# Predicting the 2016 Election Results

*Namrata Kolla*

## 1. Mapping US Election Data

### Load appropriate libraries

```
library(tidyverse)
```

```
## Warning: package 'dplyr' was built under R version 3.4.2
```

```
library(stringr)
```

### Download and clean Presidential Results data

```
vote_dat <- read.csv("US_County_Level_Presidential_Results_08-16.csv.bz2")

str(vote_dat)
```

```
## 'data.frame':    3112 obs. of  14 variables:
##  $ fips_code : int  26041 48295 1127 48389 56017 20043 37183 37147 48497 21207 ...
##  $ county    : Factor w/ 1845 levels "Abbeville County",..: 468 973 1728 1381 779 495 1725 1306 1809
##  $ total_2008: int  19064 1256 28652 3077 2546 3564 442245 74884 20639 7475 ...
##  $ dem_2008  : int  9974 155 7420 1606 619 1115 250891 40501 4471 1569 ...
##  $ gop_2008  : int  8763 1093 20722 1445 1834 2372 187001 33927 15973 5779 ...
##  $ oth_2008  : int  327 8 510 26 93 77 4353 456 195 127 ...
##  $ total_2012: int  18043 1168 28497 2867 2495 3369 526805 76814 20692 7907 ...
##  $ dem_2012  : int  8330 119 6551 1649 523 885 286939 40701 3219 1445 ...
##  $ gop_2012  : int  9533 1044 21633 1185 1894 2397 232933 35534 17178 6346 ...
##  $ oth_2012  : int  180 5 313 33 78 87 6933 579 295 116 ...
##  $ total_2016: int  18467 1322 29243 3184 2535 3366 510940 78264 24661 8171 ...
##  $ dem_2016  : int  6431 135 4486 1659 400 584 298353 40967 3412 1093 ...
##  $ gop_2016  : int  11112 1159 24208 1417 1939 2601 193607 35191 20655 6863 ...
##  $ oth_2016  : int  924 28 549 108 196 181 18980 2106 594 215 ...
```

```
head(vote_dat)
```

```
##   fips_code             county total_2008 dem_2008 gop_2008 oth_2008
## 1     26041       Delta County      19064     9974     8763      327
## 2     48295    Lipscomb County       1256      155     1093        8
## 3      1127      Walker County      28652     7420    20722      510
## 4     48389      Reeves County       3077     1606     1445       26
## 5     56017 Hot Springs County       2546      619     1834       93
## 6     20043    Doniphan County       3564     1115     2372       77
##   total_2012 dem_2012 gop_2012 oth_2012 total_2016 dem_2016 gop_2016
## 1      18043     8330     9533      180      18467     6431    11112
## 2       1168      119     1044        5       1322      135     1159
## 3      28497     6551    21633      313      29243     4486    24208
## 4       2867     1649     1185       33       3184     1659     1417
## 5       2495      523     1894       78       2535      400     1939
## 6       3369      885     2397       87       3366      584     2601
```

```
##   oth_2016
## 1      924
## 2       28
## 3      549
## 4      108
## 5      196
## 6      181
```

```
dim(vote_dat %>% filter(fips_code < 1000))
```

```
## [1]  0 14
```

```
# no places using 3 digit FIPS
dim(vote_dat %>% filter(fips_code < 10000))
```

```
## [1] 287  14
```

```
# 287 places use 4 digit FIPS
dim(vote_dat %>% filter(fips_code > 9999))
```

```
## [1] 2825   14
```

```
# 2825 places use 5 digit FIPS
drop.cols <- c('total_2008', 'dem_2008', 'gop_2008', 'oth_2008')
vote_dat <- vote_dat %>% select(-one_of(drop.cols))
colnames(vote_dat)[1] <- "FIPS"
vote_dat$FIPS <- as.character(vote_dat$FIPS)
```

Notes:

- FIPS is a Federal Information Processing Standard, which uniquely identifies counties and county equivalents in the United States
- vote_dat provides fips code, name of county, total no. of people who voted in that county in 2008 election, how many of those votes were for the democrat, how many for the republican, how many for other, and same 4 columns repeated for 2012 and 2016 elections
- Would be interesting to see how votes in 2016 compared to previous presidential elections, so I'm keeping 2012 data.
- 2010 data is not available in the county_data file, so those columns were deleted using this source: https://stackoverflow.com/questions/35839408/r-dplyr-drop-multiple-columns
- 3112 county/county-equivalents in total

