Visualization Exercises

Ethan Marcano

2022-03-21

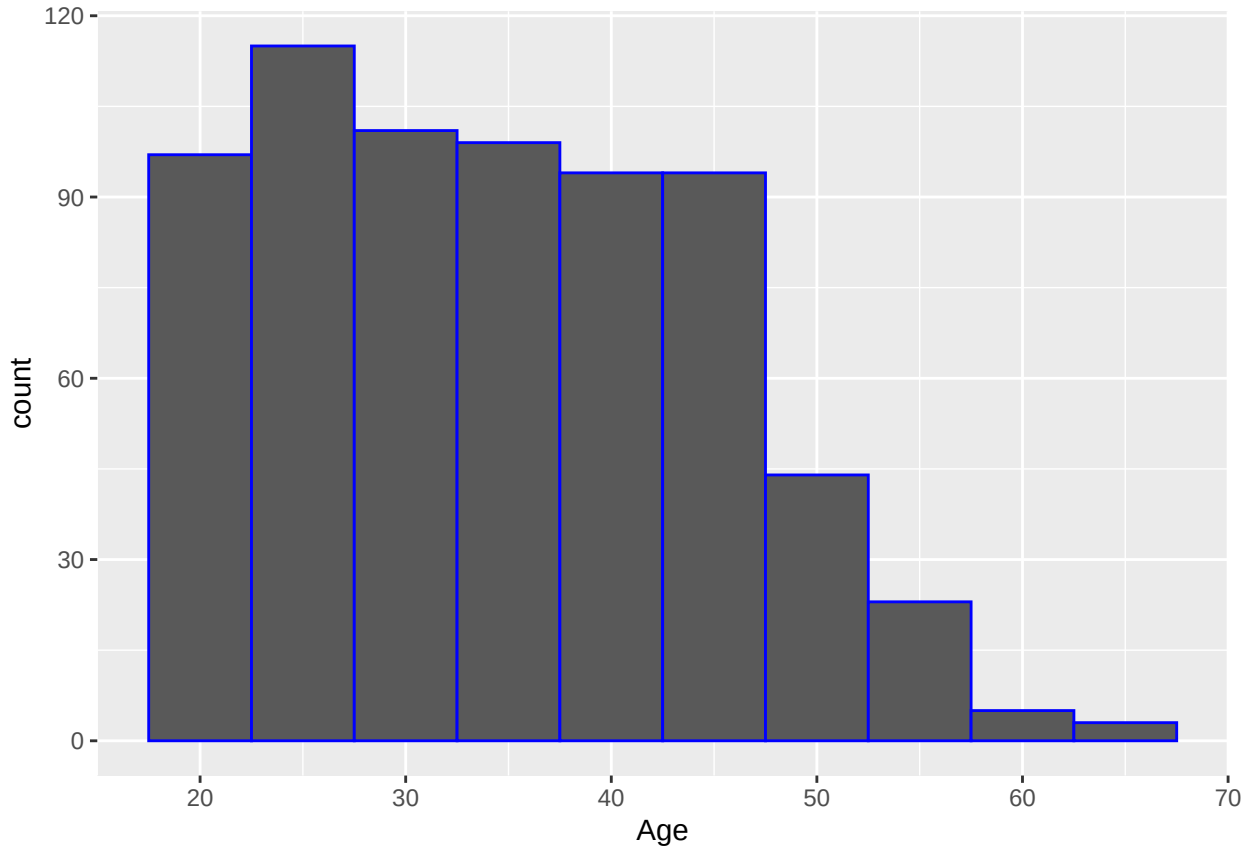
Visualization Exercises

Question 1: Visualization Attributes of Parole Violators

Part A)

```
Parole <- read_csv("Parole.csv")
```

```
ggplot(data = Parole, aes(x = Age)) + geom_histogram(binwidth = 5,
    color = "blue")
```



i)

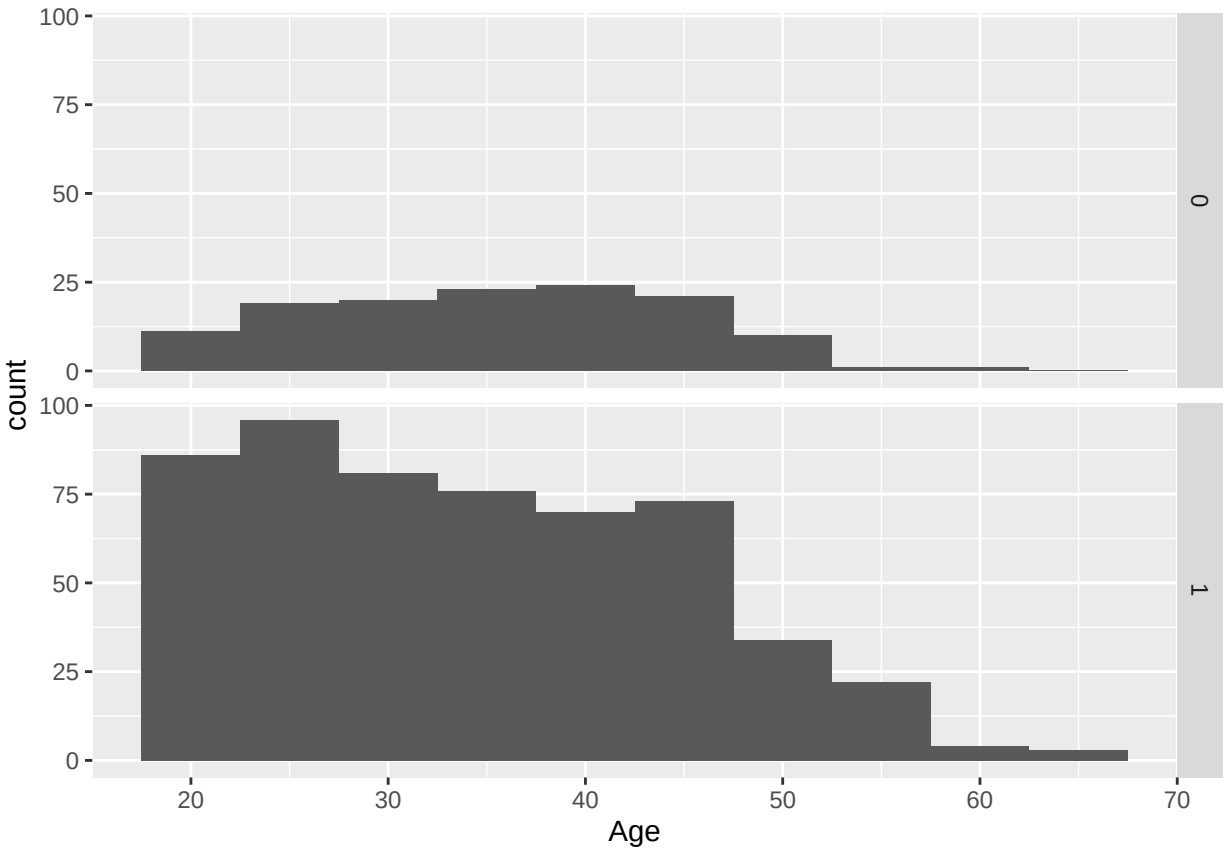
- The age bracket with the most parolees is the 25-30 age bracket. The one with the least parolees is the 65-69 age bracket.

ii)

- It creates a blue outline for each bracket, enabling the user to see the data more clearly.

