Discovery, 2

Frank Edwards

10/23/2019

Network data

- Network data captures relationships between people, groups, families, or any set of nodes
- It is unlike typical unit-variable data we have been working with
- Network analysis is a powerful tool in criminology: criminal relationships, policy diffusion, policing practices as partner relations...

Analyzing network data in R

```
library(igraph)
library(GGally)
library(network)
## load data
data(florentine)
## set row names from variable
row.names(florentine) <- florentine$FAMILY
## drop family variable
florentine <- florentine %>% select(-FAMILY)
```

Adjacency matrix: undirected

- An adjacency matrix captures the presence or absence of ties across units.
- Directed networks have direction, with senders and receivers of ties
- Undirected networks do not have direction, they simply indicate the presence of ties

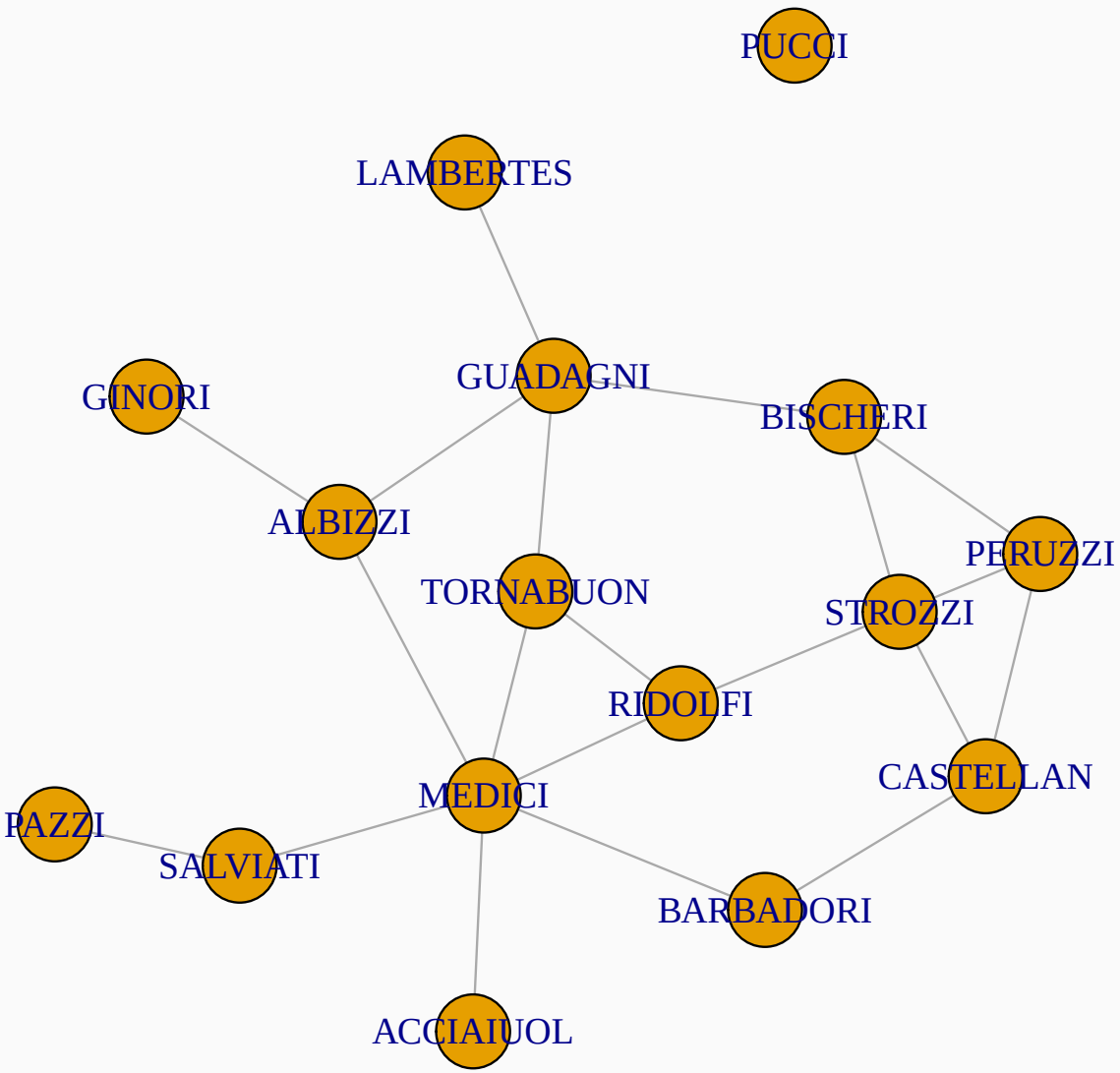
```
florentine <- as.matrix(florentine)
florentine[1:6, 1:6]
```

```
##           ACCIAIUOL ALBIZZI BARBADORI BISCHERI CASTELLAN GINORI
## ACCIAIUOL         0       0         0        0         0      0
## ALBIZZI           0       0         0        0         0      1
## BARBADORI         0       0         0        0         1      0
## BISCHERI          0       0         0        0         0      0
## CASTELLAN         0       0         1        0         0      0
## GINORI            0       1         0        0         0      0
```

```
florence <- graph.adjacency(florentine, mode = "undirected", diag = FALSE)
```

Visualizing a network graph

```
plot(florence)
```



Quantifying network graphs: centrality

How connected is each node to each other node?

- Degree: the number of ties to each node

```
degree(florence)
```

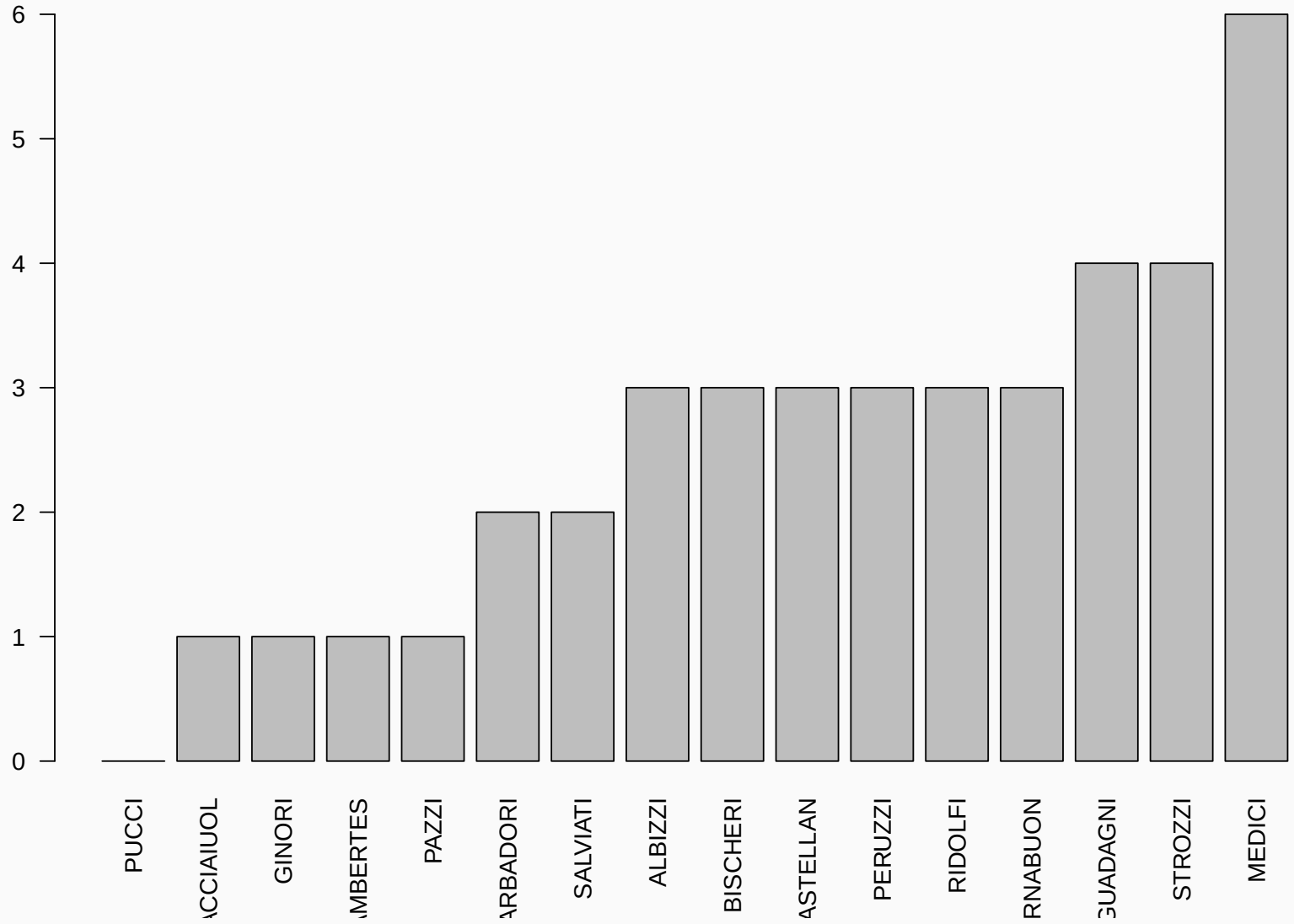
```
## ACCIAIUOL   ALBIZZI BARBADORI  BISCHERI CASTELLAN    GINORI  GUADAGNI
##         1         3         2         3         3         1         4
## LAMBERTES    MEDICI     PAZZI   PERUZZI     PUCCI   RIDOLFI  SALVIATI
##         1         6         1         3         0         3         2
##   STROZZI TORNABUON
##         4         3
```

