Final Exam

DATA 101

Due October 25th at 11:59pm

This exam consists of 16 questions. You must hand in an .rmd file as well as a knitted pdf or word document. The rmd of this exact file is available, and you are welcome to use that as a template for your answers.

You are not allowed to share code with classmates. You may ask clarifying questions to TAs/me. If you are stuck on something and can't continue to the next part of the assignment, you can ask a TA or me to give you the code to continue, but do expect to lose a couple of points. Make sure to take the time to add titles, labels, etc to make your graphs look professional.

For the final exam we are going to be working with elections returns from US Senate elections.

Comprehensive results for Senate races have been compiled by the MIT Election Data and Science lab. We're going to do some cleaning to get this into a format that we can use to analyze using a map. The changes we make in the data are going to be cumulative, so you should assume that changes you make to the data in one question (filtering, selecting variables etc) apply to all subsequent questions. For example, in Q6 you will remove Independent candidates from the data. All subsequent questions make use of this filtered data with no Independents.

1. **First, import this data using this link. Download the spreadsheet as a csv and load the election results into R.**

```
setwd("~/Documents/Upenn/Data 101/Final")
Senate <- read.csv("1976-2020-senate.csv")
head(Senate)
```

```
##   year      state state_po state_fips state_cen state_ic    office  district
## 1 1976    ARIZONA       AZ          4        86       61 US SENATE statewide
## 2 1976    ARIZONA       AZ          4        86       61 US SENATE statewide
## 3 1976    ARIZONA       AZ          4        86       61 US SENATE statewide
## 4 1976    ARIZONA       AZ          4        86       61 US SENATE statewide
## 5 1976    ARIZONA       AZ          4        86       61 US SENATE statewide
## 6 1976 CALIFORNIA       CA          6        93       71 US SENATE statewide
##   stage special           candidate       party_detailed writein  mode
## 1   gen   FALSE         SAM STEIGER           REPUBLICAN   FALSE total
## 2   gen   FALSE WM. MATHEWS FEIGHAN          INDEPENDENT   FALSE total
## 3   gen   FALSE    DENNIS DECONCINI             DEMOCRAT   FALSE total
## 4   gen   FALSE       ALLAN NORWITZ          LIBERTARIAN   FALSE total
## 5   gen   FALSE           BOB FIELD          INDEPENDENT   FALSE total
## 6   gen   FALSE          JACK MCCOY AMERICAN INDEPENDENT   FALSE total
##   candidatevotes totalvotes unofficial  version party_simplified
## 1         321236     741210      FALSE 20210114       REPUBLICAN
## 2           1565     741210      FALSE 20210114            OTHER
## 3         400334     741210      FALSE 20210114         DEMOCRAT
```

```
## 4           7310     741210      FALSE 20210114      LIBERTARIAN
## 5          10765     741210      FALSE 20210114            OTHER
## 6          82739    7470586      FALSE 20210114            OTHER
```

2. **Our first step to clean this data is removing non-substantive columns. Keep only the variables `year`, `state`, `state_po`, `stage`, `candidate`, `party_detailed`, `candidatevotes`, `totalvotes`.**

3. **Next, we're going to remove rows with some incomplete data. Remove any rows that have missing data in the "candidate" or "party" columns.**