Part B)

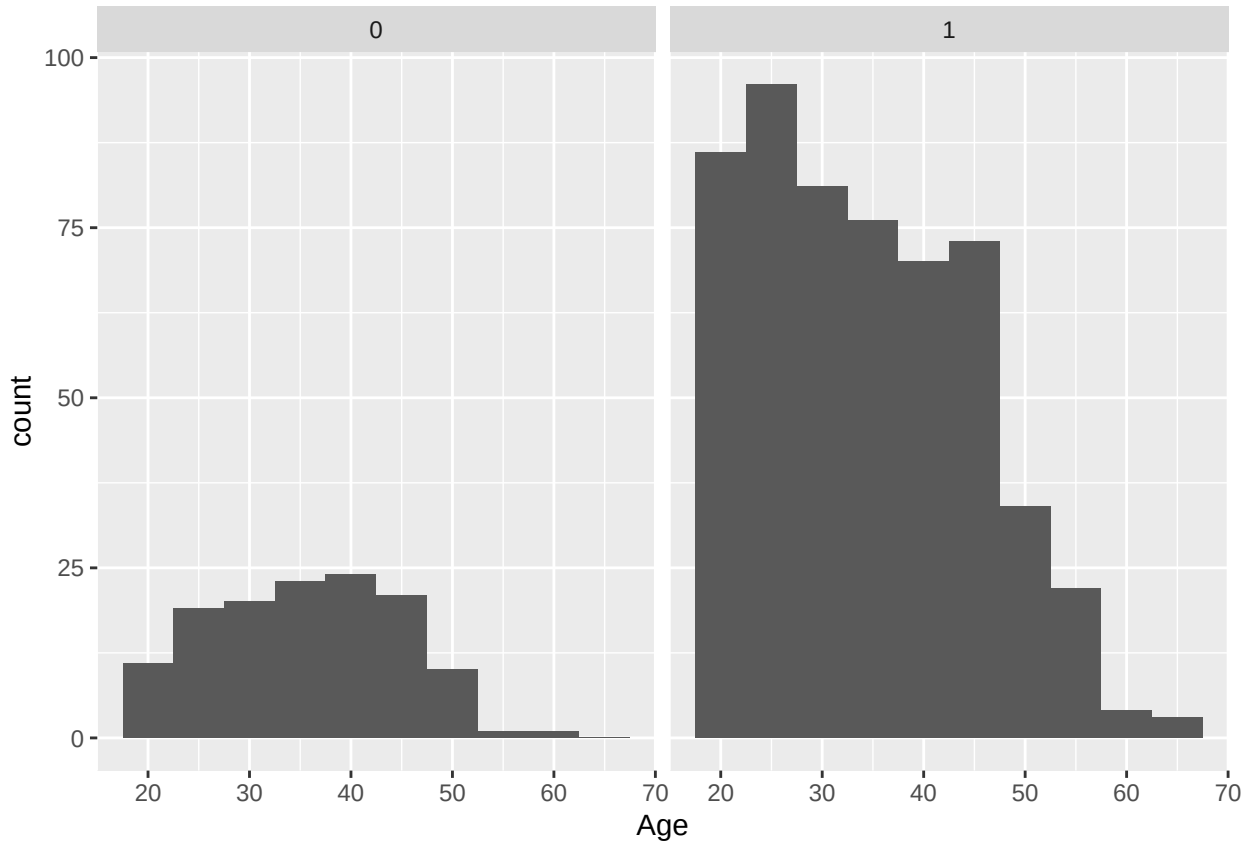
```
ggplot(data = Parole, aes(x = Age)) + geom_histogram(binwidth = 5) +
    facet_grid(Male ~ .)
```



i)

- The majority of prisoners in all age brackets are male. However the male prisoners are on the younger side, while the female prisoners trend towards middle age.

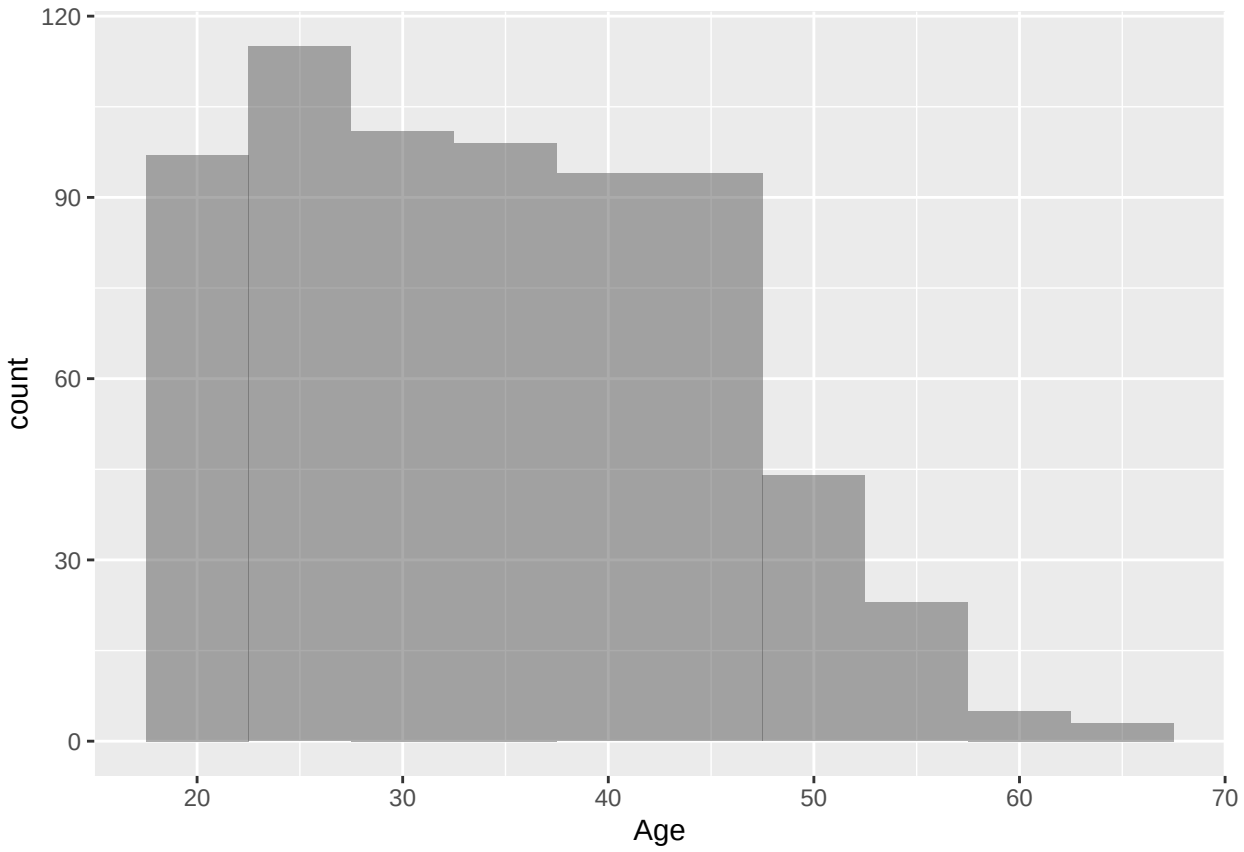
```
ggplot(data = Parole, aes(x = Age)) + geom_histogram(binwidth = 5) +
    facet_grid(. ~ Male)
```