- Farness: how far is each node from each other node in the graph
- Closeness: How close, on average, is each node to each other node: $\frac{1}{farness}$

```
closeness(florence)
```

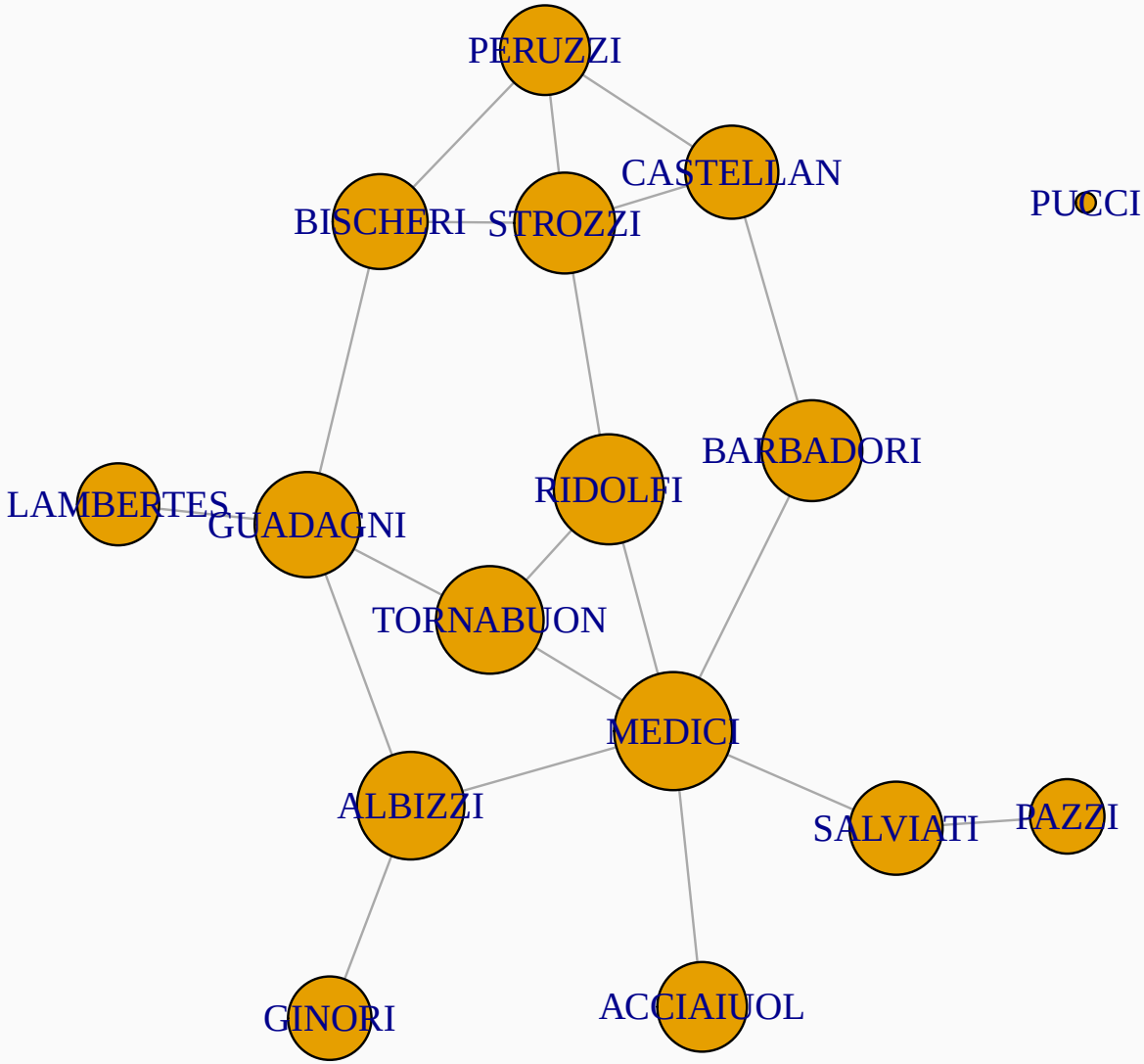
```
##   ACCIAIUOL     ALBIZZI   BARBADORI    BISCHERI   CASTELLAN      GINORI
## 0.018518519 0.022222222 0.020833333 0.019607843 0.019230769 0.017241379
##    GUADAGNI   LAMBERTES      MEDICI       PAZZI     PERUZZI       PUCCI
## 0.021739130 0.016949153 0.024390244 0.015384615 0.018518519 0.004166667
##     RIDOLFI    SALVIATI     STROZZI   TORNABUON
```

Degree


6
5
4
3
2
1
0
PUCCI
ACCIAIUOL
GINORI
MBERTES
PAZZI
ARBADORI
SALVIATI
ALBIZZI
BISCHERI
ASTELLAN
PERUZZI
RIDOLFI
RNABUON
GUADAGNI
STROZZI
MEDICI

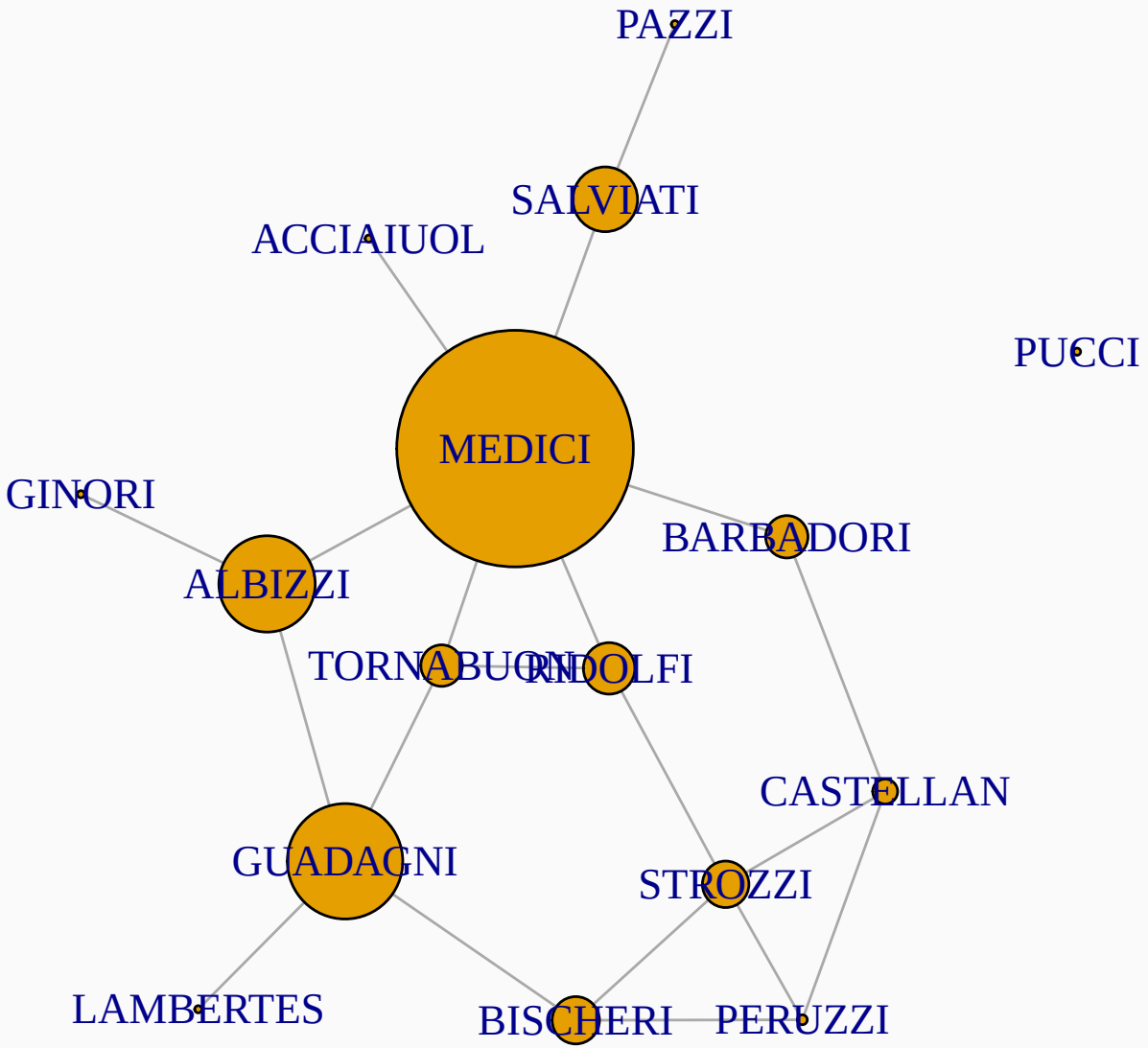
Closeness

```
plot(florence, vertex.size = closeness(florence) * 1000)
```