### Download and clean County data

```
cnty_dat <- read.csv("county_data.csv.bz2")
str(cnty_dat)
```

```
## 'data.frame':    3193 obs. of  116 variables:
##  $ SUMLEV               : int  40 50 50 50 50 50 50 50 50 50 ...
##  $ REGION               : int  3 3 3 3 3 3 3 3 3 3 ...
##  $ DIVISION             : int  6 6 6 6 6 6 6 6 6 6 ...
##  $ STATE                : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ COUNTY               : int  0 1 3 5 7 9 11 13 15 17 ...
##  $ STNAME               : Factor w/ 51 levels "Alabama","Alaska",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ CTYNAME              : Factor w/ 1927 levels "Abbeville County",..: 10 87 94 105 154 169 230 240 1
##  $ CENSUS2010POP        : int  4779736 54571 182265 27457 22915 57322 10914 20947 118572 34215 ...
##  $ ESTIMATESBASE2010    : int  4780131 54571 182265 27457 22919 57324 10911 20946 118586 34170 ...
##  $ POPESTIMATE2010      : int  4785492 54742 183199 27348 22861 57376 10892 20938 118468 34101 ...
```

```
##  $ POPESTIMATE2011      : int  4799918 55255 186653 27326 22736 57707 10722 20848 117736 34006 ...
##  $ POPESTIMATE2012      : int  4815960 55027 190403 27132 22645 57772 10654 20665 117208 34084 ...
##  $ POPESTIMATE2013      : int  4829479 54792 195147 26938 22501 57746 10576 20330 116475 34123 ...
##  $ POPESTIMATE2014      : int  4843214 54977 199745 26763 22511 57621 10712 20283 115837 33996 ...
##  $ POPESTIMATE2015      : int  4853875 55035 203690 26270 22561 57676 10455 20126 115285 34043 ...
##  $ POPESTIMATE2016      : int  4863300 55416 208563 25965 22643 57704 10362 19998 114611 33843 ...
##  $ NPOPCHG_2010         : int  5361 171 934 -109 -58 52 -19 -8 -118 -69 ...
##  $ NPOPCHG_2011         : int  14426 513 3454 -22 -125 331 -170 -90 -732 -95 ...
##  $ NPOPCHG_2012         : int  16042 -228 3750 -194 -91 65 -68 -183 -528 78 ...
##  $ NPOPCHG_2013         : int  13519 -235 4744 -194 -144 -26 -78 -335 -733 39 ...
##  $ NPOPCHG_2014         : int  13735 185 4598 -175 10 -125 136 -47 -638 -127 ...
##  $ NPOPCHG_2015         : int  10661 58 3945 -493 50 55 -257 -157 -552 47 ...
##  $ NPOPCHG_2016         : int  9425 381 4873 -305 82 28 -93 -128 -674 -200 ...
##  $ BIRTHS2010           : int  14231 151 516 70 44 183 39 65 318 81 ...
##  $ BIRTHS2011           : int  59689 636 2188 335 266 744 169 276 1384 401 ...
##  $ BIRTHS2012           : int  59066 614 2092 300 245 711 122 241 1357 393 ...
##  $ BIRTHS2013           : int  57939 574 2161 283 258 646 131 240 1309 406 ...
##  $ BIRTHS2014           : int  58906 640 2214 265 254 620 124 250 1317 425 ...
##  $ BIRTHS2015           : int  59034 636 2237 258 259 689 115 248 1289 423 ...
##  $ BIRTHS2016           : int  58556 631 2274 253 266 663 114 236 1237 419 ...
##  $ DEATHS2010           : int  11086 154 532 128 34 132 52 60 313 80 ...
##  $ DEATHS2011           : int  48817 507 1825 318 277 568 132 261 1326 441 ...
##  $ DEATHS2012           : int  48372 560 1882 293 239 593 117 272 1357 475 ...
##  $ DEATHS2013           : int  50845 582 1903 295 281 584 120 261 1411 452 ...
##  $ DEATHS2014           : int  49693 575 1989 313 250 587 115 288 1392 454 ...
##  $ DEATHS2015           : int  51407 475 2080 319 207 634 108 268 1427 465 ...
##  $ DEATHS2016           : int  52405 494 2113 314 237 622 111 241 1441 476 ...
##  $ NATURALINC2010       : int  3145 -3 -16 -58 10 51 -13 5 5 1 ...
##  $ NATURALINC2011       : int  10872 129 363 17 -11 176 37 15 58 -40 ...
##  $ NATURALINC2012       : int  10694 54 210 7 6 118 5 -31 0 -82 ...
##  $ NATURALINC2013       : int  7094 -8 258 -12 -23 62 11 -21 -102 -46 ...
##  $ NATURALINC2014       : int  9213 65 225 -48 4 33 9 -38 -75 -29 ...
##  $ NATURALINC2015       : int  7627 161 157 -61 52 55 7 -20 -138 -42 ...
##  $ NATURALINC2016       : int  6151 137 161 -61 29 41 3 -5 -204 -57 ...
##  $ INTERNATIONALMIG2010 : int  1360 33 66 2 2 5 7 0 6 7 ...
##  $ INTERNATIONALMIG2011 : int  4816 18 183 -4 10 -3 19 2 39 31 ...
##  $ INTERNATIONALMIG2012 : int  4695 2 176 -10 13 18 16 5 63 19 ...
##  $ INTERNATIONALMIG2013 : int  4179 2 209 -9 13 29 9 7 26 17 ...
##  $ INTERNATIONALMIG2014 : int  4732 6 239 -8 18 32 10 8 24 19 ...
##  $ INTERNATIONALMIG2015 : int  5110 8 257 -6 18 36 9 8 29 18 ...
##  $ INTERNATIONALMIG2016 : int  4738 7 243 -5 18 38 9 8 27 18 ...
##  $ DOMESTICMIG2010      : int  866 134 867 -54 -69 -3 -13 -11 -124 -71 ...
##  $ DOMESTICMIG2011      : int  -1416 321 2731 -31 -123 104 -242 -105 -788 -85 ...
##  $ DOMESTICMIG2012      : int  414 -294 3333 -192 -111 -67 -90 -160 -591 138 ...
##  $ DOMESTICMIG2013      : int  1619 -253 4178 -190 -148 -94 -92 -312 -647 70 ...
##  $ DOMESTICMIG2014      : int  420 118 3759 -113 2 -161 117 -8 -518 -117 ...
##  $ DOMESTICMIG2015      : int  -3114 -154 3492 -440 2 -81 -280 -150 -457 49 ...
##  $ DOMESTICMIG2016      : int  -864 228 4046 -248 34 -65 -101 -127 -462 -155 ...
##  $ NETMIG2010           : int  2226 167 933 -52 -67 2 -6 -11 -118 -64 ...
##  $ NETMIG2011           : int  3400 339 2914 -35 -113 101 -223 -103 -749 -54 ...
##  $ NETMIG2012           : int  5109 -292 3509 -202 -98 -49 -74 -155 -528 157 ...
##  $ NETMIG2013           : int  5798 -251 4387 -199 -135 -65 -83 -305 -621 87 ...
##  $ NETMIG2014           : int  5152 124 3998 -121 20 -129 127 0 -494 -98 ...
##  $ NETMIG2015           : int  1996 -146 3749 -446 20 -45 -271 -142 -428 67 ...
```

```
##  $ NETMIG2016           : int  3874 235 4289 -253 52 -27 -92 -119 -435 -137 ...
##  $ RESIDUAL2010         : int  -10 7 17 1 -1 -1 0 -2 -5 -6 ...
##  $ RESIDUAL2011         : int  154 45 177 -4 -1 54 16 -2 -41 -1 ...
##  $ RESIDUAL2012         : int  239 10 31 1 1 -4 1 3 0 3 ...
##  $ RESIDUAL2013         : int  627 24 99 17 14 -23 -6 -9 -10 -2 ...
##  $ RESIDUAL2014         : int  -630 -4 375 -6 -14 -29 0 -9 -69 0 ...
##  $ RESIDUAL2015         : int  1038 43 39 14 -22 45 7 5 14 22 ...
##  $ RESIDUAL2016         : int  -600 9 423 9 1 14 -4 -4 -35 -6 ...
##  $ GQESTIMATESBASE2010  : int  116185 455 2307 3193 2224 489 1690 333 2933 458 ...
##  $ GQESTIMATES2010      : int  116214 455 2307 3193 2224 489 1690 333 2934 458 ...
##  $ GQESTIMATES2011      : int  115521 455 2263 3379 2224 489 1690 333 2883 458 ...
##  $ GQESTIMATES2012      : int  115697 455 2242 3388 2225 489 1776 333 2959 458 ...
##  $ GQESTIMATES2013      : int  116984 455 2296 3388 2224 489 1717 333 2813 458 ...
##  $ GQESTIMATES2014      : int  119189 455 2333 3352 2241 489 1758 333 2796 458 ...
##  $ GQESTIMATES2015      : int  120174 455 2339 3198 2255 489 1656 333 2773 458 ...
##  $ GQESTIMATES2016      : int  119659 455 2341 3186 2252 489 1653 333 2776 458 ...
##  $ RBIRTH2011           : num  12.5 11.6 11.8 12.3 11.7 ...
##  $ RBIRTH2012           : num  12.3 11.1 11.1 11 10.8 ...
##  $ RBIRTH2013           : num  12 10.5 11.2 10.5 11.4 ...
##  $ RBIRTH2014           : num  12.18 11.66 11.21 9.87 11.29 ...
##  $ RBIRTH2015           : num  12.18 11.56 11.09 9.73 11.49 ...
##  $ RBIRTH2016           : num  12.05 11.43 11.03 9.69 11.77 ...
##  $ RDEATH2011           : num  10.19 9.22 9.87 11.63 12.15 ...
##  $ RDEATH2012           : num  10.06 10.16 9.98 10.76 10.53 ...
##  $ RDEATH2013           : num  10.54 10.6 9.87 10.91 12.45 ...
##  $ RDEATH2014           : num  10.3 10.5 10.1 11.7 11.1 ...
##  $ RDEATH2015           : num  10.6 8.64 10.31 12.03 9.19 ...
##  $ RDEATH2016           : num  10.79 8.95 10.25 12.02 10.49 ...
##  $ RNATURALINC2011      : num  2.268 2.346 1.963 0.622 -0.482 ...
##  $ RNATURALINC2012      : num  2.224 0.979 1.114 0.257 0.264 ...
##  $ RNATURALINC2013      : num  1.471 -0.146 1.338 -0.444 -1.019 ...
##  $ RNATURALINC2014      : num  1.905 1.184 1.14 -1.788 0.178 ...
##  $ RNATURALINC2015      : num  1.573 2.927 0.778 -2.3 2.307 ...
##  $ RNATURALINC2016      : num  1.266 2.481 0.781 -2.336 1.283 ...
##  $ RINTERNATIONALMIG2011: num  1.005 0.327 0.99 -0.146 0.439 ...
##   [list output truncated]
```

```
#head(cnty_dat)

  # Delete unnecessary columns
drop.cols <- c('SUMLEV','DIVISION','CENSUS2010POP','ESTIMATESBASE2010','POPESTIMATE2010','POPESTIMATE20
cnty_dat <- cnty_dat %>% select(-one_of(drop.cols))

  # Create concatenated FIPS code for merging with other file
cnty_dat$COUNTY <- str_pad(cnty_dat$COUNTY, 3, pad = "0")
cnty_dat <- cnty_dat %>% mutate(FIPS=paste(STATE,COUNTY))
cnty_dat$FIPS <- gsub('\\s+','',cnty_dat$FIPS)
```

Notes:

- Data about any year besides 2012 and 2016 were deleted
- Went from 116 columns to 37 columns
- Padded county variable with leading 0s so it would match up with Presidential Results file's FIPS code when merged. Used this source about padding: https://stackoverflow.com/questions/5812493/adding-leading-zeros-using-r

- Removed whitespace from strings using this source: https://stackoverflow.com/questions/20760547/removing-whitespace-from-a-whole-data-frame-in-r
- 3193 county/county-equivalents in total

### Join data sets by FIPS code and more cleaning

```
elections_df <- inner_join(vote_dat, cnty_dat, by = "FIPS")

  # delete uninteresting variables:
elections_df <- elections_df %>%
  select(-one_of('CTYNAME','RINTERNATIONALMIG2012','RINTERNATIONALMIG2016','RDOMESTICMIG2012','RDOMESTI
```

- 3111 counties/county-equivalents in total (82 counties lost)

### Compute additional interesting variables

```
  # create variable that measures % democrats in 2016:
elections_df <- elections_df %>% mutate(pct_dem_2016=dem_2016/total_2016)
elections_df <- elections_df %>% mutate(pct_gop_2016=gop_2016/total_2016)
elections_df <- elections_df %>% mutate(pct_oth_2016=oth_2016/total_2016)
  # create variable that measures change in % democrats:
elections_df <- elections_df %>%
  mutate(dem_pct_change=(dem_2016/total_2016)-(dem_2012/total_2012))
  # create categorical variable that measures whether net migration in 2016 was positive (TRUE) or nega
elections_df <- elections_df %>% mutate(rate_mig_pos_neg=(RNETMIG2016>=0))

  # summary of NATIONAL results in 2016:
elections_df_summary <- elections_df %>% summarise(votes_dem = sum(dem_2016),
                           votes_gop = sum(gop_2016),
                           votes_tot = sum(total_2016),
                           pop_vote_dem = votes_dem/votes_tot,
                           pop_vote_gop = votes_gop/votes_tot)
```