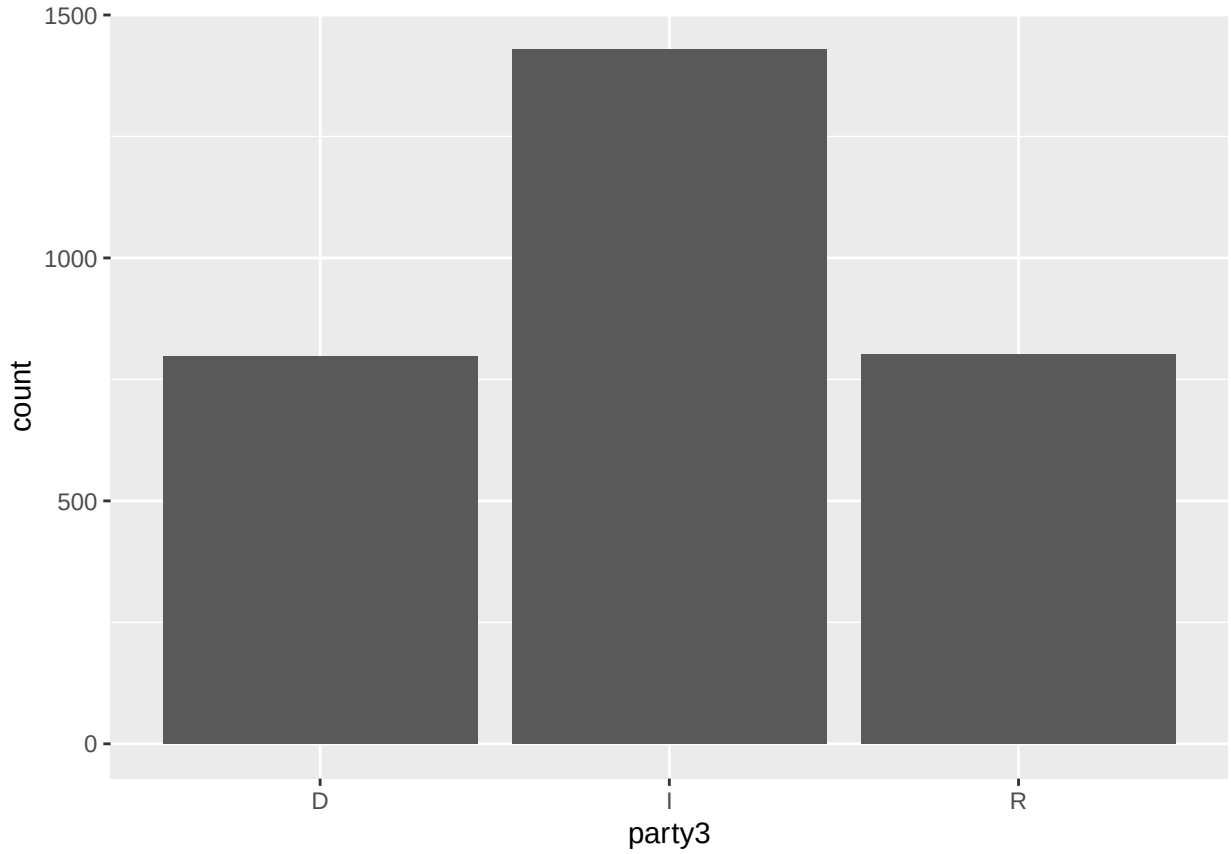
```
### Question 2 and 3
Senateclean <- Senate %>%
  select(year, state, state_po, stage, candidate, party_detailed,
         candidatevotes, totalvotes) %>%
  rename(party = "party_detailed") %>%
  na_if("") %>%
  drop_na(candidate, party)
```

4. **Next, create a new variable, "party3" which recodes the "party" column into "D" for Democrats, "R" for Republicans, and "I" for all other parties. (Hint: you may want to first create this column so that all rows equal "I". Then use the ifelse() function to recode to "R" if the row represents a Republican and otherwise stays equal to its current value.)**

```
Senateclean2 <- Senateclean %>%
  mutate(party3 = "I") %>%
  mutate(party3 = ifelse(grepl("^REPUBLICAN$",party), "R","I")) %>%
  mutate(party3 = ifelse(grepl("^DEMOCRAT$",party), "D",party3))
```

5. **How many Democrats are in this dataset? How many Republicans? How many Independents?**

```
ggplot(Senateclean2, mapping=aes(x=party3, posistion= "dodge")) +
         geom_bar()
```




```
Senateclean2 %>%
  group_by(party3) %>%
  summarise(n=n())
```

```
## # A tibble: 3 x 2
##   party3     n
##   <chr>  <int>
## 1 D        798
## 2 I       1430
## 3 R        802
```

6. **Now let's look at the 2-party vote in these data. First, remove the independent candidates from the data. Next, remove all the rows where "stage" is not equal to "gen". This ensures that we only get results from the general election.**

```
Question6 <- Senateclean2 %>%
  filter(party3 != "I") %>%
  filter(stage == "gen")
```