Betweeness: Does a node connect other nodes?

```
plot(florence, vertex.size = betweenness(florence))
```



Directed networks: Twitter

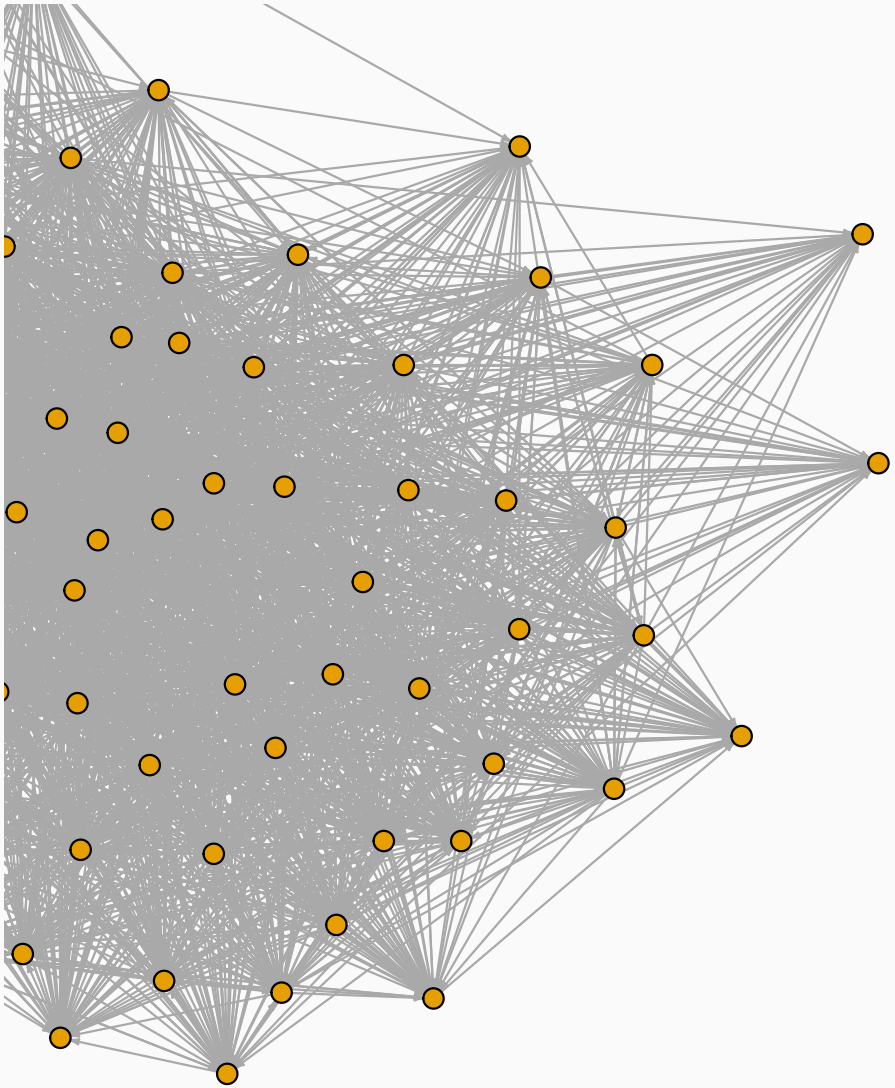
Twitter networks have direction: followers, followed

```
data("twitter.following")
data("twitter.senator")

twitter.following <- graph.edgelist(as.matrix(twitter.following))

twitter.following[1:5, 1:5]
```

```
## 5 x 5 sparse Matrix of class "dgCMatrix"
##                 SenAlexander RoyBlunt SenatorBurr JohnBoozman
## SenAlexander               .        1           1           1
## RoyBlunt                   1        .           1           1
## SenatorBurr                .        1           .           .
## JohnBoozman                .        1           .           .
## SenJohnBarrasso            1        1           1           1
##                 SenJohnBarrasso
## SenAlexander                  1
## RoyBlunt                      1
## SenatorBurr                   .
## JohnBoozman                   1
## SenJohnBarrasso               .
```

Visualizing a directed network

This is not a great visual...



Add degree counts to profiles

```
twitter.senator <- twitter.senator %>% mutate(indegree = igraph::degree(twitter.following,
    mode = "in"), outdegree = igraph::degree(twitter.following, mode = "out"))

head(twitter.senator)
```

```
##       screen_name            name party state indegree outdegree
## 1    SenAlexander Lamar Alexander     R    TN       52        38
## 2        RoyBlunt       Roy Blunt     R    MO       57        46
## 3    SenatorBoxer   Barbara Boxer     D    CA       54         7
## 4 SenSherrodBrown   Sherrod Brown     D    OH       47        30
## 5     SenatorBurr    Richard Burr     R    NC       55        50
## 6  SenatorBaldwin   Tammy Baldwin     D    WI       43        20
```

Centrality measures: indegree

How many ties/edges come to a node?

```
head(arrange(twitter.senator, desc(indegree)))
```

```
##       screen_name              name party state indegree outdegree
## 1    SenTomCotton        Tom Cotton     R    AR       64        15
## 2   SenatorDurbin Richard J. Durbin     D    IL       60        87
## 3 SenJohnBarrasso     John Barrasso     R    WY       58        79
## 4     SenDonnelly      Joe Donnelly     D    IN       58         9
## 5   SenOrrinHatch    Orrin G. Hatch     R    UT       58        50
## 6        RoyBlunt          Roy Blunt     R    MO       57        46
```

```
head(arrange(twitter.senator, indegree))
```

```
##      screen_name              name party state indegree outdegree
## 1  lisamurkowski     Lisa Murkowski     R    AK        4         0
## 2  SenWhitehouse Sheldon Whitehouse     D    RI       13        59
## 3 McConnellPress    Mitch McConnell     R    KY       17        62
## 4     MarkWarner     Mark R. Warner     D    VA       23        75
## 5 SenatorShaheen     Jeanne Shaheen     D    NH       25        69
## 6  ChuckGrassley     Chuck Grassley     R    IA       27         0
```

Centrality measures: outdegree

How many ties/edges come from a node?

```
head(arrange(twitter.senator, desc(outdegree)))
```

```
##      screen_name              name party state indegree outdegree
## 1 SenThadCochran      Thad Cochran     R    MS       55        89
## 2    SteveDaines      Steve Daines     R    MT       30        88
## 3  SenJohnMcCain       John McCain     R    AZ       41        88
## 4 Sen_JoeManchin  Joe Manchin, III     D    WV       43        88
## 5  SenatorDurbin Richard J. Durbin     D    IL       60        87
## 6  SenDeanHeller       Dean Heller     R    NV       28        87
```

```
head(arrange(twitter.senator, outdegree))
```

```
##       screen_name           name party state indegree outdegree
## 1 SenatorCantwell Maria Cantwell     D    WA       30         0
## 2   ChuckGrassley Chuck Grassley     R    IA       27         0
## 3 SenatorLankford James Lankford     R    OK       41         0
## 4   lisamurkowski Lisa Murkowski     R    AK        4         0
## 5 SenatorSessions  Jeff Sessions     R    AL       46         0
## 6   SenThomTillis    Thom Tillis     R    NC       46         1
```

Next steps

- Networks data can be used in regression: what predicts being well-connected? Do well connected have more X than poorly connected?
- Potential outcomes: degree, betweenness, closeness
- Just scratching the surface!