Notes: * dem_pct_2016 = gives the % of voters who voted for the democratic candidate in that county in 2016 * dem_pct_change = gives the change in % of voters who voted for the democratic candidate in that county between 2016 and 2012 * rate_mig_pos_neg = categorical variables that tells whether the rate of net migration was positive (TRUE) or negative (FALSE). If exactly 0, it was labeled positive. * elections_df_summary is a fun table that shows what % of the national population voted for democrats and republicans

### Describing the data and the more interesting variables

```
str(elections_df)
```

```
## 'data.frame':    3111 obs. of  31 variables:
##  $ FIPS           : chr  "26041" "48295" "1127" "48389" ...
##  $ county         : Factor w/ 1845 levels "Abbeville County",..: 468 973 1728 1381 779 495 1725 130
##  $ total_2012     : int  18043 1168 28497 2867 2495 3369 526805 76814 20692 7907 ...
##  $ dem_2012       : int  8330 119 6551 1649 523 885 286939 40701 3219 1445 ...
##  $ gop_2012       : int  9533 1044 21633 1185 1894 2397 232933 35534 17178 6346 ...
##  $ oth_2012       : int  180 5 313 33 78 87 6933 579 295 116 ...
```

```
##  $ total_2016      : int  18467 1322 29243 3184 2535 3366 510940 78264 24661 8171 ...
##  $ dem_2016        : int  6431 135 4486 1659 400 584 298353 40967 3412 1093 ...
##  $ gop_2016        : int  11112 1159 24208 1417 1939 2601 193607 35191 20655 6863 ...
##  $ oth_2016        : int  924 28 549 108 196 181 18980 2106 594 215 ...
##  $ REGION          : int  2 3 3 3 4 2 3 3 3 3 ...
##  $ STATE           : int  26 48 1 48 56 20 37 37 48 21 ...
##  $ COUNTY          : chr  "041" "295" "127" "389" ...
##  $ STNAME          : Factor w/ 51 levels "Alabama","Alaska",..: 23 44 1 44 51 17 34 34 44 18 ...
##  $ POPESTIMATE2012 : int  36838 3457 66211 13925 4843 7869 952296 172913 60412 17566 ...
##  $ POPESTIMATE2016 : int  36202 3487 64967 14921 4679 7664 1046791 177220 64455 17722 ...
##  $ NPOPCHG_2012    : int  -97 112 -450 148 24 -82 23088 2157 433 -110 ...
##  $ NPOPCHG_2016    : int  -205 -66 -324 132 -66 -92 24817 1038 1619 86 ...
##  $ NETMIG2012      : int  -32 106 -251 87 27 -82 15221 1168 229 -115 ...
##  $ NETMIG2016      : int  -138 -98 -145 23 -47 -85 17029 257 1354 111 ...
##  $ RBIRTH2012      : num  9.76 10.29 12.43 11.05 11.38 ...
##  $ RBIRTH2016      : num  10.5 12.8 12.2 14.2 10.4 ...
##  $ RDEATH2012      : num  11.49 9.11 15.46 6.5 11.8 ...
##  $ RDEATH2016      : num  11.7 5.4 14.6 6.6 14.6 ...
##  $ RNETMIG2012     : num  -0.868 31.167 -3.778 6.281 5.589 ...
##  $ RNETMIG2016     : num  -3.8 -27.84 -2.23 1.55 -9.97 ...
##  $ pct_dem_2016    : num  0.348 0.102 0.153 0.521 0.158 ...
##  $ pct_gop_2016    : num  0.602 0.877 0.828 0.445 0.765 ...
##  $ pct_oth_2016    : num  0.05 0.0212 0.0188 0.0339 0.0773 ...
##  $ dem_pct_change  : num  -0.113432 0.000234 -0.07648 -0.054123 -0.051828 ...
##  $ rate_mig_pos_neg: logi  FALSE FALSE FALSE TRUE FALSE FALSE ...
```

```
summary(elections_df)
```

```
##      FIPS                        county       total_2012
##  Length:3111        Washington County:  30   Min.   :     64
##  Class :character   Jefferson County :  25   1st Qu.:   4772
##  Mode  :character   Franklin County  :  24   Median :  10736
##                     Jackson County   :  23   Mean   :  39517
##                     Lincoln County   :  23   3rd Qu.:  27634
##                     Madison County   :  19   Max.   :2427869
##                     (Other)          :2967
##     dem_2012          gop_2012         oth_2012         total_2016
##  Min.   :      5   Min.   :    54   Min.   :    0.0   Min.   :     64
##  1st Qu.:   1554   1st Qu.:  2890   1st Qu.:   70.0   1st Qu.:   4819
##  Median :   3952   Median :  6398   Median :  168.0   Median :  10935
##  Mean   :  19996   Mean   : 18896   Mean   :  625.3   Mean   :  40908
##  3rd Qu.:  11108   3rd Qu.: 15950   3rd Qu.:  462.5   3rd Qu.:  28675
##  Max.   :1672164   Max.   :699600   Max.   :56105.0   Max.   :2314275
##
##     dem_2016          gop_2016         oth_2016          REGION
##  Min.   :      4   Min.   :    57   Min.   :     3   Min.   :1.000
##  1st Qu.:   1164   1st Qu.:  3207   1st Qu.:   165   1st Qu.:2.000
##  Median :   3140   Median :  7117   Median :   440   Median :3.000
##  Mean   :  19566   Mean   : 19350   Mean   :  1992   Mean   :2.656
##  3rd Qu.:   9536   3rd Qu.: 17396   3rd Qu.:  1394   3rd Qu.:3.000
##  Max.   :1654626   Max.   :590465   Max.   :117058   Max.   :4.000
##
##      STATE          COUNTY              STNAME     POPESTIMATE2012
##  Min.   : 1.00   Length:3111        Texas   : 254   Min.   :     81
##  1st Qu.:19.00   Class :character   Georgia : 159   1st Qu.:  11225
```

```
##  Median :29.00   Mode  :character   Virginia: 133   Median :  26024
##  Mean   :30.54                      Kentucky: 120   Mean   : 100692
##  3rd Qu.:46.00                      Missouri: 115   3rd Qu.:  67882
##  Max.   :56.00                      Kansas  : 105   Max.   :9953555
##                                     (Other) :2225
##  POPESTIMATE2016     NPOPCHG_2012      NPOPCHG_2016
##  Min.   :     113   Min.   :-8920.0   Min.   :-21324.0
##  1st Qu.:   11194   1st Qu.: -122.0   1st Qu.:  -105.0
##  Median :   26027   Median :  -14.0   Median :     3.0
##  Mean   :  103623   Mean   :  747.8   Mean   :   715.7
##  3rd Qu.:   67968   3rd Qu.:  205.5   3rd Qu.:   246.5
##  Max.   :10137915   Max.   :79489.0   Max.   : 81360.0
##
##    NETMIG2012         NETMIG2016         RBIRTH2012       RBIRTH2016
##  Min.   :-15683.0   Min.   :-47810.0   Min.   : 2.281   Min.   : 0.00
##  1st Qu.:  -158.0   1st Qu.:  -123.0   1st Qu.: 9.933   1st Qu.: 9.93
##  Median :   -33.0   Median :   -12.0   Median :11.415   Median :11.32
##  Mean   :   288.8   Mean   :   322.0   Mean   :11.592   Mean   :11.51
##  3rd Qu.:    94.0   3rd Qu.:   166.5   3rd Qu.:12.958   3rd Qu.:12.80
##  Max.   : 43669.0   Max.   : 53377.0   Max.   :30.060   Max.   :29.43
##
##    RDEATH2012        RDEATH2016      RNETMIG2012        RNETMIG2016
##  Min.   : 0.8032   Min.   : 0.000   Min.   :-147.727   Min.   :-67.4442
##  1st Qu.: 8.3882   1st Qu.: 8.627   1st Qu.:  -7.009   1st Qu.: -6.2674
##  Median :10.1080   Median :10.337   Median :  -1.652   Median : -0.7676
##  Mean   :10.1547   Mean   :10.259   Mean   :  -1.529   Mean   : -0.5072
##  3rd Qu.:11.8431   3rd Qu.:12.055   3rd Qu.:   3.815   3rd Qu.:  5.4924
##  Max.   :23.0187   Max.   :22.694   Max.   : 120.257   Max.   :157.7962
##
##   pct_dem_2016      pct_gop_2016      pct_oth_2016      dem_pct_change
##  Min.   :0.03145   Min.   :0.04122   Min.   :0.003344   Min.   :-0.23445
##  1st Qu.:0.20478   1st Qu.:0.54980   1st Qu.:0.029032   1st Qu.:-0.10210
##  Median :0.28486   Median :0.66745   Median :0.043750   Median :-0.06297
##  Mean   :0.31693   Mean   :0.63642   Mean   :0.046647   Mean   :-0.06784
##  3rd Qu.:0.39948   3rd Qu.:0.75150   3rd Qu.:0.058225   3rd Qu.:-0.03351
##  Max.   :0.92847   Max.   :0.95273   Max.   :0.353477   Max.   : 0.09346
##
##  rate_mig_pos_neg
##  Mode :logical
##  FALSE:1659
##  TRUE :1452
##
##
##
##
```

```
elections_df_summary
```

```
##   votes_dem votes_gop votes_tot pop_vote_dem pop_vote_gop
## 1  60871442  60197547 127266431    0.4782993    0.4730041
```