ii)

- It flips the axis for the graph.

```
ggplot(data = Parole, aes(x = Age, fill = Male)) + geom_histogram(binwidth = 5,
    position = "identity", alpha = 0.5)
```

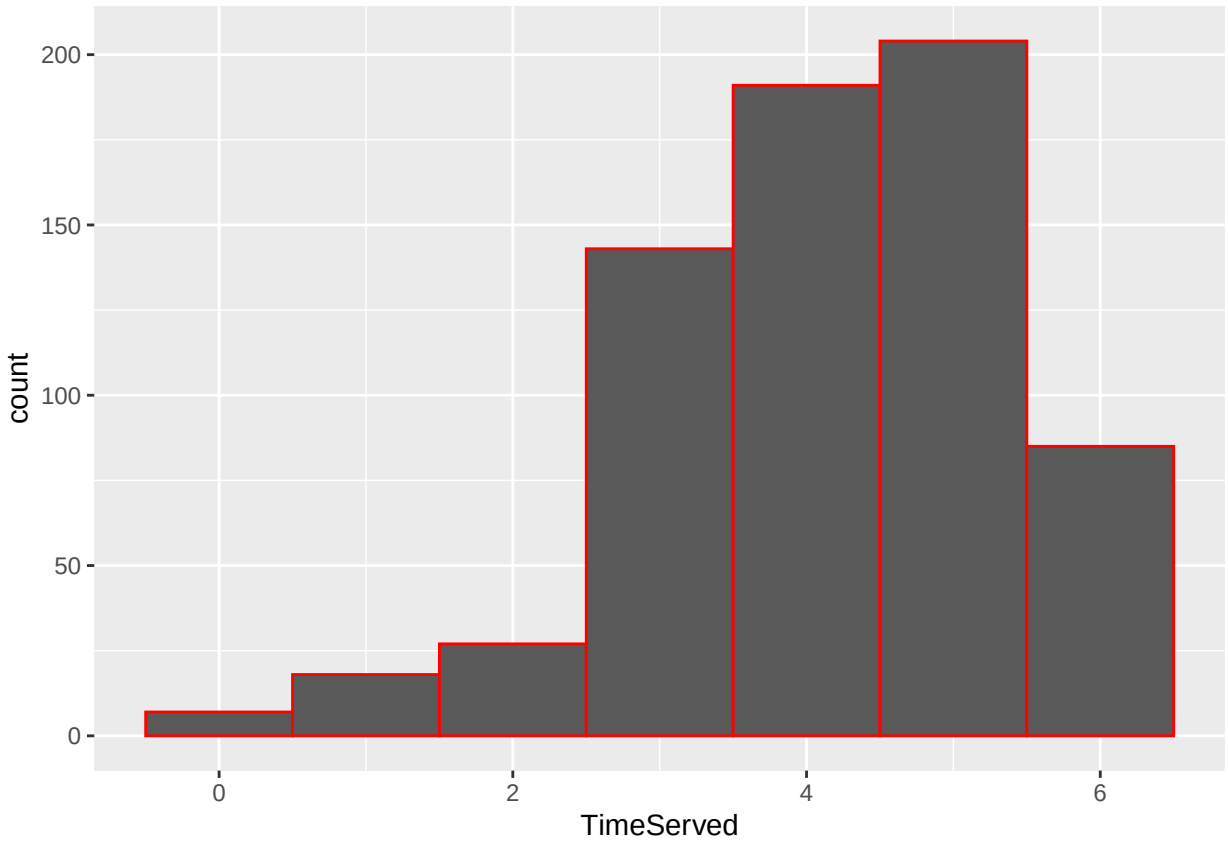


iii)

- The faceting histograms look better in delineating between different data groups, and enables easy comparison.

Part C)

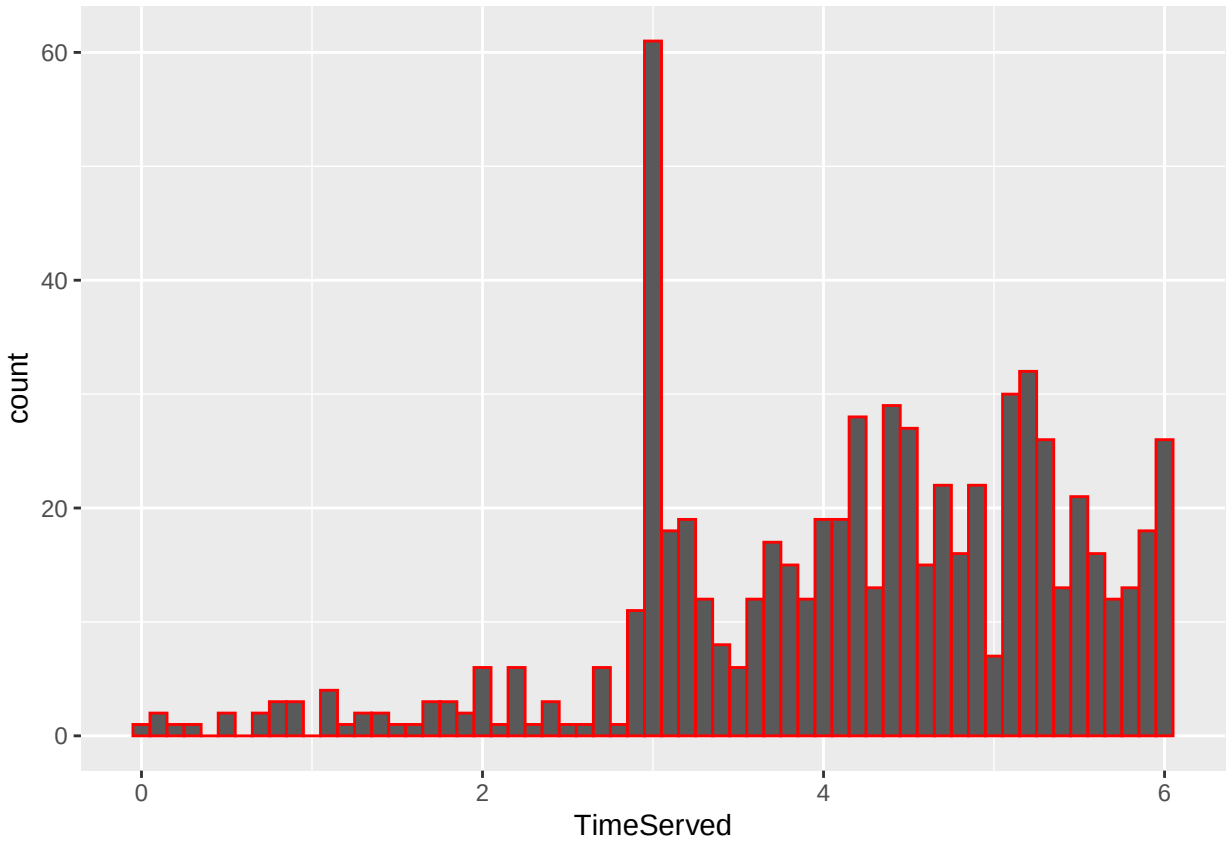
```
ggplot(data = Parole, aes(x = TimeServed)) + geom_histogram(binwidth = (1),
    color = "red")
```