Spatial data

Understanding spatial data

Place can (generally) be measured two ways:

- By points (latitude, longitude)
- By shape (polygon)

Why use spatial data

- Clusters and patterns may become apparent on a map
- Unordered plotting will miss geographic patterns
- Map making is generally a good idea when working with spatial data

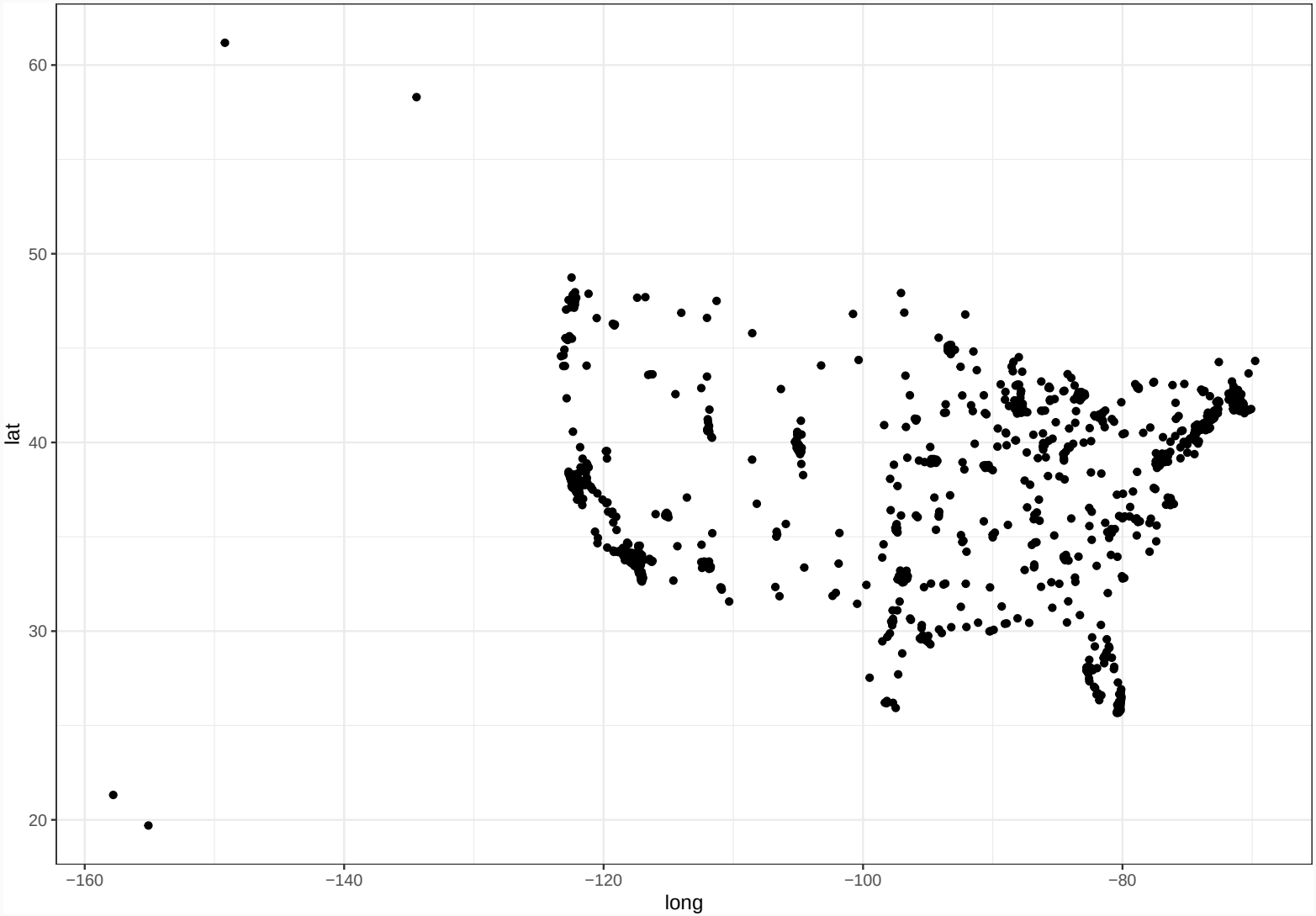
The structure of spatial data

```
library(maps)
data(us.cities)
head(us.cities)
```

```
##         name country.etc    pop   lat    long capital
## 1 Abilene TX          TX 113888 32.45  -99.74       0
## 2   Akron OH          OH 206634 41.08  -81.52       0
## 3 Alameda CA          CA  70069 37.77 -122.26       0
## 4  Albany GA          GA  75510 31.58  -84.18       0
## 5  Albany NY          NY  93576 42.67  -73.80       2
## 6  Albany OR          OR  45535 44.62 -123.09       0
```

What the cities data contain

```
ggplot(us.cities, aes(x = long, y = lat)) + geom_point()
```

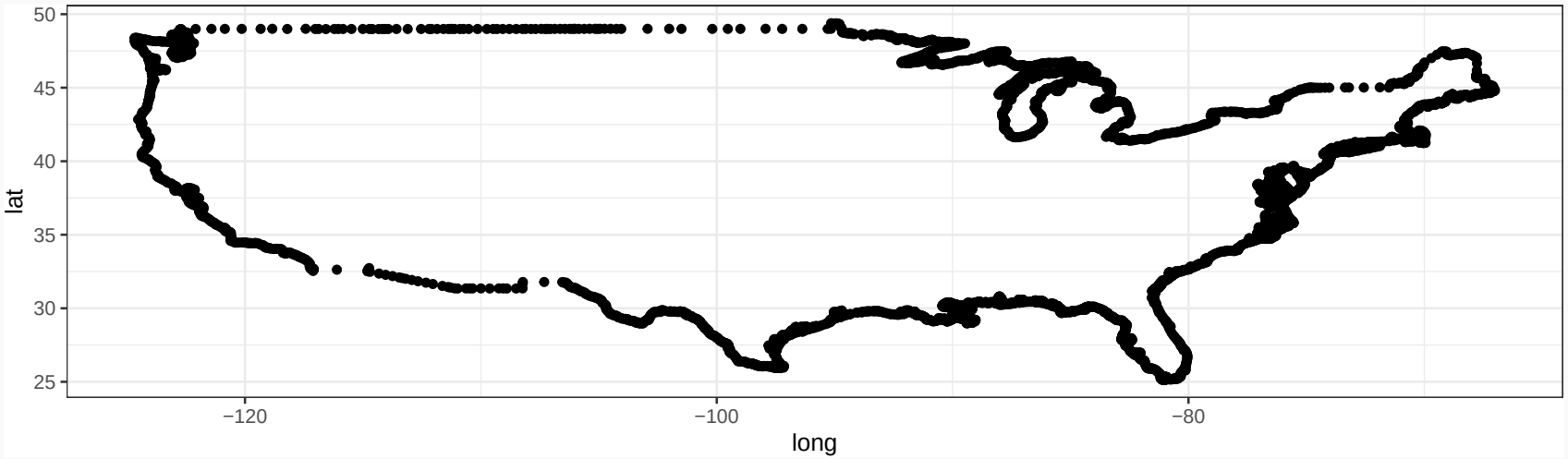


A USA map shape file

```
usa_map <- map_data("usa")
head(usa_map)
```

```
##        long      lat group order region subregion
## 1 -101.4078 29.74224     1     1   main      <NA>
## 2 -101.3906 29.74224     1     2   main      <NA>
## 3 -101.3620 29.65056     1     3   main      <NA>
## 4 -101.3505 29.63911     1     4   main      <NA>
## 5 -101.3219 29.63338     1     5   main      <NA>
## 6 -101.3047 29.64484     1     6   main      <NA>
```

```
ggplot(usa_map, aes(x = long, y = lat)) + geom_point()
```