Notes:

- This data includes voting records in the 2016 and 2012 elections for 3111 different counties
- Variables that start with total_, dem_, gop_, and oth_ are the raw # of people who voted for that particular party. These variables have really wide ranges because the population varies dramatically

from one county to another

- A better measure is the variables that start with pct_. These tell the percent of the population that voted for that particular party. As expected, these range from 0 to 1.0
- The mean % votes for dems in 2016 was 0.32 while the mean % votes for gop was 0.63. While this does not take differences in population size between counties into account, it fits the overall finding that GOP won in 2016.
- elections_df_summary shows that the popular vote nationallly was for democrats. It was a very close margin: 47.8% for democrats and 47.3% for the GOP Thus it is surprising that the GOP won, but this happens occassionally due to features like the electoral college, which I won't get into.
- The mean dem_pct_change is negative (-0.068) which supports that the final results: the percent of the county that voted for democrats decreased from 2012 to 2016
- rate_mig_pos_neg has more false than trues, which suggests more counties had people leaving the county than entering it. This corresponds with RNETMIG2016 having a mean of -0.5072.

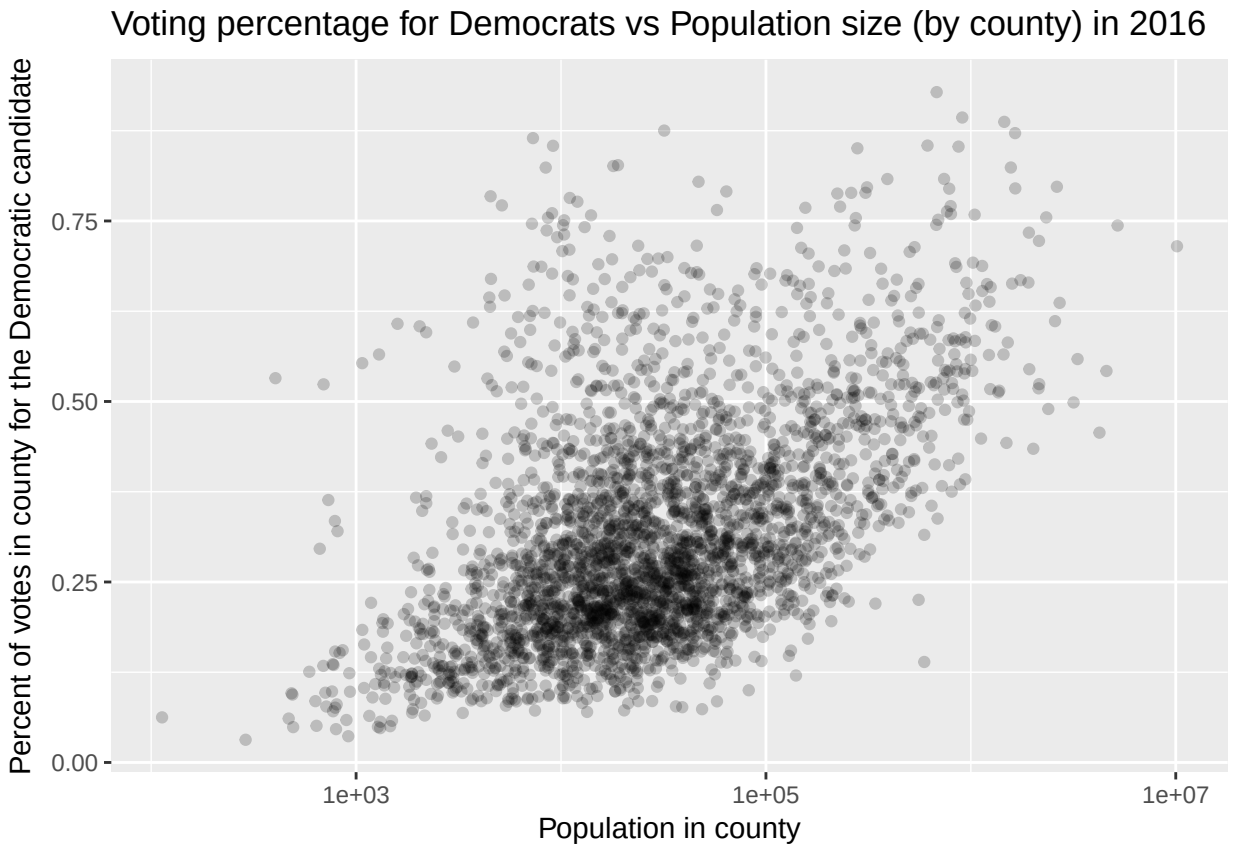
Variables to analyze:

- Population size
- dem_pct_change
- rate_mig_pos_neg (categorical) or RNETMIG2016

Interesting hypothesis to consider: More people are moving into cities -> a small percent of counties (with the cities) are growing in population size while a much larger percent of counties are becoming more rural -> the smaller the population size the more likely to vote GOP -> more votes for GOP because a larger geographical area ends up voting for GOP (though population doesn't match up)

### Plot % votes for Democrats vs County Population

```
ggplot(elections_df) +
  geom_point(aes(x=POPESTIMATE2016,y=pct_dem_2016),alpha=0.2) +
  scale_x_log10() +
  ylab('Percent of votes in county for the Democratic candidate') +
  xlab('Population in county') +
  ggtitle('Voting percentage for Democrats vs Population size (by county) in 2016')
```



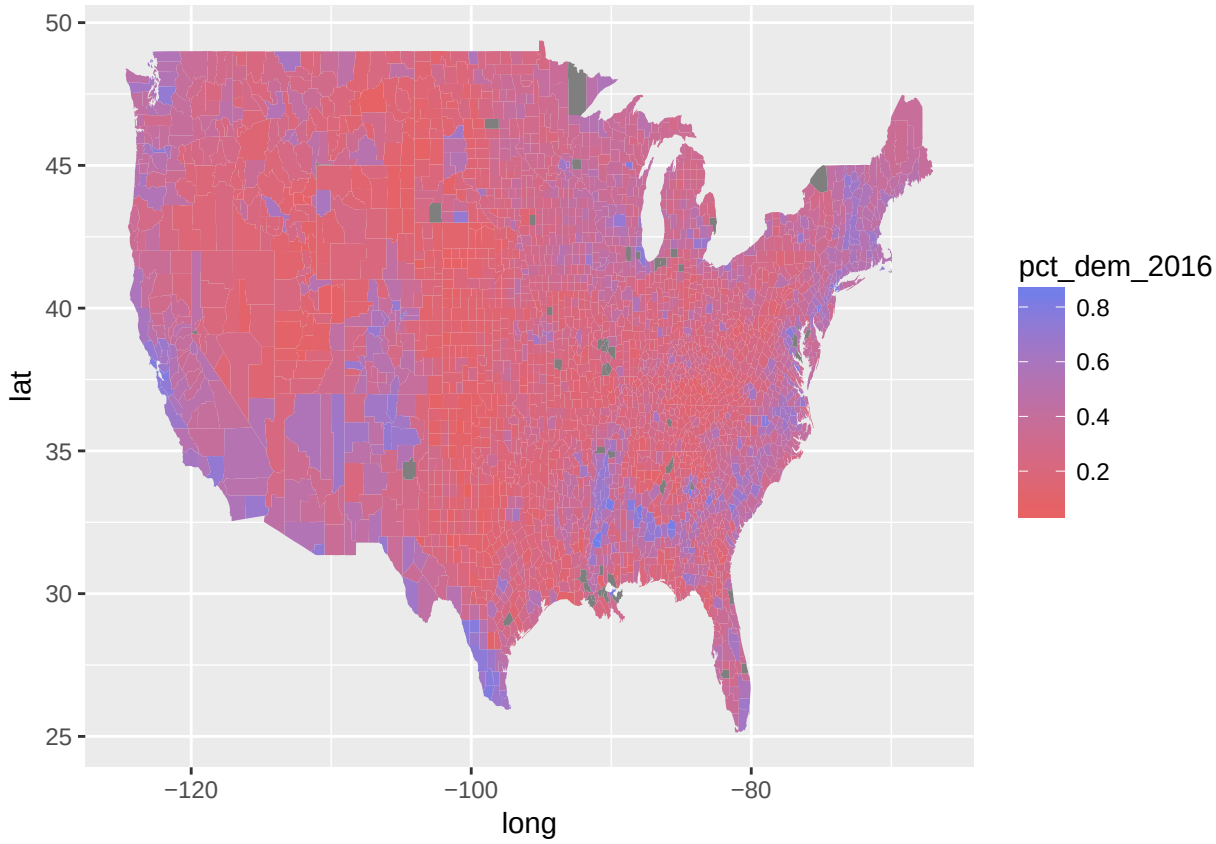

Notes:

- Generally as the population size of a county increases, the percent of votes for the Democratic candidate also increases
- Those with populations >= 1,000,000 almost always vote >50% democrat
- Those with populations < 100,000 are more likely to vote gop
- There's a big cluster of data from population 10,000 to 100,000 probably because most counties have populations this size

**Create map of % votes for democrats**

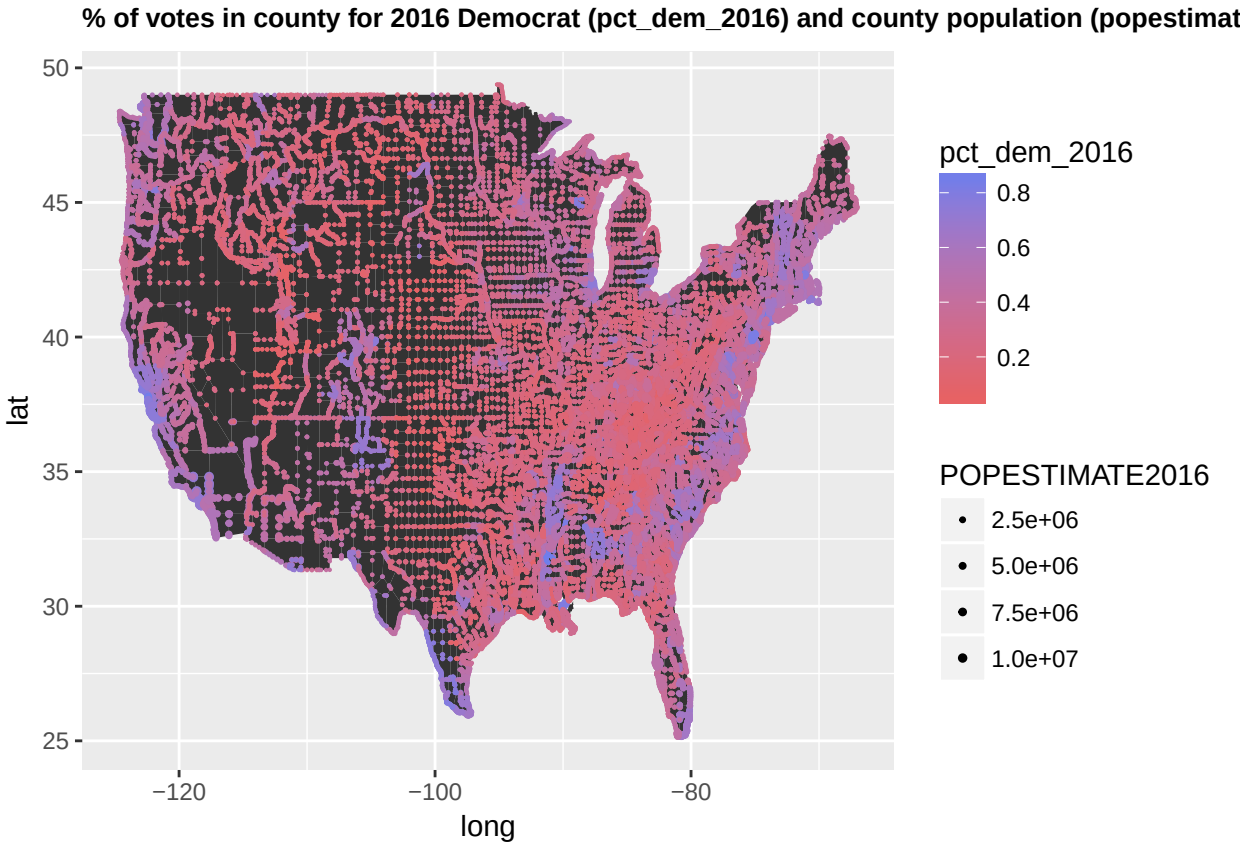
```
elections_df_plot <- elections_df

# convert state and county names to lowercase
elections_df_plot$STNAME <- tolower(elections_df_plot$STNAME)
elections_df_plot$county <- tolower(elections_df_plot$county)

# remove last word from county variable
elections_df_plot$county <- gsub("\\s*\\w*$", "", elections_df_plot$county)

# download the coordinates for various counties
counties <- map_data(map="county",exact=FALSE)

# change variable names to match with elections_df_plot
colnames(counties)[5] <- "STNAME"
colnames(counties)[6] <- "county"
```

```
# apply election results to the county map
elections_mapped <- left_join(counties,elections_df_plot, by = c("county","STNAME"))

ggplot(elections_mapped,aes(long,lat)) +
  geom_polygon(aes(group=group,fill=pct_dem_2016)) +
  scale_fill_gradient(low = "#e86161", high = "#677fef")
```




```
ggplot(elections_mapped,aes(long,lat)) +
  geom_polygon(aes(group=group)) +
  geom_point(aes(long,lat,color=pct_dem_2016,size=POPESTIMATE2016)) +
  scale_colour_gradient(low = "#e86161", high = "#677fef") +
  scale_size_continuous(range = c(0.1, 1)) +
  ggtitle('% of votes in county for 2016 Democrat (pct_dem_2016) and county population (popestimate2016
  theme(plot.title = element_text(size = 10,face="bold"))
```

```
## Warning: Removed 1464 rows containing missing values (geom_point).
```




Notes:

- Used this source for code on how to remove last word from county variable: https://stackoverflow.com/questions/13093931/r-remove-last-word-from-string
- Used this source on how to decrease font size of ggtitle: https://stackoverflow.com/questions/35458230/how-to-increase-the-font-size-of-ggtitle-in-ggplot2

What I did, what worked, and what didn't work:

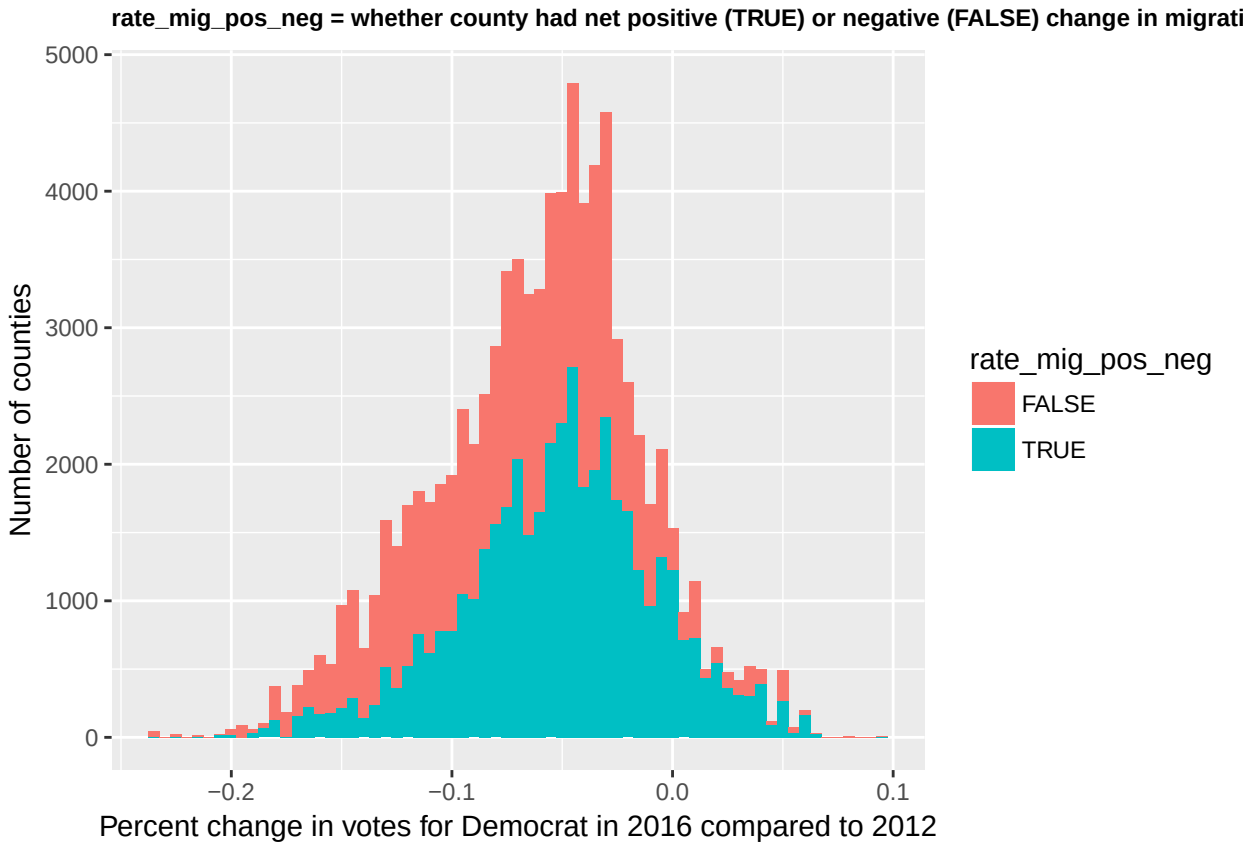
- At first, I only visualized the percentage of votes for the democratic party in each county on the map of counties in the US using a fill gradient. This worked really well because it was easy to see the county borders and to pick out larger areas where there were signficantly more votes for democrats (like major cities)
- Then, I visualized the percentage of votes for the democratic party in each county using a fill gradient on dots, and then made the size of the dots the population size. This worked well for the areas where there were not too many counties; the sparse counties generally voted more Republican. However in areas where the counties were close together, this visualization did not work well. It became hard to see the county borders, and dots overlapping each other made some of those spots where democrats took a large majority of the vote (like major cities) very difficult to see.