i)

- The most common length of time served is 5 years.

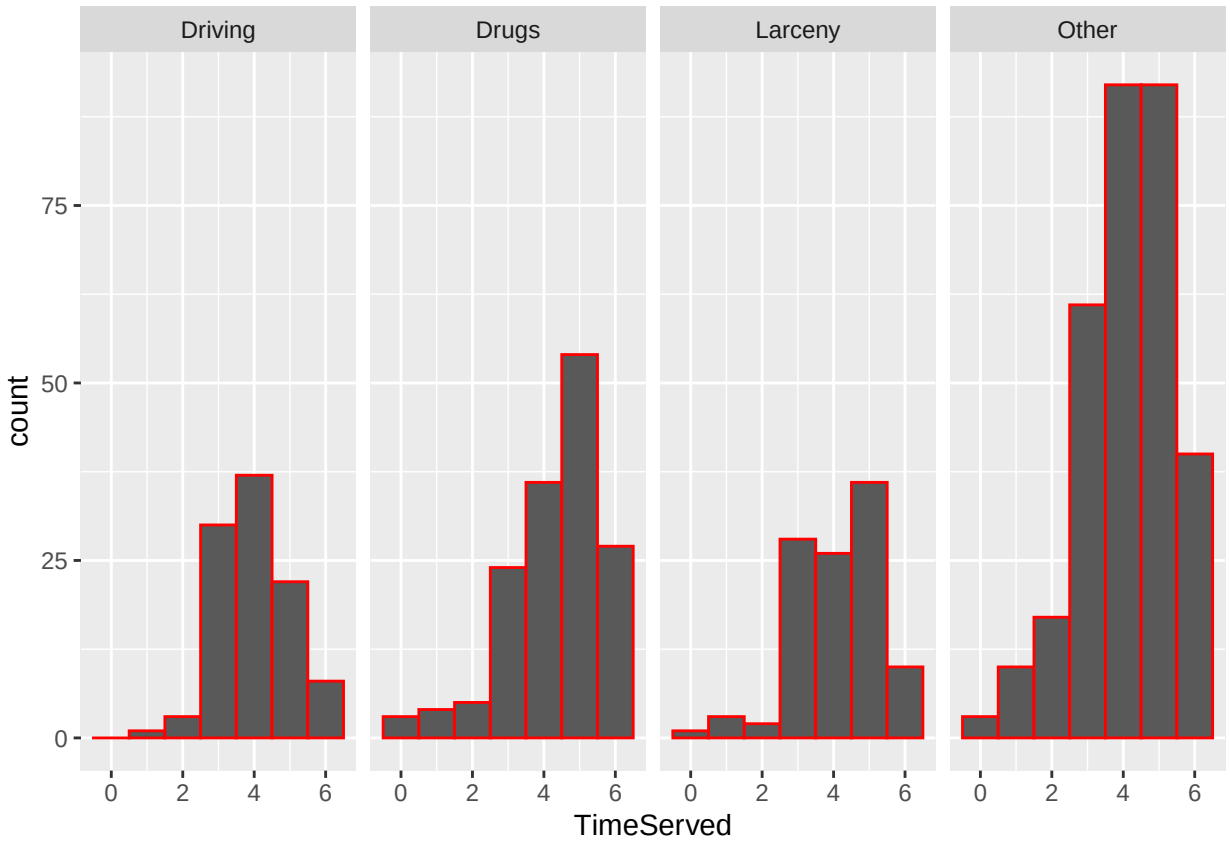
```
ggplot(data = Parole, aes(x = TimeServed)) + geom_histogram(binwidth = 0.1,
    color = "red")
```



ii)

- The most common length of time served is 3 years. It may be necessary to make visualizations more precise as to avoid interpretation errors.