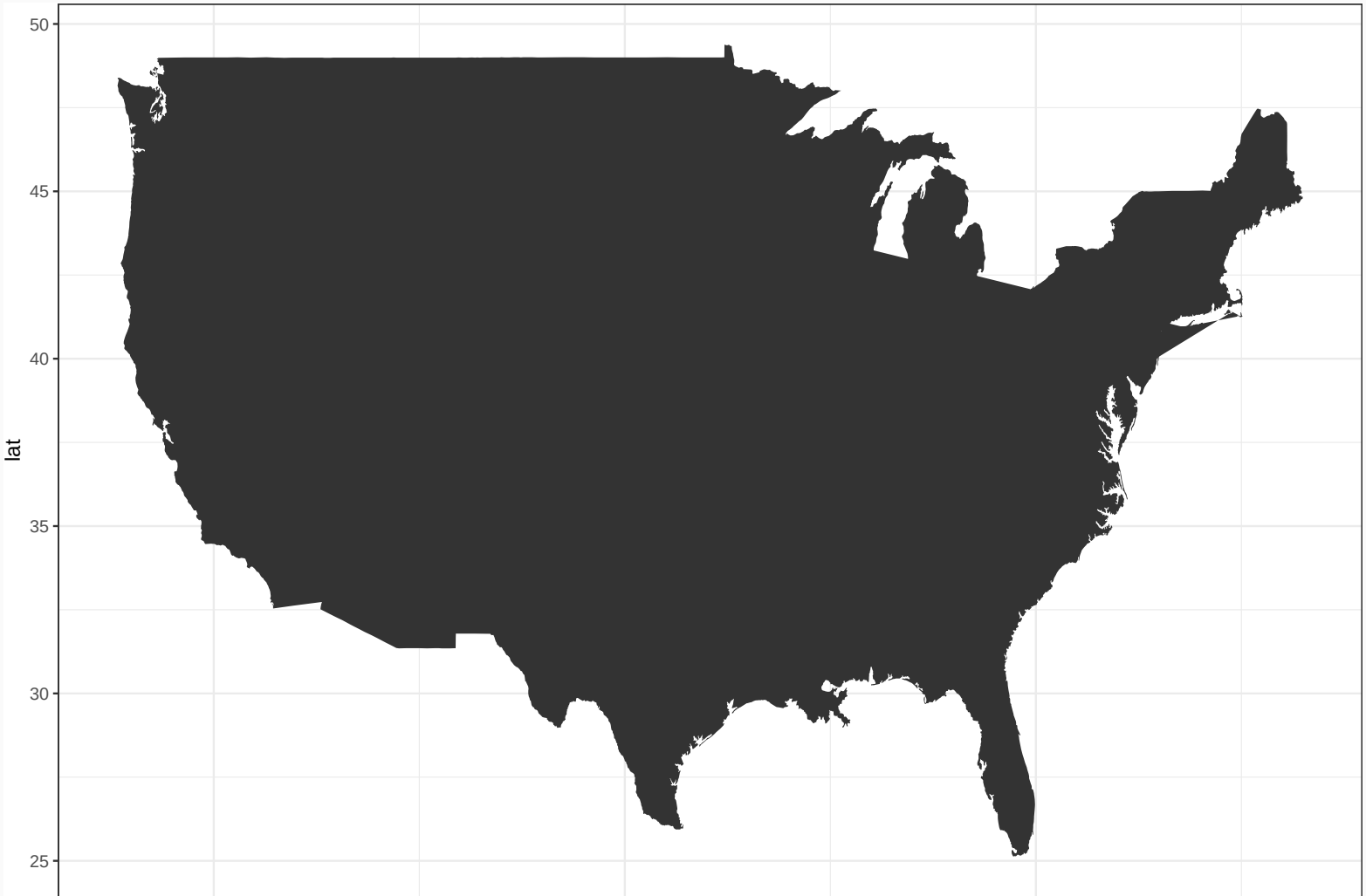


Treating this as a polygon

```
nrow(usa_map)
```

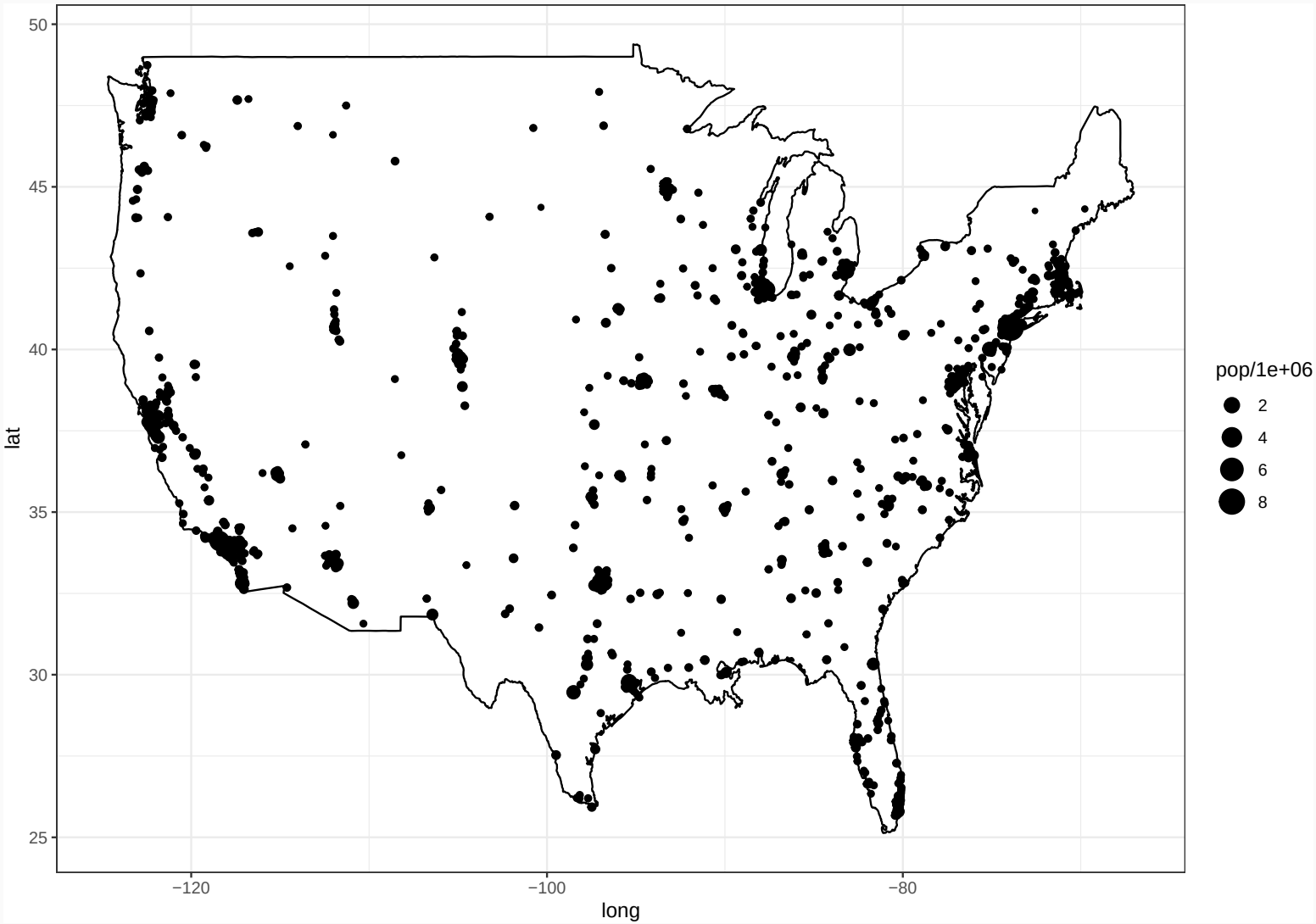
```
## [1] 7243
```

```
ggplot(usa_map, aes(x = long, y = lat)) + geom_polygon()
```



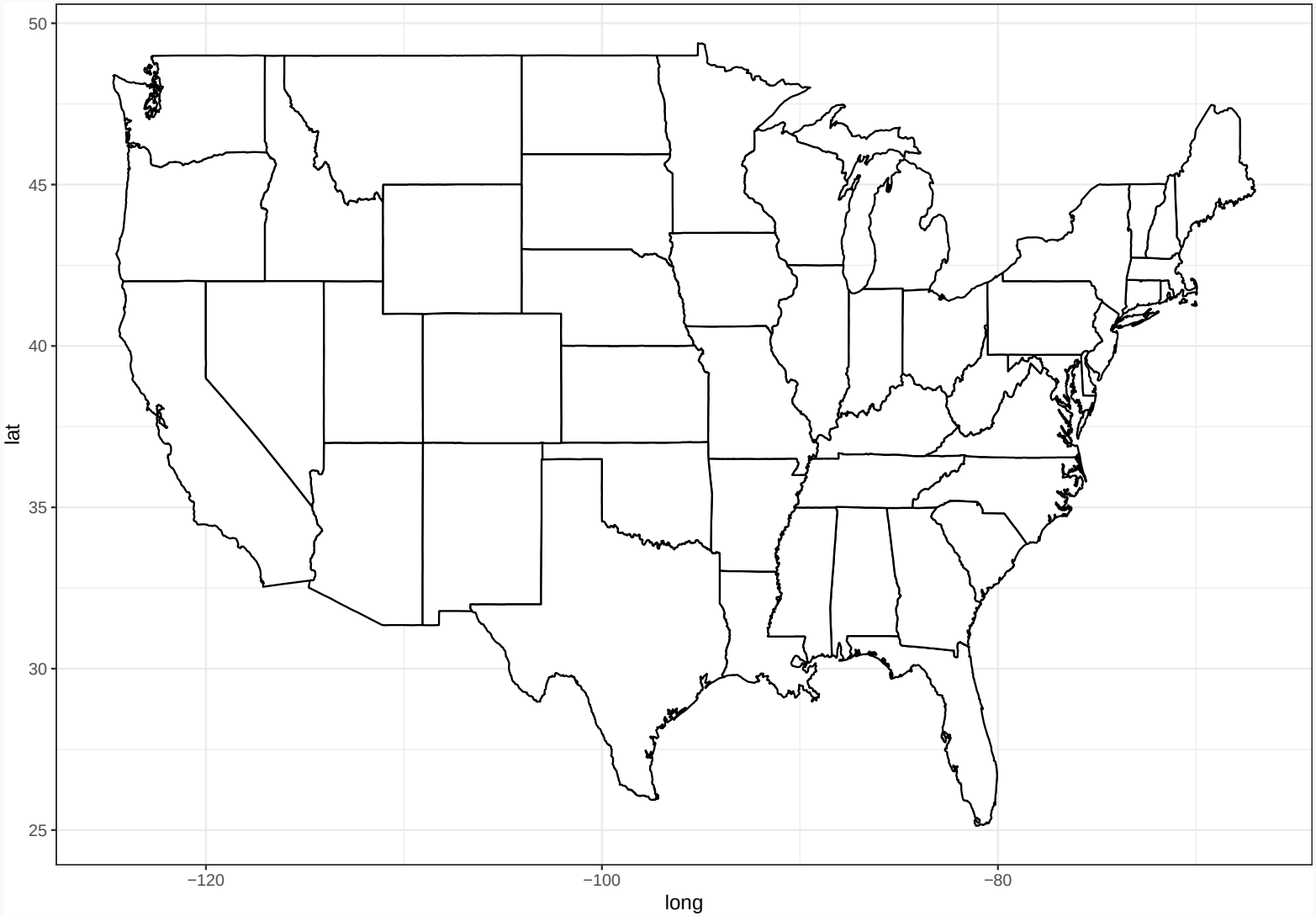
Remove AK, HI and scale by size

```
ggplot() + geom_polygon(data = usa_map, aes(x = long, y = lat, group = group),
    fill = "white", color = "black") + geom_point(aes(x = long, y = lat, size = pop/1e+06),
    data = us.cities %>% filter(!(country.etc %in% c("AK", "HI"))))
```