### Another visualization

```
ggplot(elections_mapped) +
  geom_histogram(aes(x=dem_pct_change,fill=rate_mig_pos_neg),binwidth=0.005) +
  xlab('Percent change in votes for Democrat in 2016 compared to 2012') +
  ylab('Number of counties') +
```

```
  ggtitle('rate_mig_pos_neg = whether county had net positive (TRUE) or negative (FALSE) change in migra
  theme(plot.title = element_text(size = 9,face="bold"))
```

```
## Warning: Removed 1464 rows containing non-finite values (stat_bin).
```




Notes:

- I created a histogram that looks at how percentage change in votes for Democrats from 2012 to 2016, faceted by whether the rate of migration increased or decreased from 2012 to 2016.
- I found that most of the data was lower than 0.0, which says that regardless of the migration rates fewer percentages of people in most counties were voting for Democrats in 2016 than 2012.
- At the same time, those counties who saw a positive change in rate of migration (more people migrating into the county) had data in blue that was slightly further right than those counties who saw a negative change in rate of migration (fewer people migrating into the county) with data in red. This suggests that growing counties (e.g. counties containing cities) become more likely to vote Democratic. It also suggests that shrinking counties (i.e. counties becoming more rural) become more likely to vote not-Democratic.

---

## 2. 2016 Election Model

### Add logical variable that democrats got more vote than GOP

```
  # variable describing whether more democrat or gop votes happened in the county
elections_df <- elections_df %>% mutate(dems_more=dem_2016>gop_2016)
elections_df$dems_more[elections_df$dems_more == FALSE] <- 0
```

```
  # variable describing % of democrat votes in 2012
elections_df <- elections_df %>% mutate(pct_dem_2012=dem_2012/total_2012)
```

The variable we're predicting is "dems_more" which is a logical of whether democrats got more votes (TRUE) or GOP got more votes (FALSE)

The variables I consider relevant to that are: * percent of votes for democratic candidate in 2012 election (pct_dem_2012) because the greater the percentage who voted in the 2012 election for Democrat, the more likely they were to stick with it for 2016 * region of the country (REGION) because regions like the South have historically voted GOP while regions like the west have historically voted Democrat * population size in 2016 (POPESTIMATE2016) because lower the population size probably leads to more votes for GOP * population change in 2016 (NPOPCHG_2016) because more people leaving the county lowers population size and increases chances of vote going to GOP * death rate (RDEATH2016)... I cannot explain connection, but I have noticed that poorer states or states where there are higher cancer rates from fossil fuel extraction (where death rate would be high) tend to vote GOP

**Attempt 1: includes seemingly relevant variables**

```
m <- glm(dems_more~pct_dem_2012+REGION+POPESTIMATE2016+NPOPCHG_2016+RDEATH2016, data=elections_df, famil
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(m)
```

```
##
## Call:
## glm(formula = dems_more ~ pct_dem_2012 + REGION + POPESTIMATE2016 +
##     NPOPCHG_2016 + RDEATH2016, family = binomial(link = "logit"),
##     data = elections_df)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -3.02221  -0.03383  -0.00306  -0.00018   2.68448
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -2.603e+01  1.807e+00 -14.407  < 2e-16 ***
## pct_dem_2012     5.089e+01  3.406e+00  14.945  < 2e-16 ***
## REGION           1.111e+00  1.504e-01   7.384 1.53e-13 ***
## POPESTIMATE2016  4.750e-06  8.702e-07   5.459 4.79e-08 ***
## NPOPCHG_2016    -1.514e-04  5.719e-05  -2.648  0.00811 **
## RDEATH2016      -5.012e-01  5.970e-02  -8.395  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2696.29  on 3110  degrees of freedom
## Residual deviance:  461.58  on 3105  degrees of freedom
## AIC: 473.58
##
## Number of Fisher Scoring iterations: 9
```

```
#install.packages('pscl')
library(pscl)
```

```
## Warning: package 'pscl' was built under R version 3.4.2
```

```
## Classes and Methods for R developed in the
## Political Science Computational Laboratory
## Department of Political Science
## Stanford University
## Simon Jackman
## hurdle and zeroinfl functions by Achim Zeileis
```

```
# How did model do?
tp <- predict(m, type="response") > 0.5
pR2(m)
```

```
##           llh       llhNull            G2      McFadden          r2ML
##  -230.7889366 -1348.1457842  2234.7136952     0.8288101     0.5124325
##          r2CU
##     0.8840195
```

```
table(elections_df$dems_more,tp)
```

```
##    tp
##     FALSE TRUE
##   0  2578   47
##   1    52  434
```

```
table(elections_df$dems_more,tp) %>% diag() %>% sum()
```

```
## [1] 3012
```

```
3012/(52+47+3012)
```

```
## [1] 0.9681774
```

Notes:

- As expected, pct_dem_2012 was the biggest predictor of whether democrats won in that county in 2016.
- REGION was the next best predictor
- Then, RDEATH2016, NPOPCHG_2016, and POPESTIMATE2016
- Predicted outcome correctly 96.8% of the time
- Logistic regression had a relatively strong fit; pseudo R-squared (McFaddens) has a value of .829

### Attempt 2: Add on birth rate variable

```
m2 <- glm(dems_more~pct_dem_2012+REGION+POPESTIMATE2016+NPOPCHG_2016+RDEATH2016+RBIRTH2016, data=electi
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(m2)
```

```
##
## Call:
## glm(formula = dems_more ~ pct_dem_2012 + REGION + POPESTIMATE2016 +
##     NPOPCHG_2016 + RDEATH2016 + RBIRTH2016, family = binomial(link = "logit"),
##     data = elections_df)
```

```
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -3.06258  -0.03380  -0.00304  -0.00018   2.64754
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -2.584e+01  1.837e+00 -14.065  < 2e-16 ***
## pct_dem_2012     5.095e+01  3.414e+00  14.924  < 2e-16 ***
## REGION           1.122e+00  1.521e-01   7.380 1.58e-13 ***
## POPESTIMATE2016  4.791e-06  8.768e-07   5.464 4.64e-08 ***
## NPOPCHG_2016    -1.513e-04  5.750e-05  -2.632  0.00849 **
## RDEATH2016      -4.989e-01  5.976e-02  -8.348  < 2e-16 ***
## RBIRTH2016      -2.396e-02  4.183e-02  -0.573  0.56672
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2696.29  on 3110  degrees of freedom
## Residual deviance:  461.25  on 3104  degrees of freedom
## AIC: 475.25
##
## Number of Fisher Scoring iterations: 9
```

```
tp2 <- predict(m2, type="response") > 0.5
pR2(m2)
```

```
##           llh       llhNull            G2      McFadden          r2ML
##  -230.6248832 -1348.1457842  2235.0418019     0.8289318     0.5124839
##          r2CU
##     0.8841082
```

```
table(elections_df$dems_more,tp2)
```

```
##    tp2
##     FALSE TRUE
##   0  2579   46
##   1    52  434
```

Notes:

- pct_dem_2012 is still the biggest predictor of whether democrats won in that county in 2016
- RBIRTH2016 is not a predictor variable... coefficient is small and it is not statistically significant
- Prediction accuracy and R-squared values are still the same