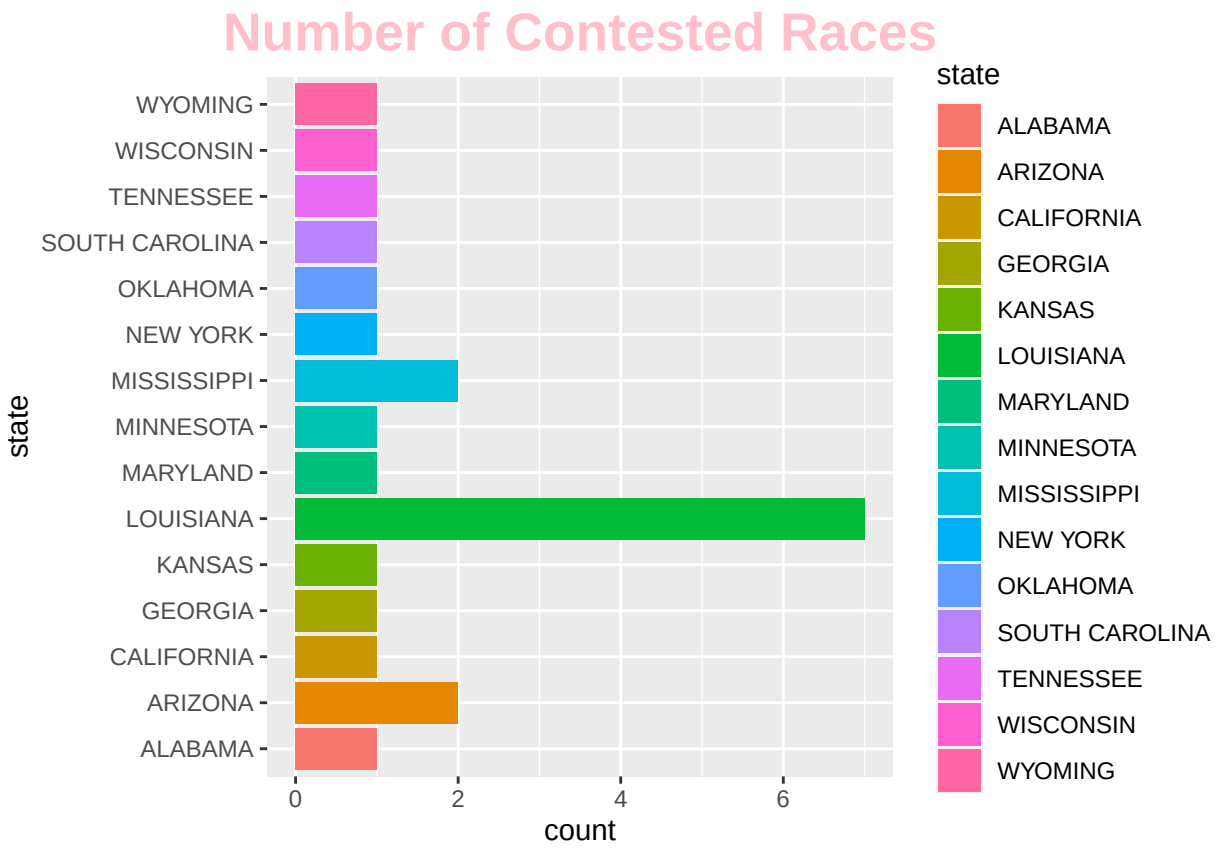
7. **How many races were contested between more than two candidates? Which state had the most of these races?**

```
Question7 <- Question6 %>%
  group_by(state,year) %>%
  summarise(party3 = n ())
```

```
## `summarise()` has grouped output by 'state'. You can override using the
## `.groups` argument.
```

```
Question7.a <- Question7 %>%
  mutate(contested = ifelse(party3>2, 1, 0)) %>%
  filter(contested == 1)

ggplot(Question7.a, mapping=aes(x=state, posistion = "dodge", fill=state)) +
  geom_bar() +
  coord_flip() +
  theme(plot.title = element_text(size = 20, face = "bold", color = "pink",
                                  hjust = 0.5))+
  labs(title="Number of Contested Races")
```




```
## Louisiana had the most contested races. In total there were 22 contested races.
```

