Technical Report

Shisham Adhikari, Maggie Slein, Grayson White

Abstract

As American politics has become increasingly polarized over the last several decades, predicting the likelihood of presidential victories has become more difficult. With just a few states' electoral votes deciding the figurehead of the executive branch, two out of the four last elections have been decided based on the electoral college and not on the popular vote. Given this, many previous models used to predict election outcomes have failed to predict the true winner. We aim to address in two ways: First, by using the `survey` package to create the "classic" logistic regression models with weights, and then by using `tidymodels` and `stacks` to layer multiple models and model types to better predict party affiliation using several predictors from the General Social Survey (GSS) dataset from 2000-2016.

Introduction

An overview of the topic and relevant background information, a discussion of existing theories and models, a description of how your investigation differs from prior ones, and a precise statement of your research question.

In the last two decades, the polarization of American politics has yielded unexpected victories by minority political groups, both in the 2000 and 2016 election. The Republican party represents less Americans by volume than both the Democratic party and the Independent parties, yet has continued to secure political power as a result of the electoral college. The two unprecedented victories in 2000 and 2016 by candidates who lost the popular vote but won the presidency as a result of the electoral college have been attributed to highly variable party affiliation in what have been dubbed "swing states". In these states, the split of Democratic to Republican votes is fairly even, which makes winning their popular votes crucial for ultimately securing their electoral votes, in the electoral college's "winner takes all" framework. These states have been a source of contention in 2 of the 4 elections.

In 2000, the race between Governor George W. Bush vs. Vice President Al Gore for the presidency came down to one state: Florida. Gore had secured the popular vote, but he needed to win Florida to win the necessary electoral votes to secure the presidency. Early on during election night, Bush was called as the winner with margin of 100,000 votes. However, as votes from highly democratic districts poured in, the margin between Bush and Gore narrowed to just 2,000 votes. The vote counts in Florida were so close, the law demanded a recount. However, the legally mandated recount was to be performed by machine, not by hand. At the time, some counties used punch ballots and there was concern over the anomalies present in ballots cast with such a narrow margin between the two candidates. Gore pushed to have hand-counted recounts in particular counties via litigation. With the saga of who would ultimately become president-elect enduring for over a month after election day, the Supreme Court Case, Bush v. Gore ended the recount on December 12th, 2000. The verdict in that case was essentially that Gore did not have the grounds to request anything beyond a machine-automated recount of the votes. Thus, the electoral votes from the state of Florida ultimately decided that outcome of the 2000 election: President-elect Governor George W. Bush.

For the fifth time in US history, the 2016 race between businessman Donald J. Trump and former Secretary of State Hilary Clinton was also decided by the electoral college and not the popular vote. Less contentious than the 2000 election, the results of the 2016 election have been attributed to a lack of concern for winning electoral votes in key swing states. As these highly influential and dynamic "swing states" shift with each passing election, a better of understanding and prediction of their political party tendencies becomes increasing important. To better understand how external factors drive political party affiliation and ultimately predict political party affiliation, we are interested using techniques that combine many model types to provide more accurate conclusions.

Previous models have utilized one statistical learning method: Principal Component Analysis (Newman and Sheth 1985), LASSO (Kristensen et al. 2017), and multiple linear regression (Ben-Bassat et al. 2009) when attempting understand how social attributes influence party affiliation or election outcome. While these models have demonstrated that social behavior, like a single Facebook like can predict party affiliation (Kristensen et al 2017) and identity, like Arab voter turnout and political affinity is Israeli elections (Ben-Bassat et al. 2009), they do not explain the trend towards polarization in last decade of American elections, domestically or globally. Thus, to better understand this phenomenon, we asked the question: How can we use a combination of models to predict political party affiliation using small sample sizes? Our group wishes to understand the external factors related to political party affiilation, and to understand how well we can predict political party affiliation by using a combination of the modeling techniques learned in class. Model stacking is an ensemble method that combines a variety of model types to optimize model predictions by training the model with a variety of model types.