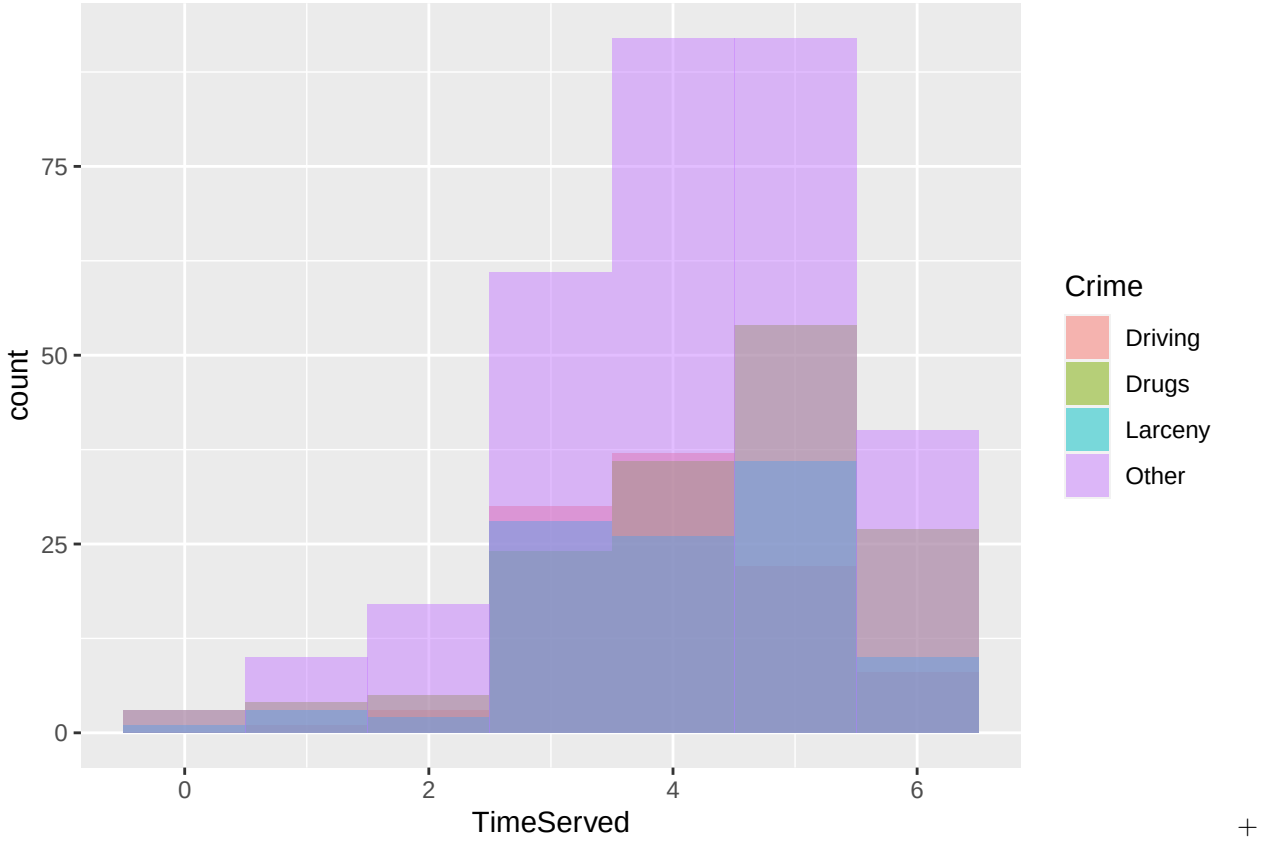
```
ggplot(data = Parole, aes(x = TimeServed)) + geom_histogram(binwidth = 1,
    color = "red") + facet_grid(. ~ Crime)
```



iii)

- Out of the named crimes, a drug arrest suggests a longer time served. Other crimes (may referring to more severe misdemeanors or felonies) also tend to carry a harder penalty.

```
ggplot(data = Parole, aes(x = TimeServed, fill = Crime)) + geom_histogram(binwidth = 1,
    position = "identity", alpha = 0.5)
```



The faceting histogram enables you to look at all types of data, and I prefer for much of the same reasons as the gender histograms.

Question 2: Visualizing Network Data

Part A)

```
users <- read_csv("Users.csv")
edges <- read_csv("Edges.csv")
```

```
nrow(users)
```

```
## [1] 59
```

i)

- There are 59 users in our dataset.

```
nrow(edges)/nrow(users)
```

```
## [1] 2.474576
```

ii)

- There is an average of about 2.5 friends between users in the network.

```
locale_count <- users %>%
    count(Locale)
locale_count
```

```
## # A tibble: 3 x 2
##   Locale     n
```

```
##   <chr>  <int>
## 1 A          6
## 2 B         50
## 3 <NA>       3
```

iii)

- Locale B is the most common, outnumbering A by 44 users.

```
gender_school <- users %>%
    count(School, Gender)
gender_school
```

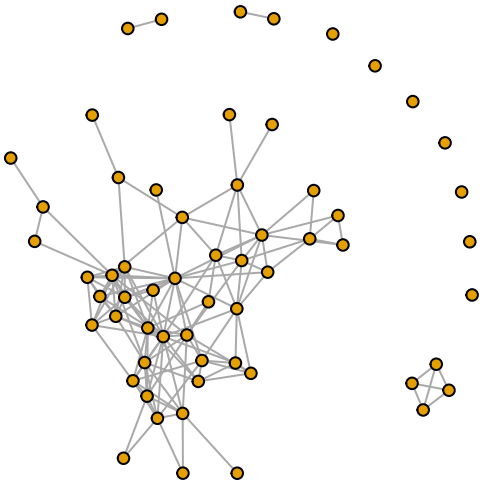
```
## # A tibble: 8 x 3
##   School Gender     n
##   <chr>  <chr>  <int>
## 1 A      A          3
## 2 A      B         13
## 3 A      <NA>       1
## 4 AB     A          1
## 5 AB     B          1
## 6 <NA>   A         11
## 7 <NA>   B         28
## 8 <NA>   <NA>       1
```

iv)

- Neither school is an all-girls or all-boys school.

Part B)

```
g <- graph.data.frame(edges, FALSE, users)
plot(g, vertex.size = 5, vertex.label = NA)
```



i)

- There are three connected components outside the main group.

ii)

- There are seven users with no friends.

```
degree(g)
```

```
## 3981 3982 3983 3984 3985 3986 3987 3988 3989 3990 3991 3992 3993 3994 3995  594
##    7   13    1    0    5    8    1    6    5    3    2    2    5   10    8    3
## 3996 3997 3998 3999 4000 4001 4002 4003 4004 4005 4006 4007 4008 4009 4010 4011
##    3   10   13    3    8    1    6    4    9    2    1    3    0    9    0    3
## 4012 4013 4014 4015 4016 4017 4018 4019 4020 4021 4022 4023 4024 4025 4026 4027
##    1    5   11    0    3    8    6    7    7   10    0   17    0    3    8    6
## 4028 4029 4030 4031 4032 4033 4034 4035 4036 4037 4038
##    1    1   18   10    1    2    1    0    1    3    8
```

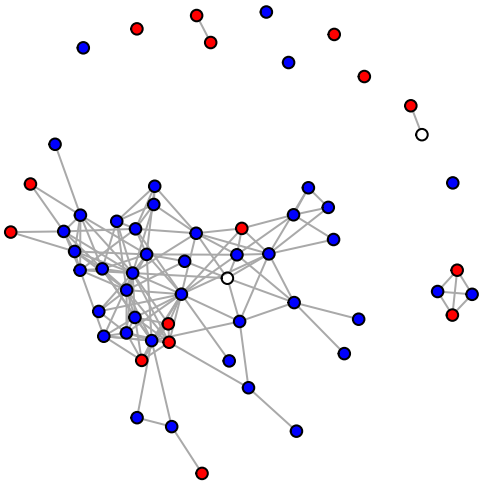
iii)

- There are nine users with 10 or more friends.