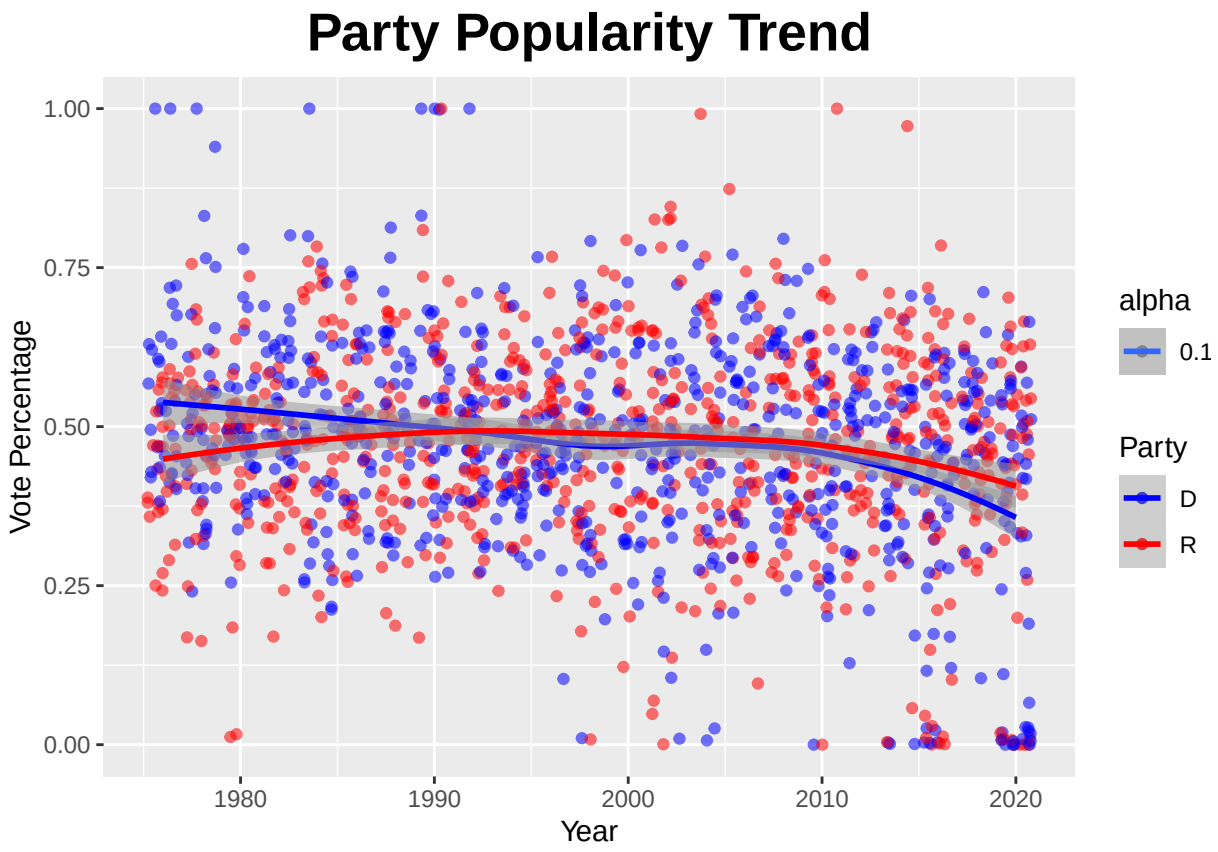
8. **For Democratic and Republican candidates create a figure that displays year on the x-axis and each candidate's percent of the vote on the y-axis. Be sure to color code each candidate by their respective party. Add two lines – one for each party – that represents the trend in that parties' support overtime.**

```
Question8 <- Question6 %>%
  mutate(perc_vote = (candidatevotes/totalvotes))

ggplot(Question8, mapping=aes(x=year,y=perc_vote, group=party3,color=party3, alpha=.1)) +
  scale_color_manual(values =c("blue", "red")) +
```

```
  theme(plot.title = element_text(size = 20, face = "bold", color = "black",
                                  hjust = 0.5)) +
  labs(title="Party Popularity Trend", y ="Vote Percentage", x = "Year", color = "Party") +
  geom_jitter() +
  geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



9. **Let's take a look at the races from 2012. Filter your dataset so that it only contains the results for 2012, and only the columns `year`, `state`, `party3`, and the candidate percent you calculated in the previous question. Reshape this data so that there is only one row per state, and two columns that represent the percent of the vote won by the Republican candidate and the percent of the vote won by the Democratic candidate. Note that you will not have 50 rows because not all states have a Senate election in an election year.**

