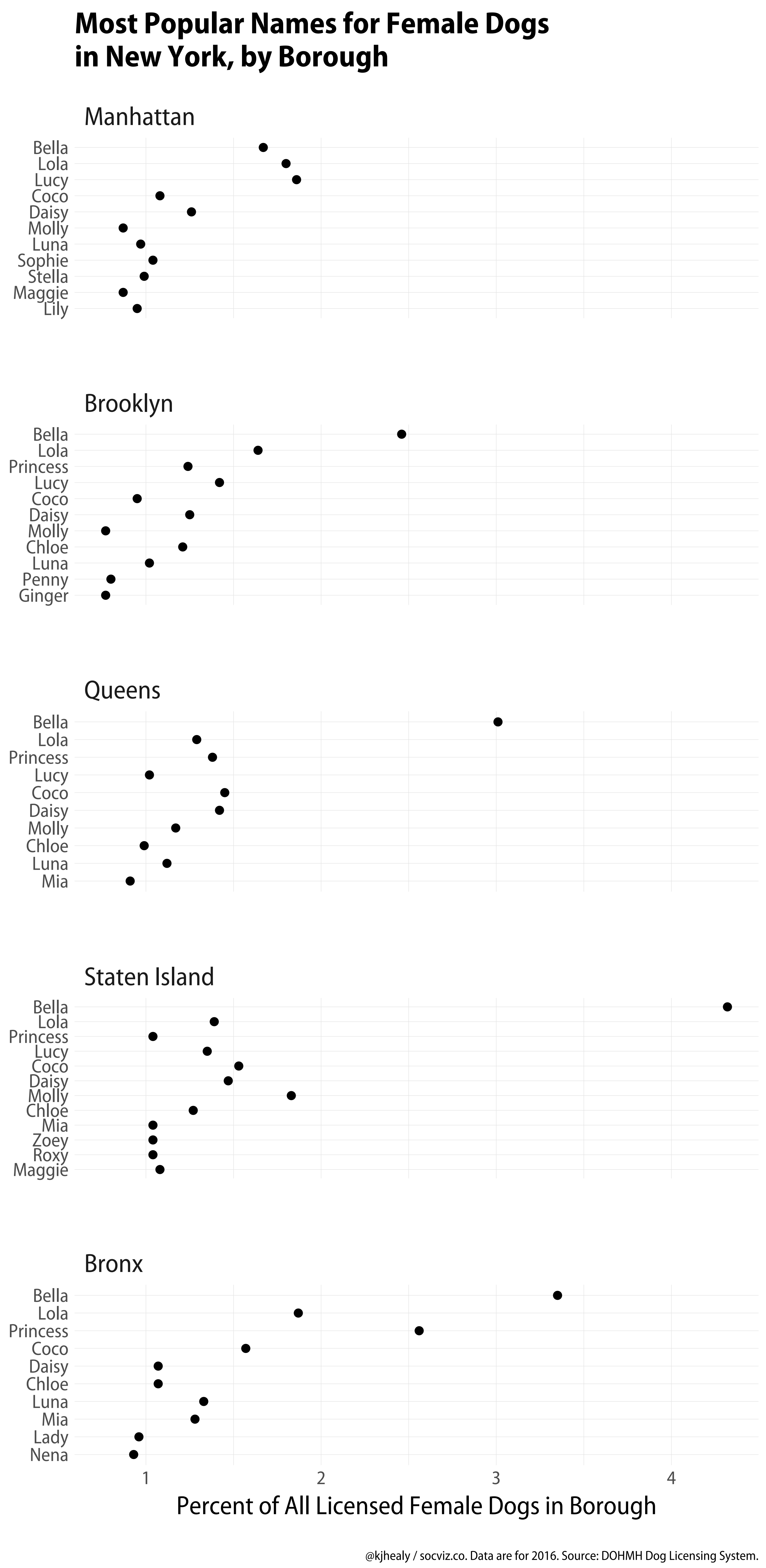

New York City's French Bulldogs
By Zip Code. Based on Licensing Data
Percent of All Licensed Dogs in the City
0.00
0.01
0.02
0.03
0.04
0.05
0.06