Part C)

```
V(g)$color[V(g)$Gender == "A"] = "red"
V(g)$color[V(g)$Gender == "B"] = "blue"

plot(g, vertex.size = 5, vertex.label = NA)
```

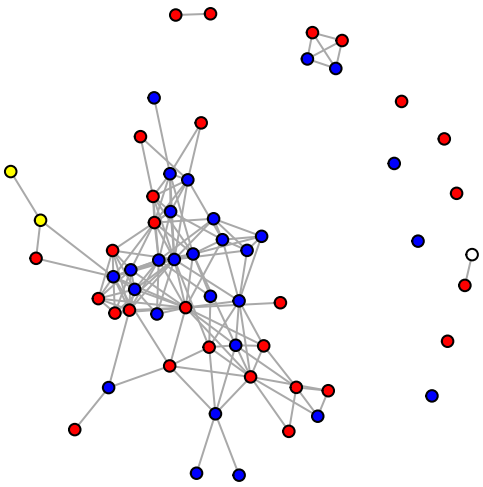


i)

- It seems that users of Gender B tend to have a high number of friends and make up a vast majority of connections.

```
V(g)$color[V(g)$School == "A"] = "red"
V(g)$color[V(g)$School == "B"] = "blue"
V(g)$color[V(g)$School == "AB"] = "yellow"

plot(g, vertex.size = 5, vertex.label = NA)
```



ii)

- Yes, those users are both friends with each other.

Question 3: Election Forecasting

Part A)

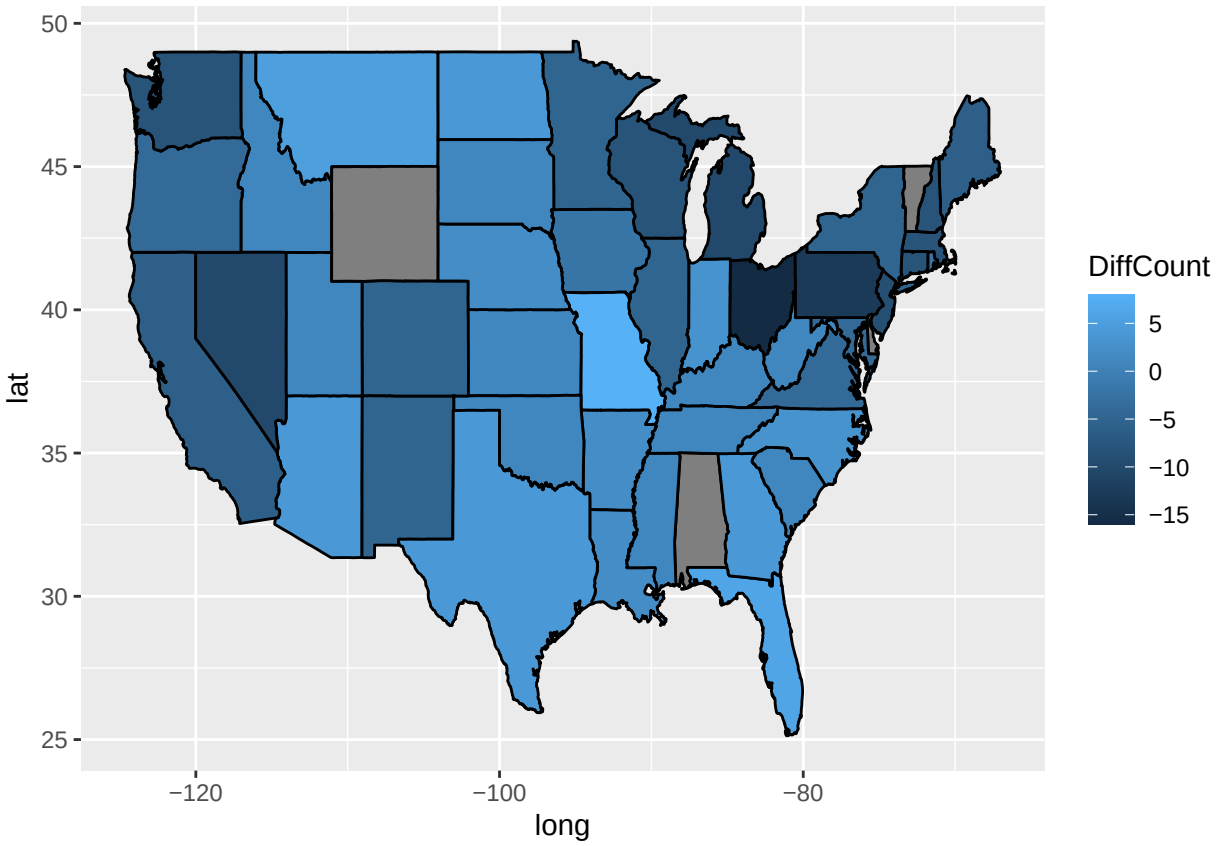
```
polling <- read_csv("Polling.csv")
```

```
## Rows: 50 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): State
## dbl (5): Rasmussen, SurveyUSA, DiffCount, PropR, Republican
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

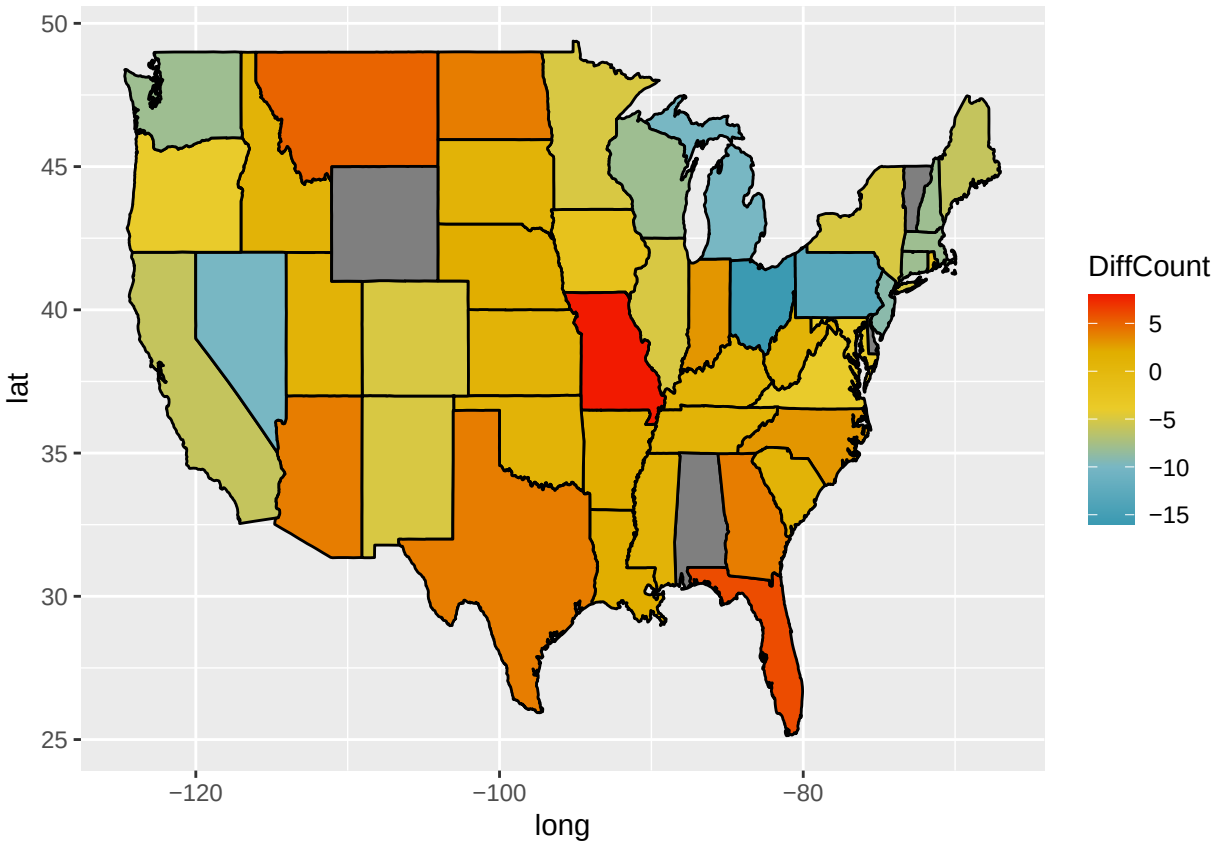
```
pal <- wes_palette("Zissou1", 100, type = "continuous")
```