```
Question9 <- Question8 %>%
  filter(year == 2012) %>%
  select(year, state, party3, perc_vote) %>%
  spread(key = party3, value = perc_vote)
```

10. **Create a variable "demwin" that records if the Democrat received a higher vote share than the Republican in each race in 2012.**

```
Question10 <- Question9 %>%
  mutate(demwin = ifelse(D > R, 1, 0))
Question10
```

```
##    year         state         D         R demwin
## 1  2012       ARIZONA 0.4620361 0.4923091      0
## 2  2012    CALIFORNIA 0.6252428 0.3747572      1
## 3  2012   CONNECTICUT 0.5245415 0.3999096      1
## 4  2012      DELAWARE 0.6641917 0.2895352      1
## 5  2012       FLORIDA 0.5523176 0.4222576      1
## 6  2012        HAWAII 0.6164553 0.3682733      1
## 7  2012       INDIANA 0.5004414 0.4428031      1
## 8  2012         MAINE 0.1281874 0.2972169      0
## 9  2012      MARYLAND 0.5597786 0.2632850      1
## 10 2012 MASSACHUSETTS 0.5327392 0.4579015      1
## 11 2012      MICHIGAN 0.5879807 0.3798446      1
## 12 2012     MINNESOTA 0.6522898 0.3052799      1
## 13 2012   MISSISSIPPI 0.4055090 0.5715563      0
## 14 2012      MISSOURI 0.5481432 0.3911372      1
## 15 2012       MONTANA 0.4857838 0.4486037      1
## 16 2012      NEBRASKA 0.4222557 0.5777443      0
## 17 2012        NEVADA 0.4470613 0.4586628      0
## 18 2012    NEW JERSEY 0.5886546 0.3937436      1
## 19 2012    NEW MEXICO 0.5100807 0.4527754      1
## 20 2012      NEW YORK 0.6210867 0.2128321      1
## 21 2012  NORTH DAKOTA 0.5023821 0.4932398      1
## 22 2012          OHIO 0.5070070 0.4470002      1
## 23 2012  PENNSYLVANIA 0.5369002 0.4458759      1
## 24 2012  RHODE ISLAND 0.6481137 0.3496553      1
## 25 2012     TENNESSEE 0.3040659 0.6489158      0
## 26 2012         TEXAS 0.4062300 0.5645566      0
## 27 2012          UTAH 0.2998041 0.6531010      0
## 28 2012       VERMONT        NA 0.2490009     NA
## 29 2012      VIRGINIA 0.5286595 0.4696081      1
## 30 2012    WASHINGTON 0.6045099 0.3954901      1
## 31 2012 WEST VIRGINIA 0.6057207 0.3647172      1
## 32 2012     WISCONSIN 0.5140886 0.4586034      1
## 33 2012       WYOMING 0.2114838 0.7389310      0
```

11. **Create a variable "demdiff" that records the difference between the Democratic and Republican share of the vote in each race in 2012.**

```
Question11 <- Question10 %>%
  mutate(demdiff = (D - R))
Question11
```

```
##    year         state         D         R demwin      demdiff
## 1  2012       ARIZONA 0.4620361 0.4923091      0 -0.030272949
## 2  2012    CALIFORNIA 0.6252428 0.3747572      1  0.250485689
## 3  2012   CONNECTICUT 0.5245415 0.3999096      1  0.124631887
## 4  2012      DELAWARE 0.6641917 0.2895352      1  0.374656537
## 5  2012       FLORIDA 0.5523176 0.4222576      1  0.130059954
```