### Attempt 3: Turn region into a factor variable

```
# Attempt 3: Turn region into a factor variable
m3 <- glm(dems_more~pct_dem_2012+as.factor(REGION)+POPESTIMATE2016+NPOPCHG_2016+RDEATH2016, data=electi
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(m3)
```

```
##
```

```
## Call:
## glm(formula = dems_more ~ pct_dem_2012 + as.factor(REGION) +
##     POPESTIMATE2016 + NPOPCHG_2016 + RDEATH2016, family = binomial(link = "logit"),
##     data = elections_df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.2699  -0.0257  -0.0020  -0.0001   2.4675
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -2.511e+01  1.891e+00 -13.276  < 2e-16 ***
## pct_dem_2012        5.422e+01  3.870e+00  14.010  < 2e-16 ***
## as.factor(REGION)2 -9.189e-01  3.797e-01  -2.420   0.0155 *
## as.factor(REGION)3  2.080e+00  4.089e-01   5.087 3.64e-07 ***
## as.factor(REGION)4  2.331e+00  4.817e-01   4.839 1.31e-06 ***
## POPESTIMATE2016     4.640e-06  8.654e-07   5.362 8.22e-08 ***
## NPOPCHG_2016       -1.636e-04  5.390e-05  -3.035   0.0024 **
## RDEATH2016         -5.737e-01  6.709e-02  -8.552  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2696.29  on 3110  degrees of freedom
## Residual deviance:  415.87  on 3103  degrees of freedom
## AIC: 431.87
##
## Number of Fisher Scoring iterations: 10
```

```
tp3 <- predict(m3, type="response") > 0.5
pR2(m3)
```

```
##           llh       llhNull            G2      McFadden          r2ML
##  -207.9368145 -1348.1457842  2280.4179395     0.8457609     0.5195431
##          r2CU
##     0.8962862
```

```
table(elections_df$dems_more,tp3)
```

```
##    tp3
##     FALSE TRUE
##   0  2589   36
##   1    49  437
```

```
table(elections_df$dems_more,tp3) %>% diag() %>% sum()
```

```
## [1] 3026
```

```
3026/(36+49+3026)
```

```
## [1] 0.9726776
```

Notes:

- pct_dem_2012 is still the biggest predictor, but being in Region 3 or 4 is not far behind. REGIONS 3 and 4 correspond with the South and the West. Source: https://www2.census.gov/geo/docs/maps-data/maps/reg_div.txt

- This makes a lot of sense because the South votes broadly GOP while the West votes broadly Democrat. Comparatively, the Northeast (REGION 1) and the Midwest (REGION 2) have more swing states.
- Then, RDEATH2016, NPOPCHG_2016, and POPESTIMATE2016
- Predicted outcome correctly 97.3% of the time
- Logistic regression had a relatively strong fit; pseudo R-squared (McFaddens) has a value of .846

**Attempt 4: Add state variable, as a factor (BEST REGRESSION I GOT!)**

```
m4 <- glm(dems_more~pct_dem_2012+as.factor(REGION)+POPESTIMATE2016+NPOPCHG_2016+RDEATH2016+as.factor(ST
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(m4)
```

```
##
## Call:
## glm(formula = dems_more ~ pct_dem_2012 + as.factor(REGION) +
##     POPESTIMATE2016 + NPOPCHG_2016 + RDEATH2016 + as.factor(STATE),
##     family = binomial(link = "logit"), data = elections_df)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.80325  -0.01064  -0.00047  -0.00001   2.56778
##
## Coefficients: (3 not defined because of singularities)
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -3.066e+01  3.067e+00  -9.996  < 2e-16 ***
## pct_dem_2012        6.672e+01  5.433e+00  12.281  < 2e-16 ***
## as.factor(REGION)2 -1.706e+00  1.558e+00  -1.095  0.27359
## as.factor(REGION)3  4.392e+00  1.934e+00   2.270  0.02319 *
## as.factor(REGION)4 -9.752e-02  3.822e+00  -0.026  0.97965
## POPESTIMATE2016     4.226e-06  9.737e-07   4.341 1.42e-05 ***
## NPOPCHG_2016       -9.339e-05  5.577e-05  -1.675  0.09399 .
## RDEATH2016         -7.188e-01  8.848e-02  -8.124 4.52e-16 ***
## as.factor(STATE)4  -1.314e+00  3.881e+00  -0.339  0.73495
## as.factor(STATE)5   5.993e-01  1.823e+00   0.329  0.74233
## as.factor(STATE)6   4.437e+00  3.710e+00   1.196  0.23174
## as.factor(STATE)8   1.938e+00  3.590e+00   0.540  0.58927
## as.factor(STATE)9  -7.219e-01  1.906e+00  -0.379  0.70491
## as.factor(STATE)10 -8.042e+00  1.424e+01  -0.565  0.57218
## as.factor(STATE)11 -1.019e+01  1.773e+04  -0.001  0.99954
## as.factor(STATE)12 -4.632e+00  2.004e+00  -2.311  0.02082 *
## as.factor(STATE)13 -1.687e+00  1.303e+00  -1.295  0.19543
## as.factor(STATE)15  7.877e+00  8.601e+03   0.001  0.99927
## as.factor(STATE)16  2.961e+00  3.936e+00   0.752  0.45190
## as.factor(STATE)17  2.586e+00  8.511e-01   3.038  0.00238 **
## as.factor(STATE)18  1.961e+00  1.293e+00   1.517  0.12925
## as.factor(STATE)19 -7.057e-01  8.148e-01  -0.866  0.38647
## as.factor(STATE)20  7.726e-01  9.104e+00   0.085  0.93237
## as.factor(STATE)21 -1.734e+00  1.847e+00  -0.939  0.34777
## as.factor(STATE)22 -1.781e-01  1.822e+00  -0.098  0.92213
## as.factor(STATE)23  1.136e+00  1.637e+00   0.694  0.48752
## as.factor(STATE)24 -1.842e+00  1.841e+00  -1.000  0.31717
## as.factor(STATE)25  2.133e+01  3.447e+03   0.006  0.99506
```

```
## as.factor(STATE)26  9.332e-01  1.017e+00   0.917  0.35894
## as.factor(STATE)27  8.906e-01  8.415e-01   1.058  0.28986
## as.factor(STATE)28 -1.497e+00  1.395e+00  -1.073  0.28321
## as.factor(STATE)29  4.548e+00  2.116e+00   2.149  0.03161 *
## as.factor(STATE)30  1.586e+00  3.793e+00   0.418  0.67588
## as.factor(STATE)31  3.824e+00  1.299e+00   2.943  0.00325 **
## as.factor(STATE)32  4.359e+00  4.415e+00   0.987  0.32354
## as.factor(STATE)33 -4.002e-01  1.790e+00  -0.224  0.82308
## as.factor(STATE)34  2.138e-02  1.931e+00   0.011  0.99116
## as.factor(STATE)35  3.512e+00  3.657e+00   0.960  0.33687
## as.factor(STATE)36 -1.743e+00  1.649e+00  -1.057  0.29066
## as.factor(STATE)37 -1.961e+00  1.350e+00  -1.453  0.14615
## as.factor(STATE)38 -2.952e+00  6.753e+00  -0.437  0.66200
## as.factor(STATE)39  3.856e-01  1.125e+00   0.343  0.73172
## as.factor(STATE)40 -1.345e+01  1.386e+03  -0.010  0.99226
## as.factor(STATE)41  2.608e+00  3.685e+00   0.708  0.47917
## as.factor(STATE)42  1.964e+00  1.694e+00   1.160  0.24611
## as.factor(STATE)44 -6.169e-01  1.876e+00  -0.329  0.74225
## as.factor(STATE)45 -1.793e+00  1.496e+00  -1.198  0.23084
## as.factor(STATE)46  7.262e-01  1.219e+00   0.596  0.55130
## as.factor(STATE)47 -3.205e+00  2.499e+00  -1.283  0.19960
## as.factor(STATE)48 -2.409e+00  1.503e+00  -1.603  0.10900
## as.factor(STATE)49  6.197e+00  3.808e+00   1.628  0.10362
## as.factor(STATE)50         NA         NA      NA       NA
## as.factor(STATE)51 -1.960e+00  1.303e+00  -1.504  0.13260
## as.factor(STATE)53  3.002e+00  3.603e+00   0.833  0.40474
## as.factor(STATE)54 -1.541e+01  1.647e+03  -0.009  0.99254
## as.factor(STATE)55         NA         NA      NA       NA
## as.factor(STATE)56         NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2696.29  on 3110  degrees of freedom
## Residual deviance:  338.59  on 3057  degrees of freedom
## AIC: 446.59
##
## Number of Fisher Scoring iterations: 19
```

```
tp4 <- predict(m4, type="response") > 0.5
pR2(m4)
```

```
##           llh       llhNull            G2      McFadden          r2ML
##  -169.2951526 -1348.1457842  2357.7012631     0.8744237     0.5313315
##          r2CU
##     0.9166230
```

```
table(elections_df$dems_more,tp4)
```

```
##    tp4
##     FALSE TRUE
##   0  2593   32
##   1    37  449
```

```
table(elections_df$dems_more,tp4) %>% diag() %>% sum()
```

```
## [1] 3042
```

```
3042/(32+37+3042)
```

```
## [1] 0.9778206
```

Notes:

- STATES 31, 29, 17, and 12 are statistically significant predictors. They represent Nebraska, Missouri, Illinois, and Florida respectively. Florida is surprising because it is a swing state. But perhaps these states all have an exceptionally large number of counties with very polarized beliefs.