```
statesMap <- map_data("state")
electionMap <- merge(statesMap, polling, by.x = "region", by.y = "State")
electionMap <- electionMap[order(electionMap$order), ]
```

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



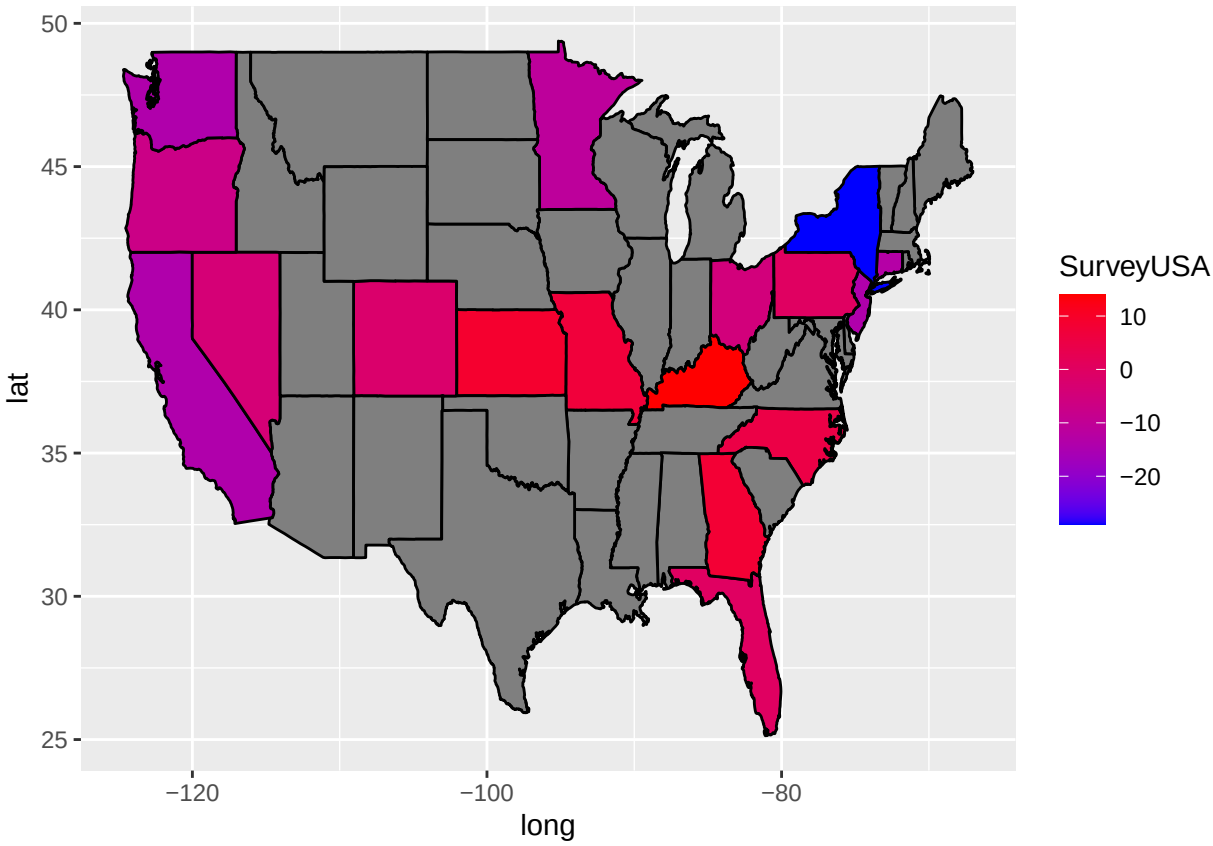