Methods

The GSS dataset

The general social survey (GSS) is a massive survey conducted of people within the United States since 1972. The GSS aims to get a representative sample of people in the United States and to understand information about them and how they feel about social and political issues. We have chosen some key variables collected from this survey, along with participants from 2000 or more recent, in order for us to attempt to classify political affiliation of participants. Our subset of the GSS dataset contains 5,800 rows, 16 columns, and 0 NA's.

Filtering

Most of this filtering was done for the `infer` package `gss` dataset and can be attributed to authors of that package. We have included more rows and columns than that package, however, much initial tidying and subsetting can be attributed to them (Bray et al. 2020). Below is the code adapted from the `infer` package to attain our dataset, `gss_subset`:

```r
load("gss/gss_orig.rda")
gss_subset <- gss_orig %>%
  filter(!stringr::str_detect(sample, "blk oversamp")) %>% # this is for weighting
  select(year, age, sex, college = degree, partyid, hompop, hours = hrs1, income,
         class, finrela, wrkgovt, marital, educ, race, incom16, weight = wtssall) %>%
  mutate_if(is.factor, ~ fct_collapse(., NULL = c("IAP", "NA", "iap", "na"))) %>%
  mutate(
    age = age %>%
      fct_recode("89" = "89 or older",
                 NULL = "DK") %>%
      as.character() %>%
      as.numeric(),
    hompop = hompop %>%
      fct_collapse(NULL = c("DK")) %>%
      as.character() %>%
      as.numeric(),
    hours = hours %>%
      fct_recode("89" = "89+ hrs",
                 NULL = "DK") %>%
      as.character() %>%
      as.numeric(),
    weight = weight %>%
      as.character() %>%
      as.numeric(),
    partyid = fct_collapse(
      partyid,
      dem = c("strong democrat", "not str democrat"),
      rep = c("strong republican", "not str republican"),
      ind = c("ind,near dem", "independent", "ind,near rep"),
      other = "other party"
    ),
    income = factor(income, ordered = TRUE),
    college = fct_collapse(
      college,
      degree = c("junior college", "bachelor", "graduate"),
      "no degree"  = c("lt high school", "high school"),
      NULL = "dk"
    )
  ) %>%
  filter(year >= 2000) %>%
  filter(partyid %in% c("dem", "rep")) %>%
  drop_na()
```

Given our goal to understand which factors influence party affiliation in the US, we selected `year` (year of the election), `age` (age of time of survey), `college` (degree or no degree), `partyid` (democrat or republican), `hompop` (number of people in the respondent's household), `hours` (number of hours worked in the last week), `income` (total family income, categorical), `class` (socioeconomic class as described by respondent), `finrela` (respondent's opinion on family's income level), `wrkgovt` (whether or not the respondent works for the government), `marital` (respondent's martial status), `educ` (highest year of school completed), `race` (race of respondent), `income16` (respondent's family income at the age of 16), and `weight` (survey weight).

We made some choices while filtering the dataset which will effect the final results of our models. First of all, we have filtered all observations which do not state that their political affiliation was either democrat or republican. We are most interested in answering the question of whether or not we can classify between these parties rather than considering much smaller third parties. Also, we have filtered all observations with any NA's. We chose to do this for ease of analysis and because many of the models we use will not consider a row that includes NA's in any of the columns being used for the model.