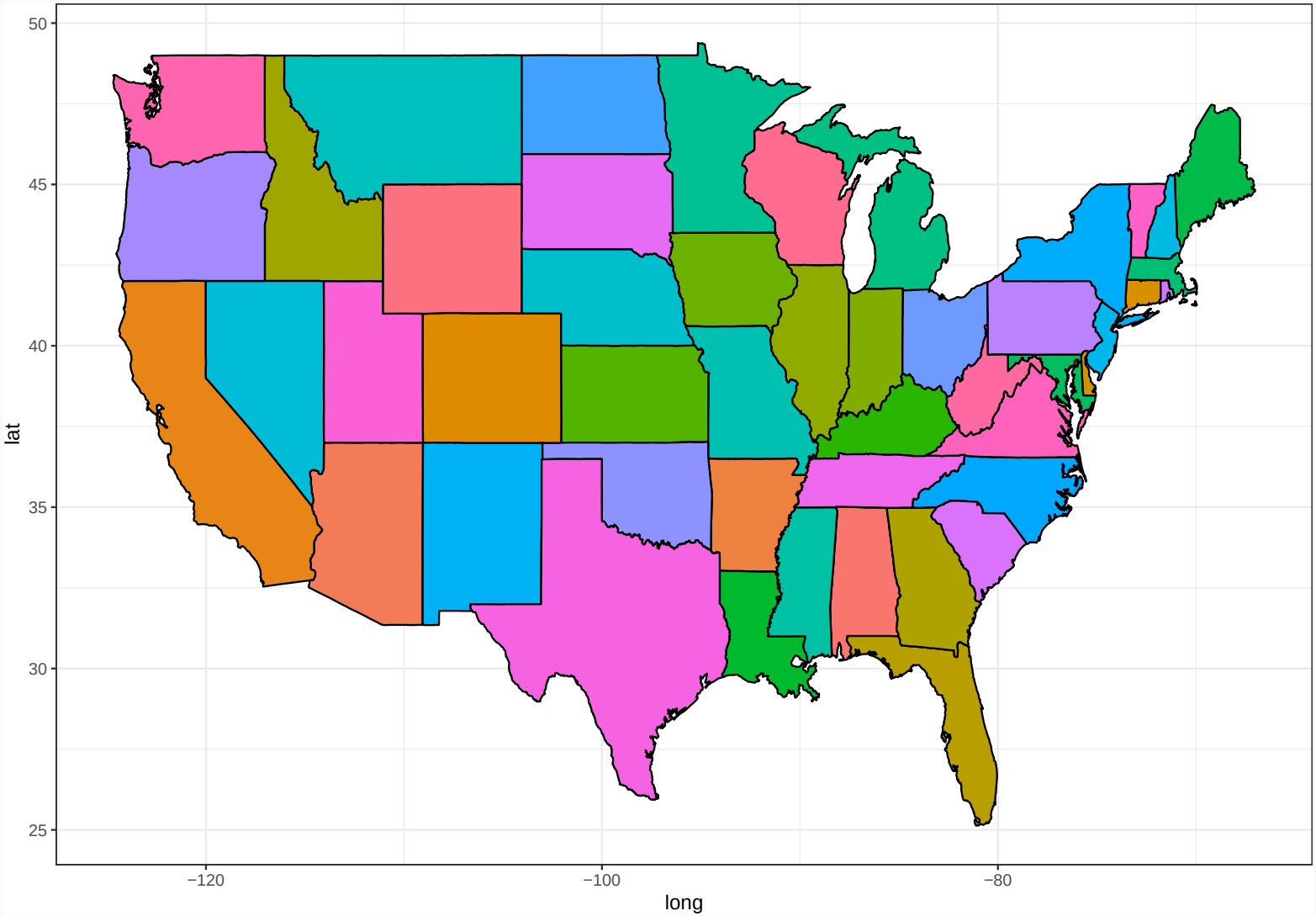
Add states

```
us_states <- map_data("state")
ggplot(us_states, aes(x = long, y = lat, group = group)) + geom_polygon(color = "black",
    fill = "white")
```



Add a fill

```
ggplot(us_states, aes(x = long, y = lat, group = group, fill = region)) + geom_polygon(color = "black") +
    guides(fill = FALSE)
```



Add election data

```
data(pres08)
head(pres08)
```

```
##   state.name state Obama McCain EV
## 1    Alabama    AL    39     60  9
## 2     Alaska    AK    38     59  3
## 3    Arizona    AZ    45     54 10
## 4   Arkansas    AR    39     59  6
## 5 California    CA    61     37 55
## 6   Colorado    CO    54     45  9
```

```
head(us_states)
```

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

How can we join these two data frames?

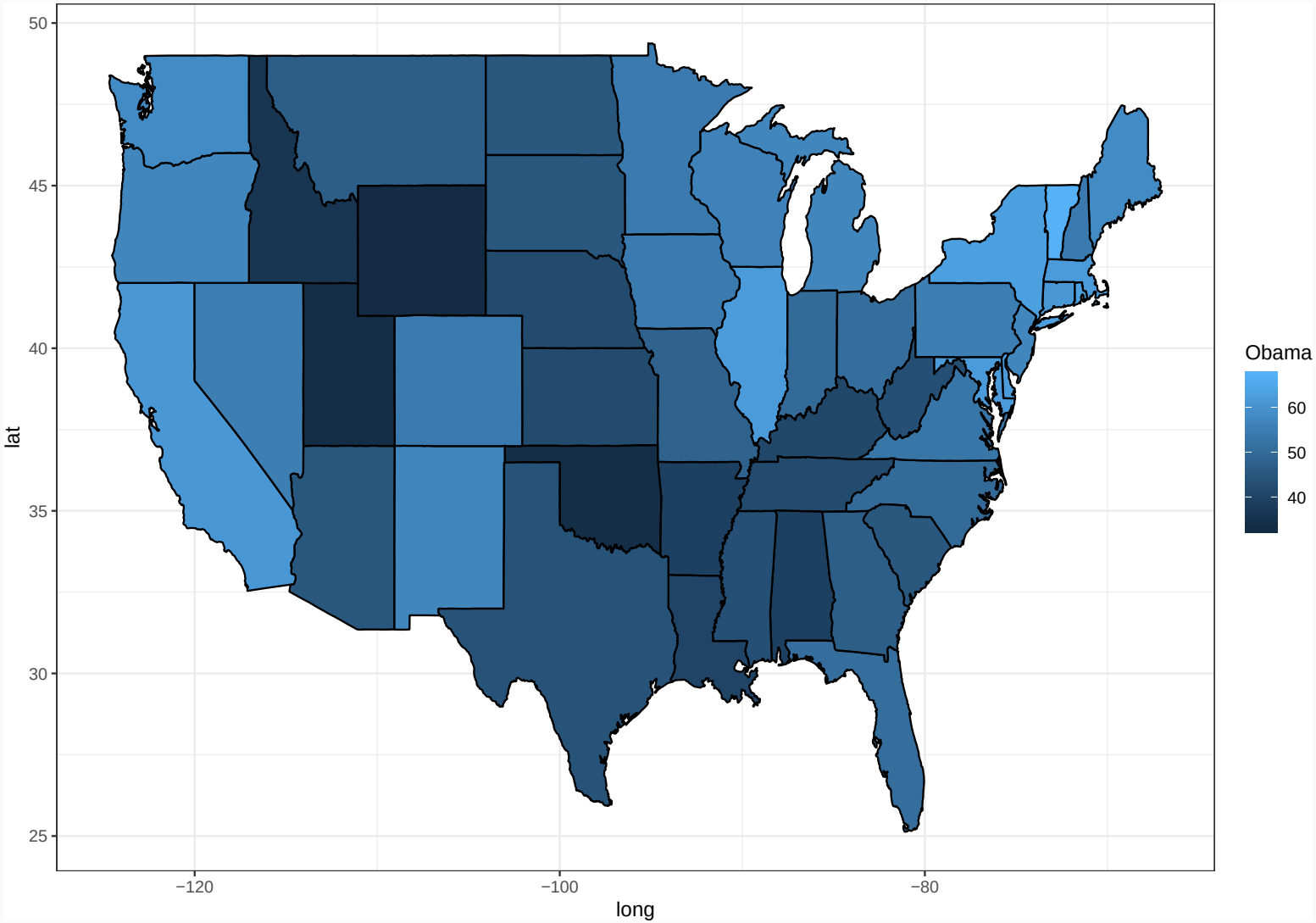
Joining

```
## rename and mutate joining variable
pres08 <- pres08 %>% mutate(region = tolower(state.name))

us_states <- us_states %>% left_join(pres08)

head(us_states)
```

```
##        long      lat group order  region subregion state.name state Obama
## 1 -87.46201 30.38968     1     1 alabama      <NA>    Alabama    AL    39
## 2 -87.48493 30.37249     1     2 alabama      <NA>    Alabama    AL    39
## 3 -87.52503 30.37249     1     3 alabama      <NA>    Alabama    AL    39
## 4 -87.53076 30.33239     1     4 alabama      <NA>    Alabama    AL    39
## 5 -87.57087 30.32665     1     5 alabama      <NA>    Alabama    AL    39
## 6 -87.58806 30.32665     1     6 alabama      <NA>    Alabama    AL    39
##   McCain EV
## 1     60  9
## 2     60  9
## 3     60  9
## 4     60  9
## 5     60  9
## 6     60  9
```