```
## 6  2012        HAWAII 0.6164553 0.3682733      1  0.248182012
## 7  2012       INDIANA 0.5004414 0.4428031      1  0.057638328
## 8  2012         MAINE 0.1281874 0.2972169      0 -0.169029418
## 9  2012      MARYLAND 0.5597786 0.2632850      1  0.296493589
## 10 2012 MASSACHUSETTS 0.5327392 0.4579015      1  0.074837730
## 11 2012      MICHIGAN 0.5879807 0.3798446      1  0.208136056
## 12 2012     MINNESOTA 0.6522898 0.3052799      1  0.347009908
## 13 2012   MISSISSIPPI 0.4055090 0.5715563      0 -0.166047289
## 14 2012      MISSOURI 0.5481432 0.3911372      1  0.157006053
## 15 2012       MONTANA 0.4857838 0.4486037      1  0.037180136
## 16 2012      NEBRASKA 0.4222557 0.5777443      0 -0.155488655
## 17 2012        NEVADA 0.4470613 0.4586628      0 -0.011601465
## 18 2012    NEW JERSEY 0.5886546 0.3937436      1  0.194910990
## 19 2012    NEW MEXICO 0.5100807 0.4527754      1  0.057305235
## 20 2012      NEW YORK 0.6210867 0.2128321      1  0.408254583
## 21 2012  NORTH DAKOTA 0.5023821 0.4932398      1  0.009142316
## 22 2012          OHIO 0.5070070 0.4470002      1  0.060006776
## 23 2012  PENNSYLVANIA 0.5369002 0.4458759      1  0.091024274
## 24 2012  RHODE ISLAND 0.6481137 0.3496553      1  0.298458353
## 25 2012     TENNESSEE 0.3040659 0.6489158      0 -0.344849852
## 26 2012         TEXAS 0.4062300 0.5645566      0 -0.158326533
## 27 2012          UTAH 0.2998041 0.6531010      0 -0.353296898
## 28 2012       VERMONT        NA 0.2490009     NA           NA
## 29 2012      VIRGINIA 0.5286595 0.4696081      1  0.059051401
## 30 2012    WASHINGTON 0.6045099 0.3954901      1  0.209019824
## 31 2012 WEST VIRGINIA 0.6057207 0.3647172      1  0.241003511
## 32 2012     WISCONSIN 0.5140886 0.4586034      1  0.055485276
## 33 2012       WYOMING 0.2114838 0.7389310      0 -0.527447148
```

12. **Next, we're going to do some analysis to map this data. Load in the state-level mapping data that we've worked with from the package `mapdata`**

```
counties <- map_data("county")
states <-map_data("state")
```

13. **Join the 2012 Senate election data to this mapping data. Be cautious about the format of the state names!**

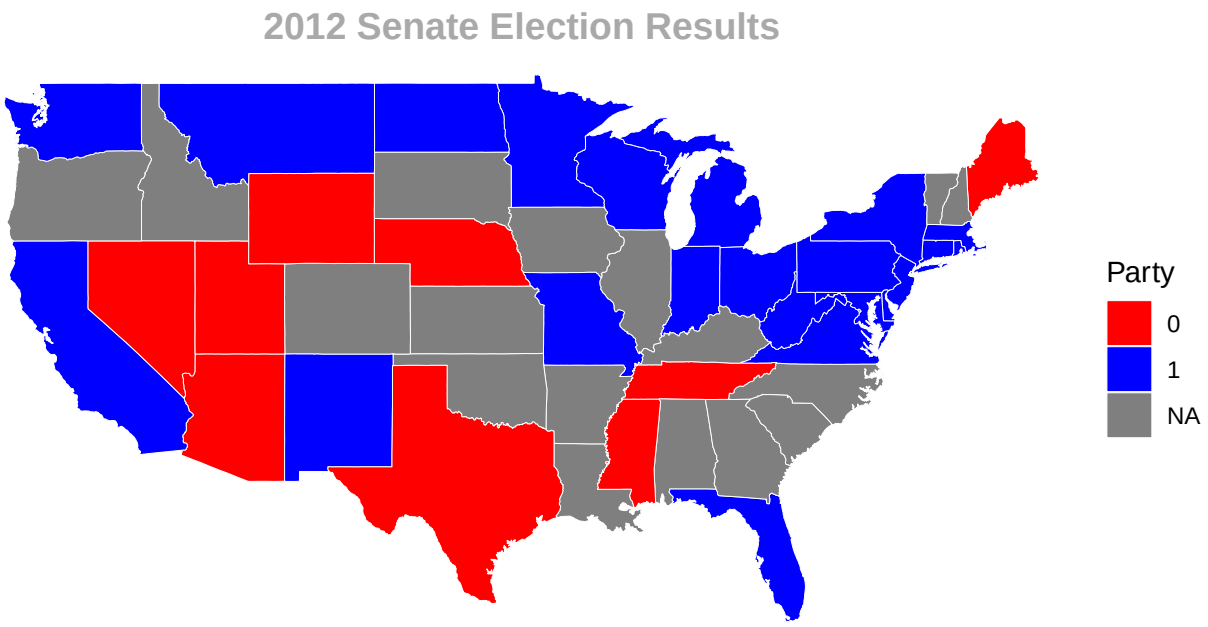
```
Question11$state = tolower(Question11$state)
map.states<-rename(states, "state" = region)