Exploratory Data Analysis

Before we dig too deeply in to the dataset, it is important to understand its structure:

```
# Number of rows
nrow(gss_subset)
```

```
## [1] 5800
```

```r
# Number of columns
ncol(gss_subset)
```

```
## [1] 16
```

```r
# Response variable summary
summary(gss_subset$partyid)
```

```
##   dem   ind   rep other    DK
##  3316     0  2484     0     0
```

```r
# Data Structure
str(gss_subset)
```

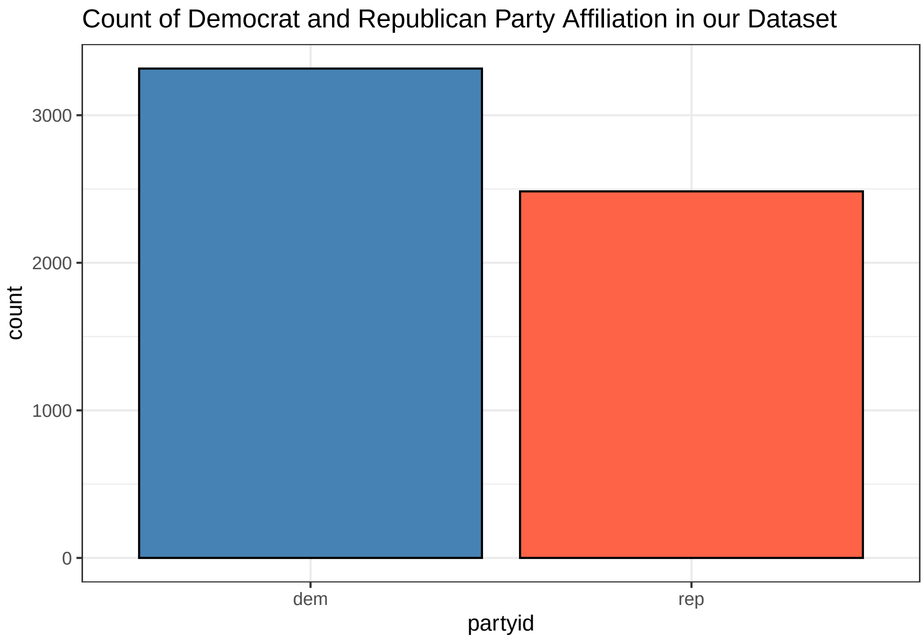
```
## tibble [5,800 x 16] (S3: tbl_df/tbl/data.frame)
##  $ year   : num [1:5800] 2002 2002 2002 2002 2002 ...
##   ..- attr(*, "label")= chr "gss year for this respondent "
##   ..- attr(*, "format.stata")= chr "%8.0g"
##  $ age    : num [1:5800] 25 43 46 71 37 23 33 57 42 63 ...
##  $ sex    : Factor w/ 2 levels "male","female": 2 1 1 2 1 1 1 1 2 1 ...
##  $ college: Factor w/ 2 levels "no degree","degree": 1 2 1 1 1 1 2 2 2 2 ...
##  $ partyid: Factor w/ 5 levels "dem","ind","rep",..: 3 3 3 3 3 1 1 1 1 1 ...
##  $ hompop : num [1:5800] 1 1 2 1 1 3 4 2 1 1 ...
##  $ hours  : num [1:5800] 40 72 40 24 50 60 70 40 65 44 ...
##  $ income : Ord.factor w/ 12 levels "lt $1000"<"$1000 to 2999"<..: 12 12 12 11 12 12 12 12 12 12 ...
##  $ class  : Factor w/ 6 levels "lower class",..: 3 3 3 2 3 2 2 3 2 3 ...
##  $ finrela: Factor w/ 6 levels "far below average",..: 3 4 4 3 3 3 3 3 4 4 ...
##  $ wrkgovt: Factor w/ 3 levels "government","private",..: 2 2 2 2 2 2 2 2 2 1 ...
##  $ marital: Factor w/ 5 levels "married","widowed",..: 3 1 3 3 5 4 1 1 5 5 ...
##  $ educ   : Factor w/ 22 levels "0","1","2","3",..: 15 17 15 13 16 13 17 17 17 18 ...
##  $ race   : Factor w/ 3 levels "white","black",..: 1 1 1 1 1 2 3 1 1 1 ...
##  $ incom16: Factor w/ 7 levels "far below average",..: 3 4 4 3 2 3 3 4 2 4 ...
##  $ weight : num [1:5800] 0.558 0.558 1.116 0.558 0.558 ...
```

```r
# Glimpse of dataset
gss_subset %>%
  select(-weight) %>%
  rename(home = hompop, party = partyid) %>%
  head() %>%
  knitr::kable()
```