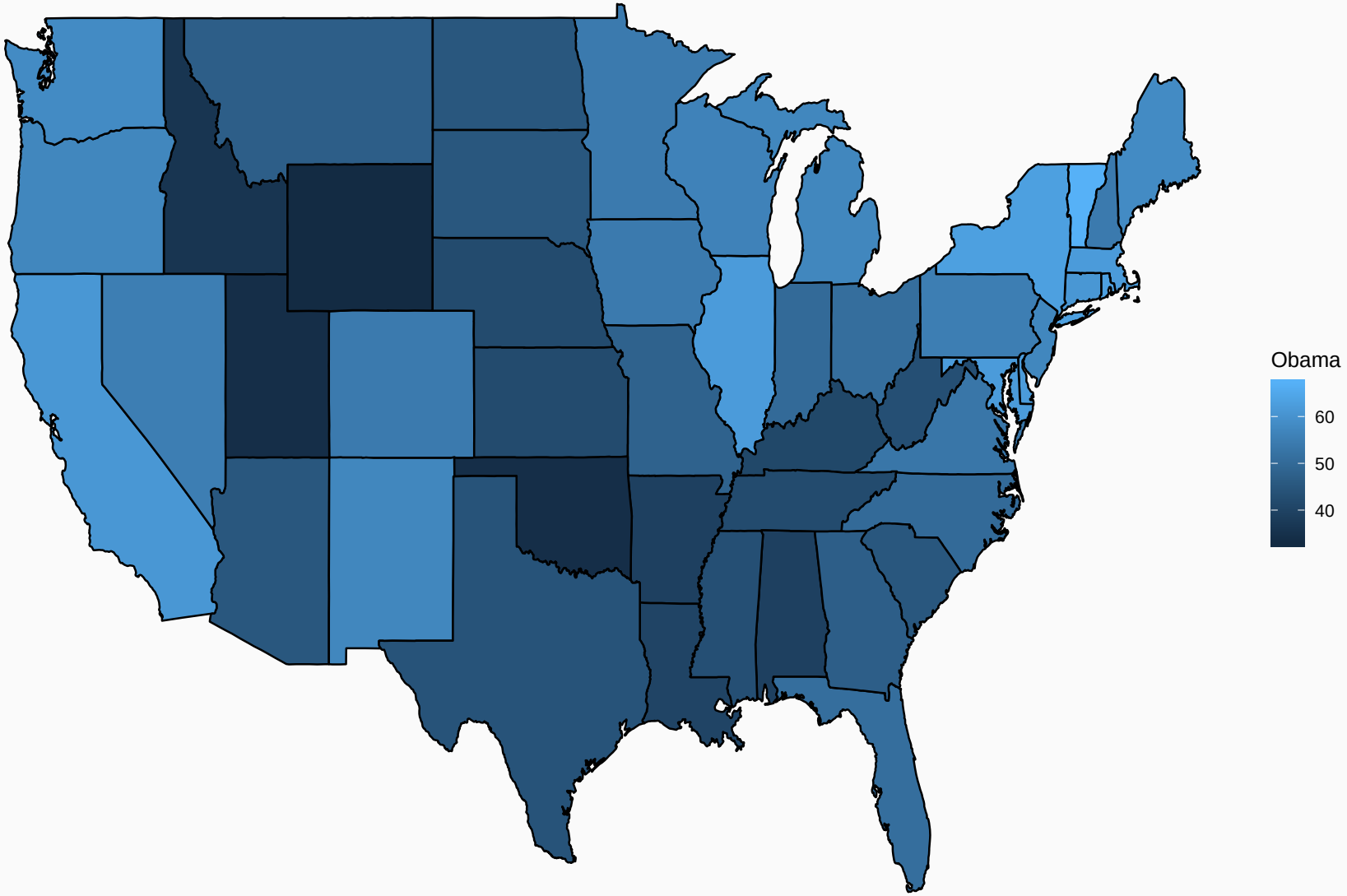
Make Obama vote share the fill color

```
ggplot(us_states, aes(x = long, y = lat, group = group, fill = Obama)) + geom_polygon(color = "black")
```



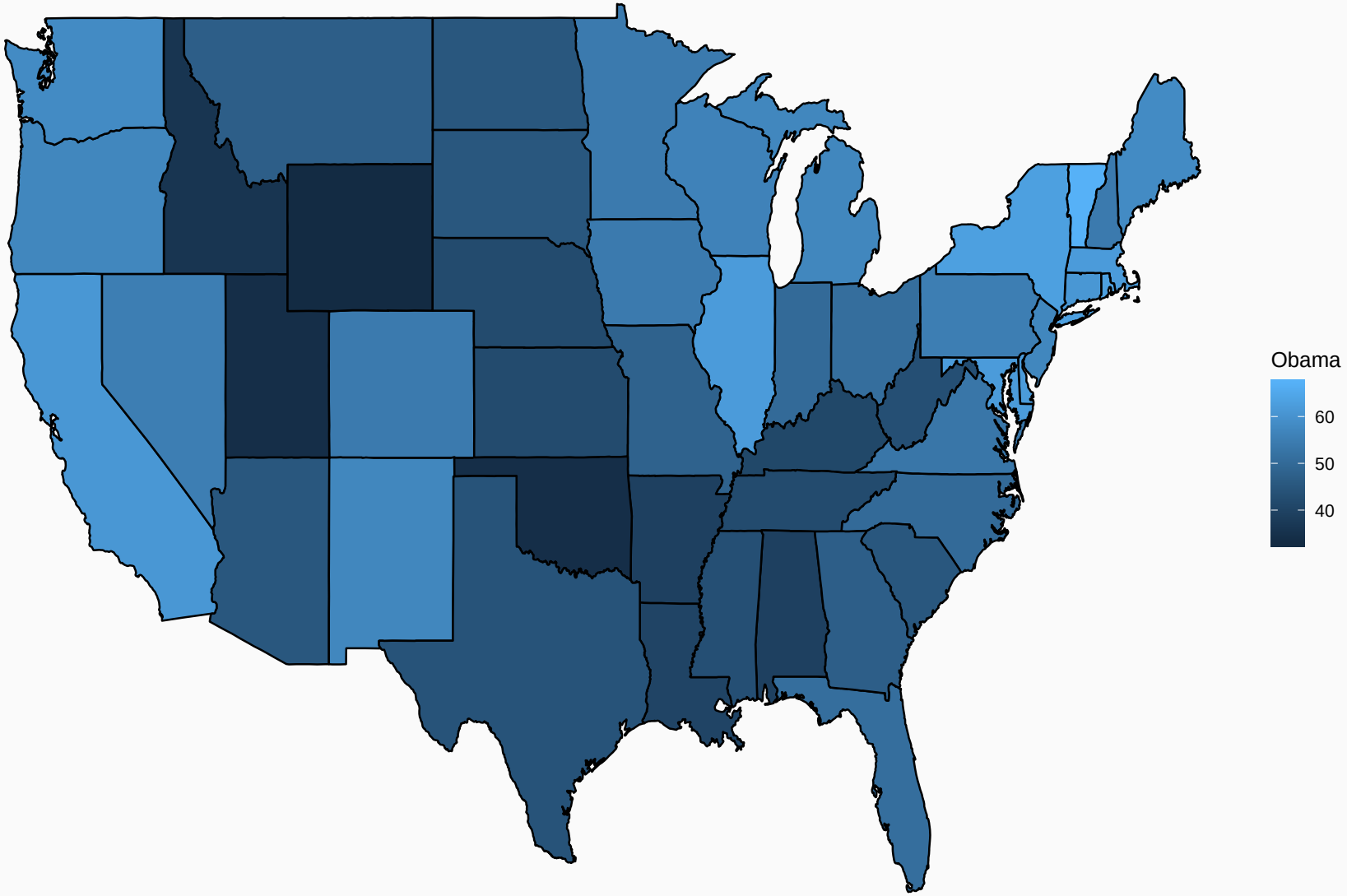
Make Obama vote share the fill color: remove junk

```
ggplot(us_states, aes(x = long, y = lat, group = group, fill = Obama)) + geom_polygon(color = "black") +
    theme_void()
```