- Predicted outcome correctly 97.8% of the time
- Logistic regression had a relatively strong fit; pseudo R-squared (McFaddens) has a value of .874

**Explain statistical significance (#2.4)**

Statistical significance is the likelihood that the difference between a given variation and your model is not due to random chance. The null hypothesis for any model is that there is no relationship between the (dependent) variable you are predicting and the (independent) variables you are using to predict it. When something is statistically significant, it means the null hypothesis was rejected and the likelihood of there actually BEING a relationship is high. You can estimate how high that likelihood is using confidence level. For example, when an estimated coefficient is statistically significant at 5% confidence level, it means there is only a 5% chance that the coefficient is correctly your dependent variable due to random chance. In other words, there is a 95% chance the coefficient is predicting the dependent variable to the magnitude of that coefficient.

**Statistical significance in my model (#2.5, #2.6)**

- In my model, the coefficients that are significant at a 0.1% confidence level are pct_dem_2012, POPESTIMATE2016, and RDEATH2016
- The coefficients that are significant at a 1% confidence level are STATE(17) and STATE(31)
- The coefficients that are significant at the 5% confidence level are REGION(3), STATE(29), and STATE(12)
- pct_dem_2012 has a coefficient of 66.72. This means that one additional percentage of votes for Democrat in 2012 increases the odds ratio of the Democrat getting the most votes in 2016 / the Democrat not getting the most votes in 2016 by 66.72. It makes sense that this variable has the greatest effect because a similar percentage of the county that voted for the democratic candidate in 2012 is likely to vote for the democratic candidate again in 2016.

- POPESTIMATE2016 has a coefficient of 4.226e-06. This means that as population increases by 1, the odds ratio of the Democrat getting the most votes in 2016 / the Democrat not getting the most votes in 2016 increased by 4.226e-06.
- RDEATH2016 has a coefficient of -7.188e-01. This means that as the death rate increases by 1, the odds ratio of the Democrat getting the most votes in 2016 / the Democrat not getting the most votes in 2016 decreases by 0.7188.

---

## 3. Simulate effect of additional random coefficients

### Choose logistic model from above to add random coefficients

```
m3 <- glm(dems_more~pct_dem_2012+as.factor(REGION)+POPESTIMATE2016+NPOPCHG_2016+RDEATH2016, data=electi
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(m3)
```

```
##
## Call:
## glm(formula = dems_more ~ pct_dem_2012 + as.factor(REGION) +
##     POPESTIMATE2016 + NPOPCHG_2016 + RDEATH2016, family = binomial(link = "logit"),
##     data = elections_df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.2699  -0.0257  -0.0020  -0.0001   2.4675
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -2.511e+01  1.891e+00 -13.276  < 2e-16 ***
## pct_dem_2012        5.422e+01  3.870e+00  14.010  < 2e-16 ***
## as.factor(REGION)2 -9.189e-01  3.797e-01  -2.420   0.0155 *
## as.factor(REGION)3  2.080e+00  4.089e-01   5.087 3.64e-07 ***
## as.factor(REGION)4  2.331e+00  4.817e-01   4.839 1.31e-06 ***
## POPESTIMATE2016     4.640e-06  8.654e-07   5.362 8.22e-08 ***
## NPOPCHG_2016       -1.636e-04  5.390e-05  -3.035   0.0024 **
## RDEATH2016         -5.737e-01  6.709e-02  -8.552  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2696.29  on 3110  degrees of freedom
## Residual deviance:  415.87  on 3103  degrees of freedom
## AIC: 431.87
##
## Number of Fisher Scoring iterations: 10
```

```
tp3 <- predict(m3, type="response") > 0.5
pR2(m3)
```

```
##           llh       llhNull            G2      McFadden          r2ML
##  -207.9368145 -1348.1457842  2280.4179395     0.8457609     0.5195431
##          r2CU
##     0.8962862
```

```
table(elections_df$dems_more,tp3)
```

```
##    tp3
##     FALSE TRUE
##   0  2589   36
##   1    49  437
```

```
table(elections_df$dems_more,tp3) %>% diag() %>% sum()
```

```
## [1] 3026
```

```
3026/(36+49+3026)
```

```
## [1] 0.9726776
```

### Generate a bunch of coefficients from random vector

```
  # 3111 observations in elections_df
coef_val <- c()
  # takes ONE minute to run (couldn't figure out how to use sapply with this model)
for (val in c(1:1001)){
  set.seed(val)
  rd_nos <- runif(3111,min=0,max=10000)
  elections_df$rd_nos <- rd_nos
  m_rep <- glm(dems_more~pct_dem_2012+as.factor(REGION)+POPESTIMATE2016+NPOPCHG_2016+RDEATH2016+rd_nos,
  coef_val[val] <- coef(m_rep)["rd_nos"]
  elections_df$rd_nos <- NULL
}
```

### Sample mean and std dev (#3.2, #3.3)

```
mean(coef_val)
```

```
## [1] 2.786558e-08
```

```
sd(coef_val)
```

```
## [1] 4.652352e-05
```

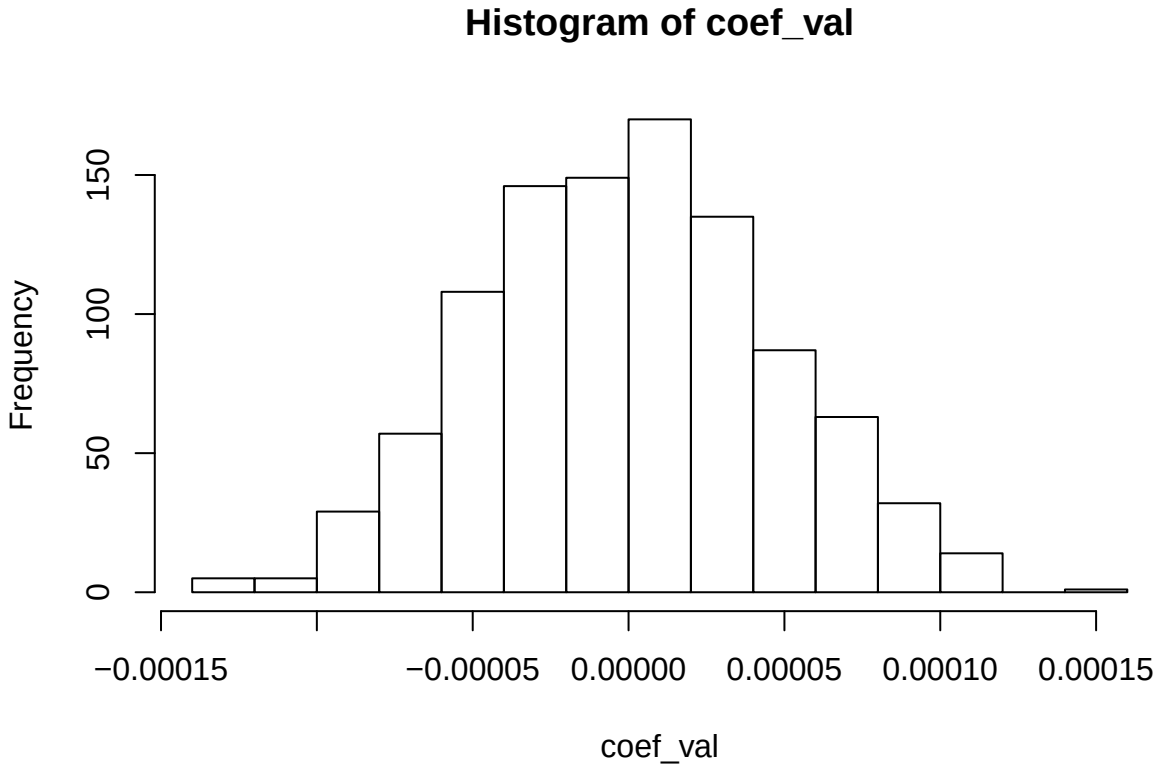
```
quantile(coef_val,c(0.025,0.975))
```

```
##          2.5%         97.5%
## -8.730061e-05  9.290061e-05
```

- The sample mean is 2.786558e-08
- The standard deviation is 4.652352e-05
- The 95% confidence interval is -8.730061e-05 to 9.290061e-05

### Plot distribution (#3.4, 3.5)

```
hist(coef_val)
```




```
mean(coef_val) - (1.96*sd(coef_val))
```

```
## [1] -9.115822e-05
```

```
mean(coef_val) + (1.96*sd(coef_val))
```

```
## [1] 9.121396e-05
```

- The distribution looks approximately normal
- The theoretical 95% confidence intervals would be mean - 1.96 standard deviations to mean + 1.96 standard deviations. This is -9.115822e-05 to 9.121396e-05
- The actual 95% confidence interval described previously is not far from this, so the distribution is probably close to normal