year	age	sex	college	party	home	hours	income	class	finrela	wrkgovt	marital	educ	race	incom16
2002	25	female	no degree	rep	1	40	$25000 or more	middle class	average	private	divorced	14	white	average
2002	43	male	degree	rep	1	72	$25000 or more	middle class	above average	private	married	16	white	above average
2002	46	male	no degree	rep	2	40	$25000 or more	middle class	above average	private	divorced	14	white	above average

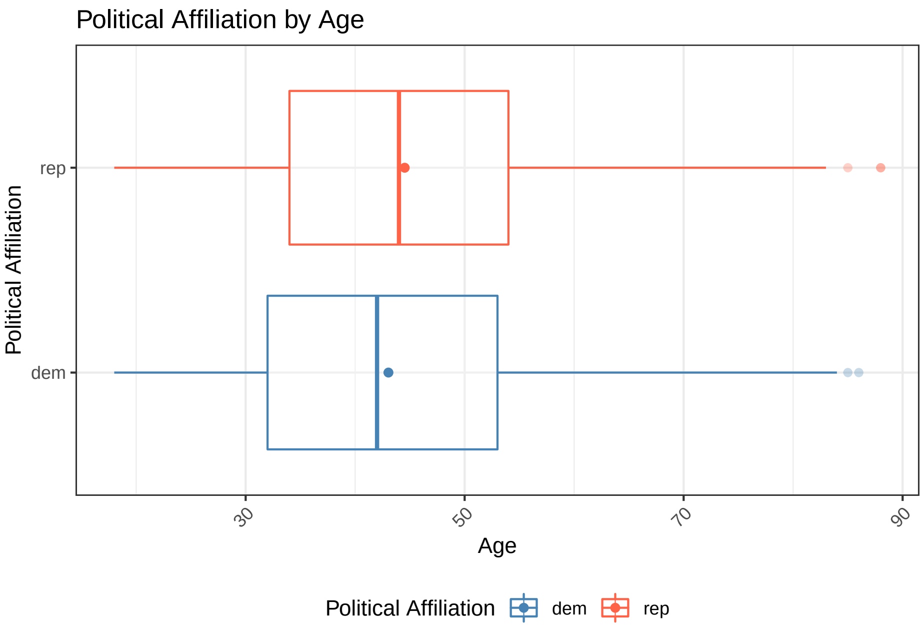
year	age	sex	college	party	home	hours	income	class	finrela	wrkgovt	marital	educ	race	incom16
2002	71	female	no degree	rep	1	24	$20000 - 24999	working class	average	private	divorced	12	white	average
2002	37	male	no degree	rep	1	50	$25000 or more	middle class	average	private	never married	15	white	below average
2002	23	male	no degree	dem	3	60	$25000 or more	working class	average	private	separated	12	black	average

As we first explore the dataset, we can look at the distribution of democrats and republications in our dataset in counts:

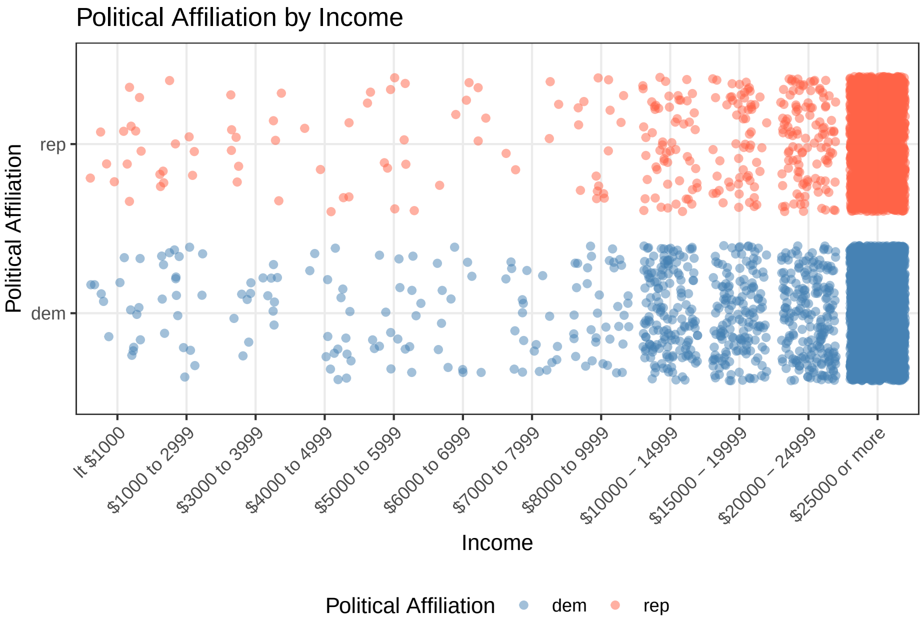


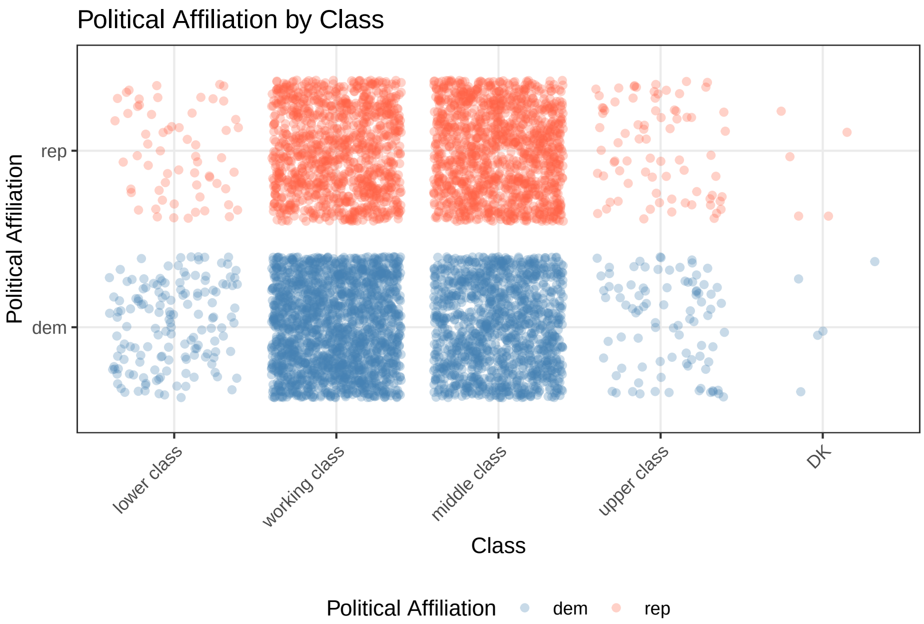
There appears to me more democrats than republicans represented in this dataset, which could be because democrats are more likely to participate in this survey, or it could be that the way we selected our data systemically oversampled democrats. Notably from this, it is the case that our the weights associated with our sample of the GSS dataset would not be the same as the weights that the GSS uses for the dataset, so the `weight` variable should be ignored entirely.