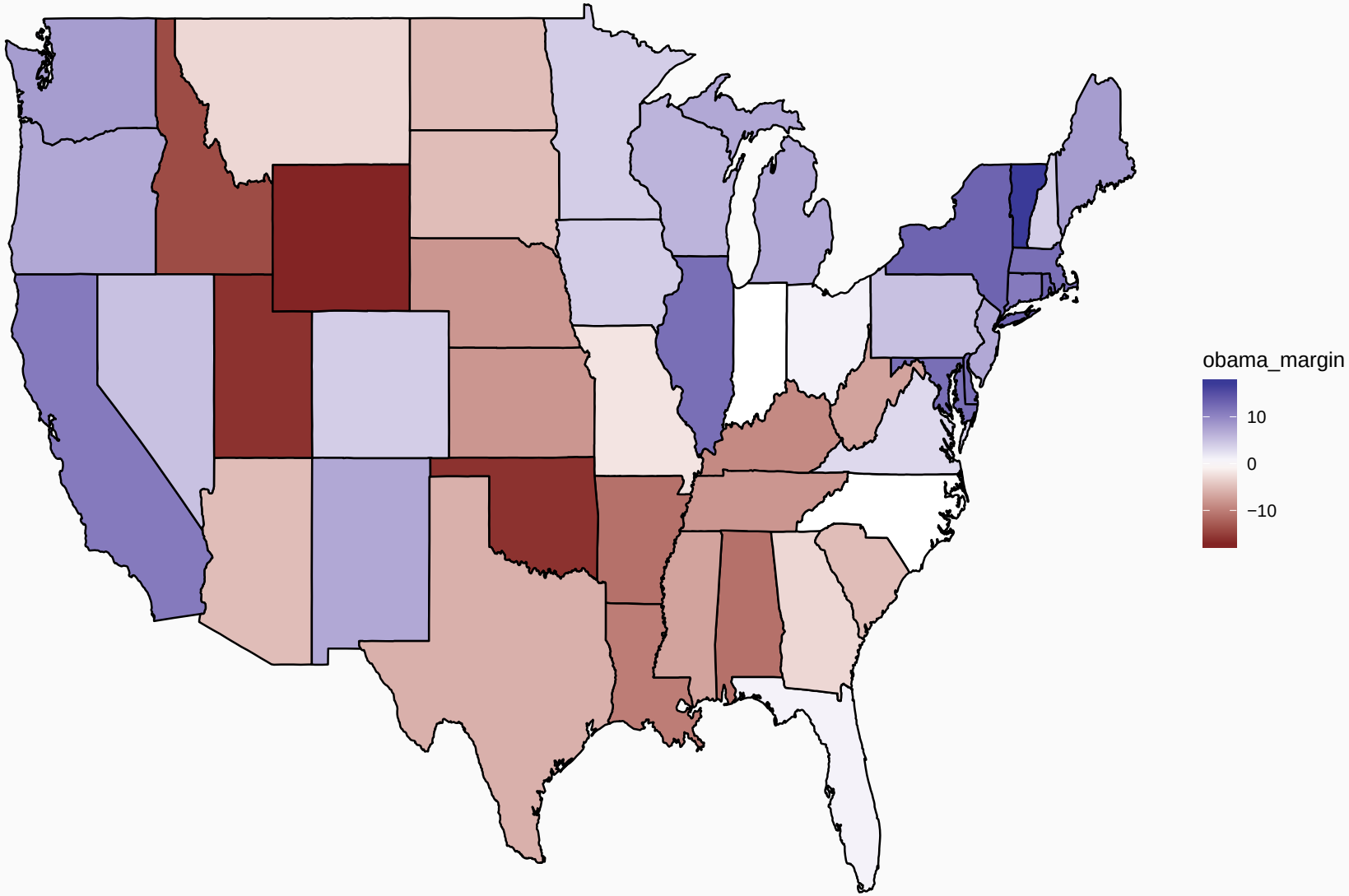
Make Obama vote share the fill color: remove junk

```
ggplot(us_states, aes(x = long, y = lat, group = group, fill = Obama)) + geom_polygon(color = "black") +
    theme_void()
```



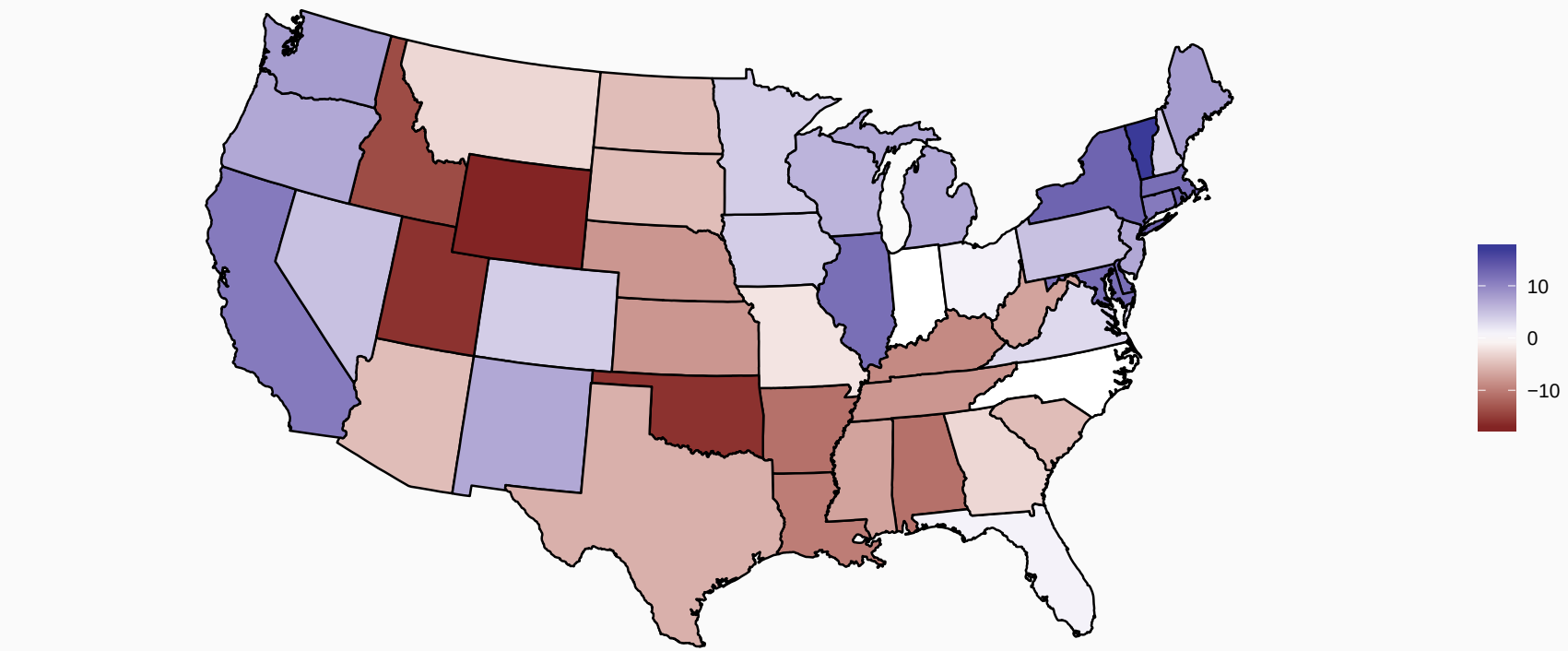
Vote share as diverging gradient

```
us_states$obama_margin <- us_states$Obama - 50
ggplot(us_states, aes(x = long, y = lat, group = group, fill = obama_margin)) +
    geom_polygon(color = "black") + scale_fill_gradient2() + theme_void()
```



Make it prettier - Albers projection, title and subtitle

```
library(mapproj)
ggplot(us_states, aes(x = long, y = lat, group = group, fill = obama_margin)) +
    geom_polygon(color = "black") + scale_fill_gradient2() + theme_void() +
    coord_map(projection = "albers", lat0 = 39, lat1 = 45) + labs(title = "US Election results 2008",
    subtitle = "Barack Obama margin of victory", fill = "")
```



Housekeeping

- No office hours on Friday this week, I'll have extra office hours on Monday 10-2PM
- Bring homework questions to lab today
- Question 5.5.3 you can choose to either animate or facet based on your preferences. I generally go with a facet
- When instructed to write a loop on the homework, you often don't need to. See Arnold for examples