Question13 <- Question11 %>%
  full_join(map.states, by="state")
```

14. **Create a map that shows the winner of each Senate contest in 2012, with Democrats in blue and Republicans in red. If there was no Senate contest in a state (or if a party other than Democrats or Republicans won the seat), leave the state blank.**

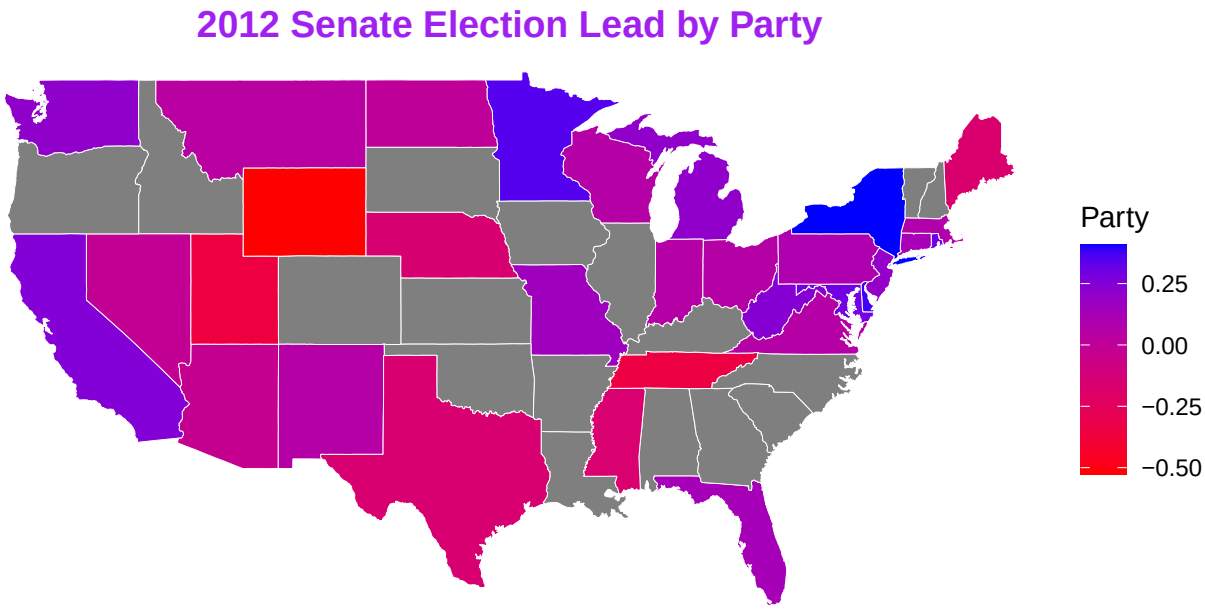
```
ggplot() +
  geom_polygon(data = Question13, aes(x=long, y=lat, group=group,
                                      fill=as.factor(demwin)), col="white", lwd=0.15) +
  coord_quickmap() +
```

```
  theme_void() +
  scale_fill_manual(values =c("red", "blue")) +
  theme(plot.title = element_text(size = 14, face = "bold", color = "darkgrey",
                                  hjust = 0.5)) +
  labs(title="2012 Senate Election Results", fill = "Party")
```



15. **Create a map that shades each state by the Democratic vote difference you created above. Again, If there was no Senate contest in a state (or if a party other than Democrats or Republicans won the seat), leave the state blank.**

```
ggplot() +
  geom_polygon(data = Question13, aes(x=long, y=lat, group=group,
                                      fill=demdiff), col="white", lwd=0.15) +
  coord_quickmap() +
  theme_void() +
  scale_fill_gradient(high="blue", low="red") +
  theme(plot.title = element_text(size = 14, face = "bold", color = "purple",
                                  hjust = 0.5)) +
  labs(title="2012 Senate Election Lead by Party", fill = "Party")
```


2012 Senate Election Lead by Party
Party
0.25
0.00
−0.25
−0.50