New York City's French Bulldogs

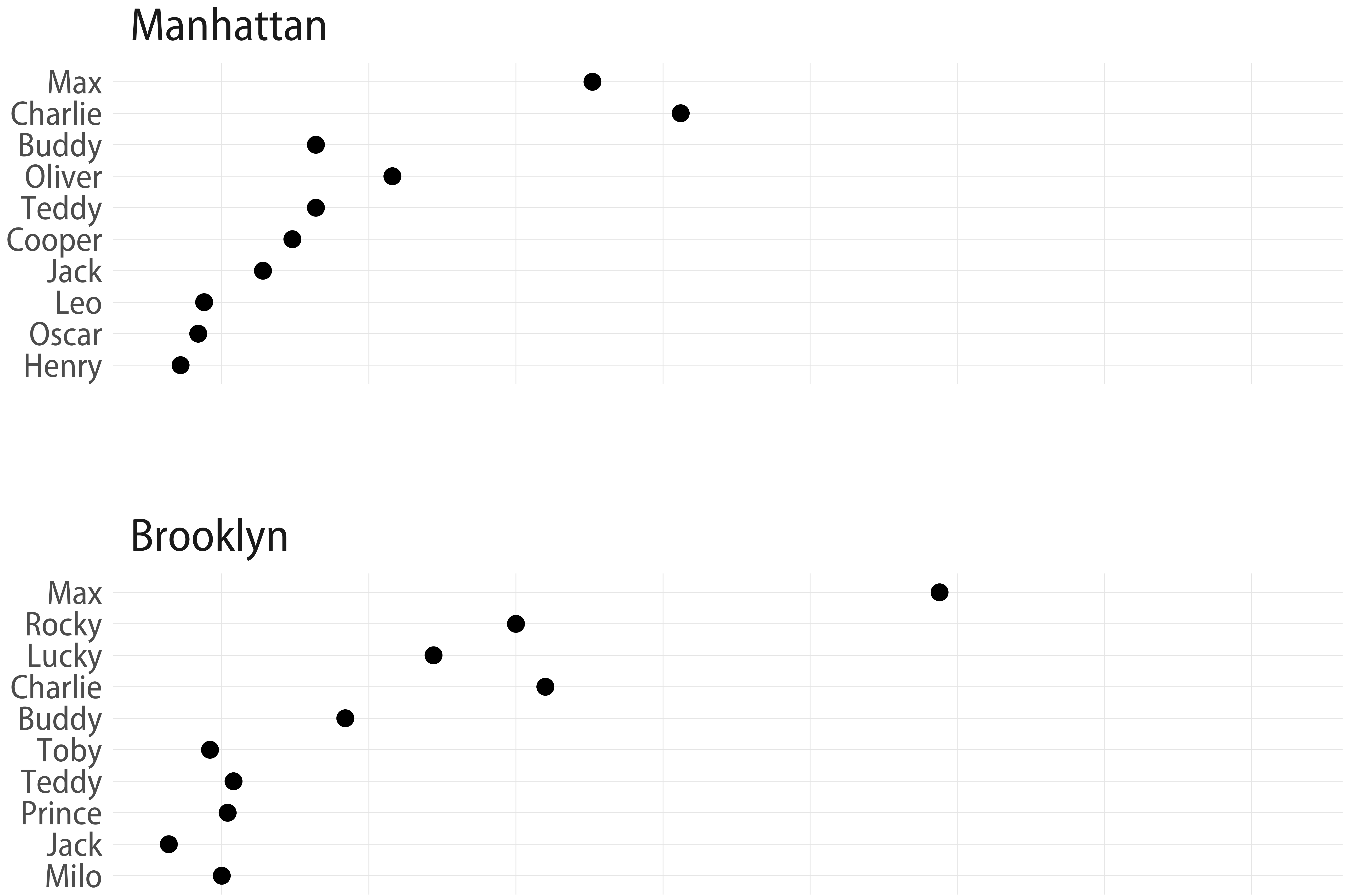
By Zip Code. Based on Licensing Data

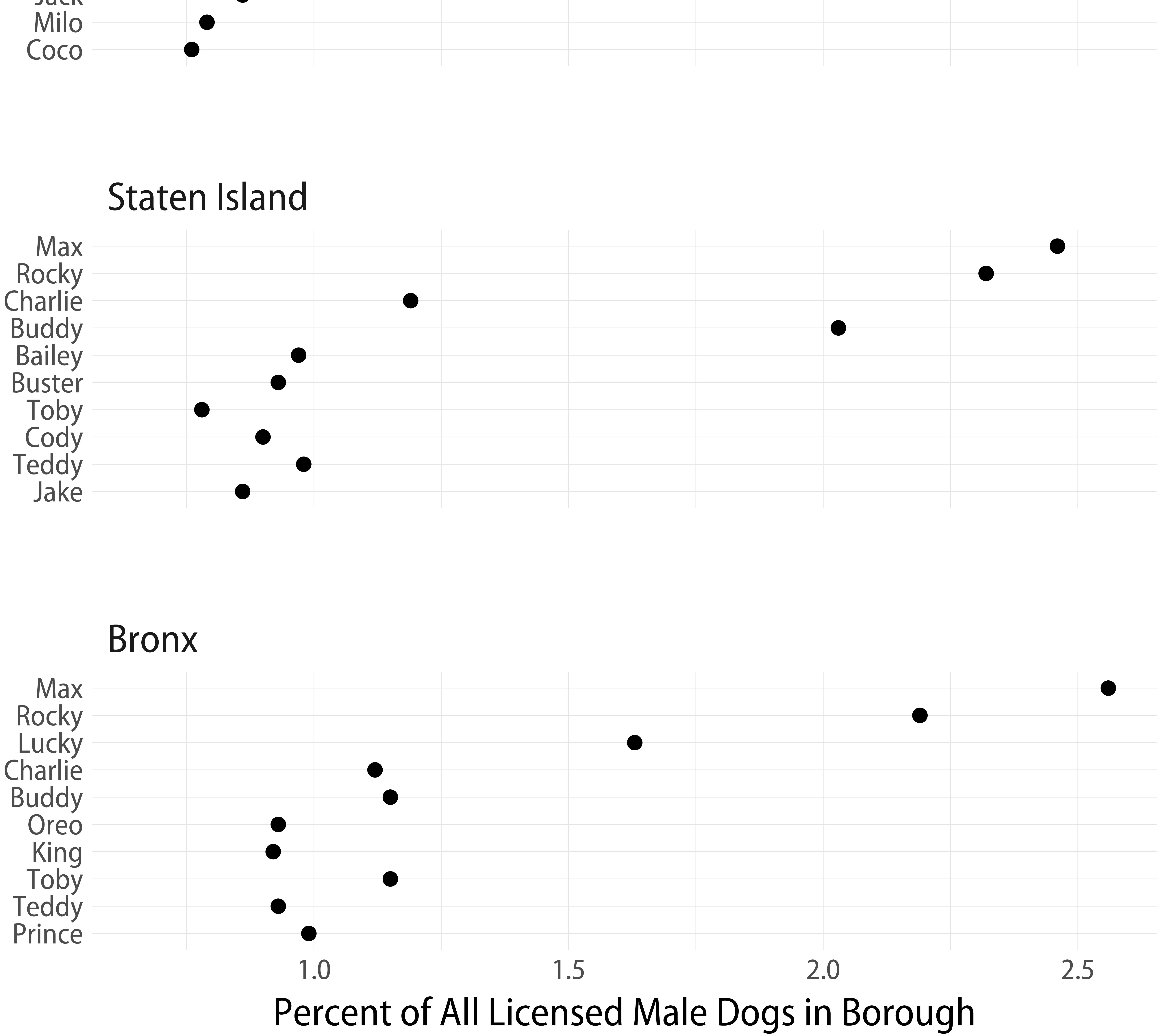



Most Popular Names for Male Dogs
in New York, by Borough
Manhattan
Max
Charlie
Buddy
Oliver
Teddy
Cooper
Jack
Leo
Oscar
Henry
Brooklyn
Max
Rocky
Lucky
Charlie
Buddy
Toby
Teddy
Prince
Jack
Milo
Queens
Max
Rocky
Lucky
Charlie
Buddy
Toby
Teddy
Jack
Milo
Coco
Staten Island
Max
Rocky
Charlie
Buddy
Bailey
Buster
Toby
Cody
Teddy
Jake
Bronx
Max
Rocky
Lucky
Charlie
Buddy
Oreo
King
Toby
Teddy
Prince
1.0
1.5
2.0
2.5
Percent of All Licensed Male Dogs in Borough
@kjhealy / socviz.co. Data are for 2016. Source: DOHMH Dog Licensing System.


Most Popular Names for Female Dogs
in New York, by Borough
Manhattan
Bella
Lola
Lucy
Coco
Daisy
Molly
Luna
Sophie
Stella
Maggie
Lily
Brooklyn
Bella
Lola
Princess
Lucy
Coco
Daisy
Molly
Chloe
Luna
Penny
Ginger
Queens
Bella
Lola
Princess
Lucy
Coco
Daisy
Molly
Chloe
Luna
Mia
Staten Island
Bella
Lola
Princess
Lucy
Coco
Daisy
Molly
Chloe
Mia
Zoey
Roxy
Maggie
Bronx
Bella
Lola
Princess
Coco
Daisy
Chloe
Luna
Mia
Lady
Nena
1
2
3
4
Percent of All Licensed Female Dogs in Borough
@kjhealy / socviz.co. Data are for 2016. Source: DOHMH Dog Licensing System.

Most Popular Names for Male Dogs in New York, by Borough




Milo
Coco
Staten Island
Max
Rocky
Charlie
Buddy
Bailey
Buster
Toby
Cody
Teddy
Jake
Bronx
Max
Rocky
Lucky
Charlie
Buddy
Oreo
King
Toby
Teddy
Prince
1.0
1.5
2.0
2.5
Percent of All Licensed Male Dogs in Borough