```
ggplot(electionMap, aes(x = long, y = lat, group = group, fill = DiffCount)) +
    geom_polygon(color = "black") + scale_fill_gradientn(colours = pal)
```



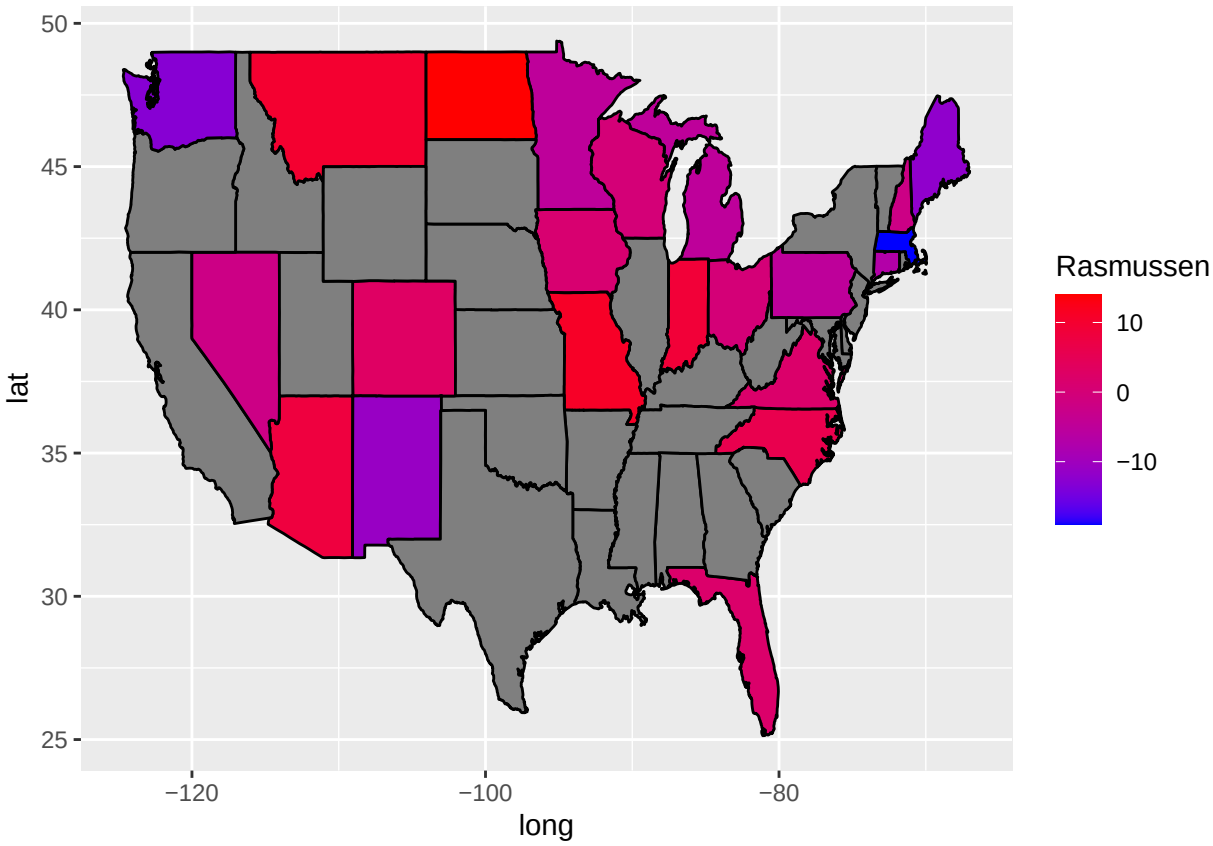
- Missouri, Montana, and Florida stick out as consistent polling states. Ohio is a notable outlier.

Part B)

```
ggplot(electionMap, aes(x = long, y = lat, group = group, fill = SurveyUSA)) +
    geom_polygon(color = "black") + scale_fill_gradient(low = "blue",
    high = "red")
```




```
ggplot(electionMap, aes(x = long, y = lat, group = group, fill = Rasmussen)) +
    geom_polygon(color = "black") + scale_fill_gradient(low = "blue",
    high = "red")
```



i)

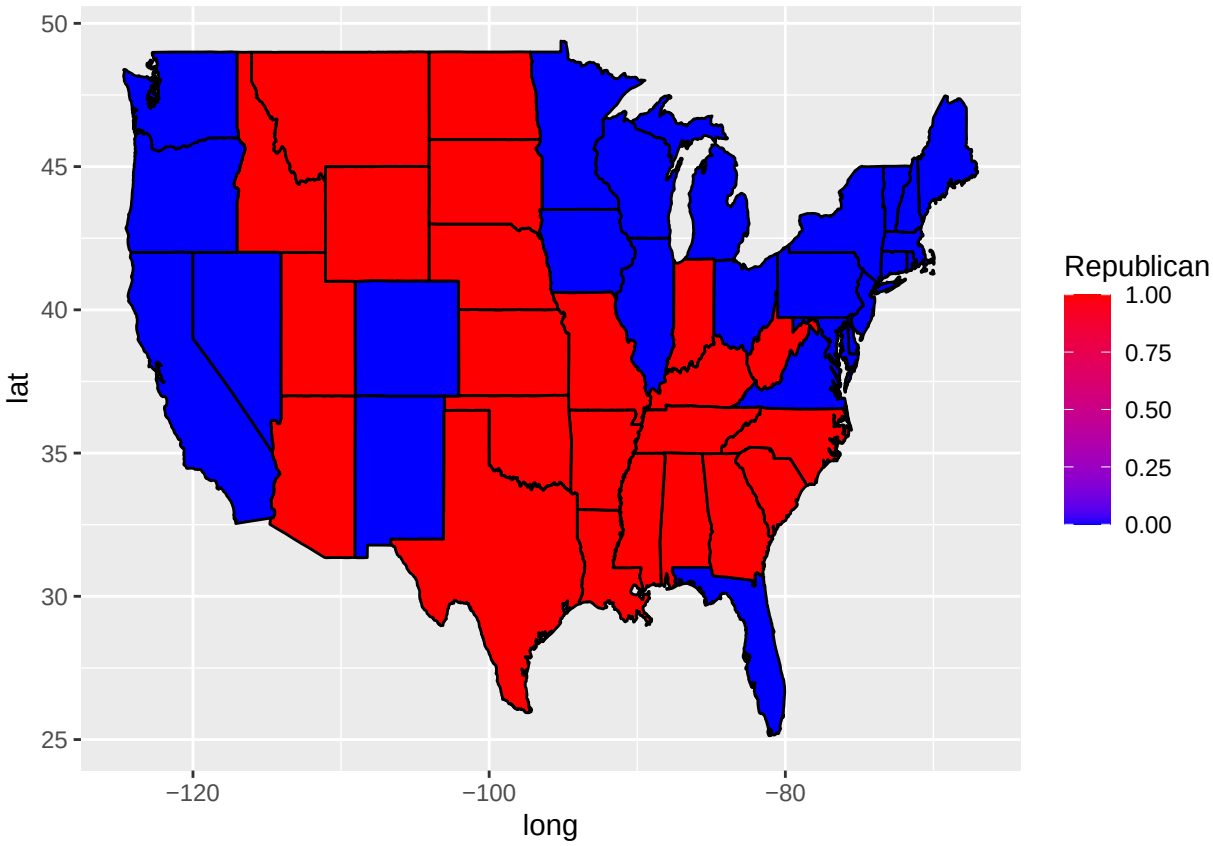
- The most notable thing that while both groups share some common states between them, SurveyUSA focuses on primarily on the Mid-Atlantic region while Rasmussen covers the Midwest. Rasmussen also trends centrist, while SurveyUSA trends Republican. Compared to the DiffCount map, both groups predict a redder Midwest than DiffCount.

ii)

- All three maps predict a tight race, leaning Republican. For the 2012 election, none of these maps are correct.

Part C)

```
ggplot(electionMap, aes(x = long, y = lat, group = group, fill = Republican)) +
    geom_polygon(color = "black") + scale_fill_gradient(low = "blue",
    high = "red")
```



iii)

- All of the maps got a purple Midwest relatively right, as well as a blue West Coast and Northeast.
- If I were to create a predictive model, I would stick to aggregating different polling sources, particularly in swing state regions such as the Midwest and Mid-Atlantic.