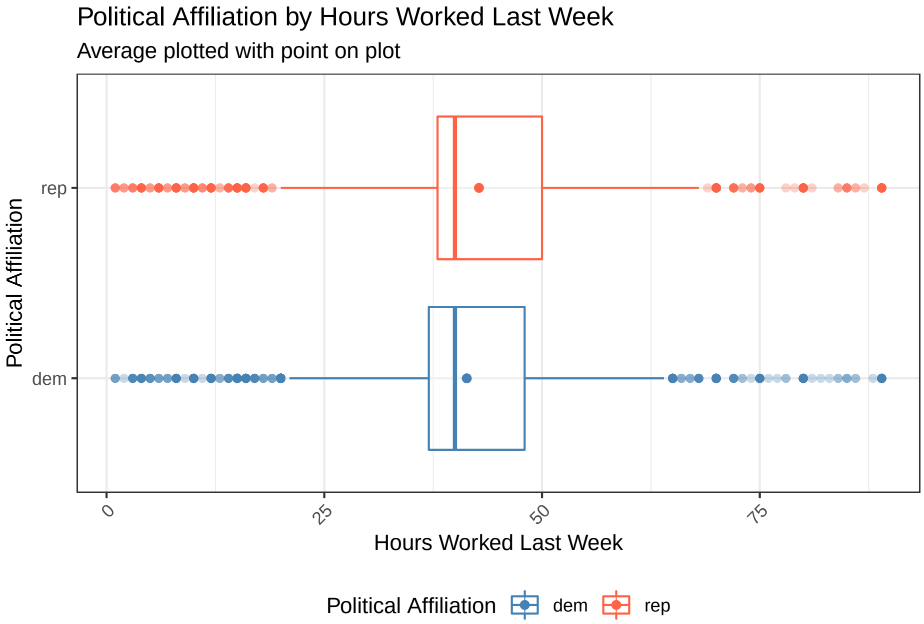
Now, we can examine some of our predictor variables with our response, `partyid`, to see the relationships there are between variables. First, we see in this side-by-side boxplot with the means plotted on top that republicans tend to be older on average:



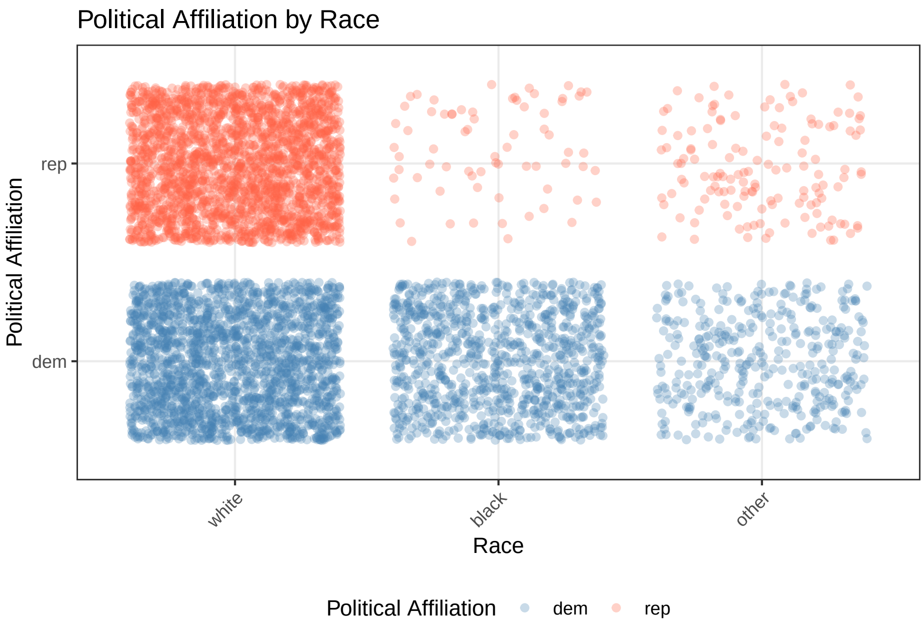
Next, it is interesting to consider economic status across political affiliations. By comparing political affiliation to income, class, and hours worked in the last week we can see small relationships between political affiliation and economic status:

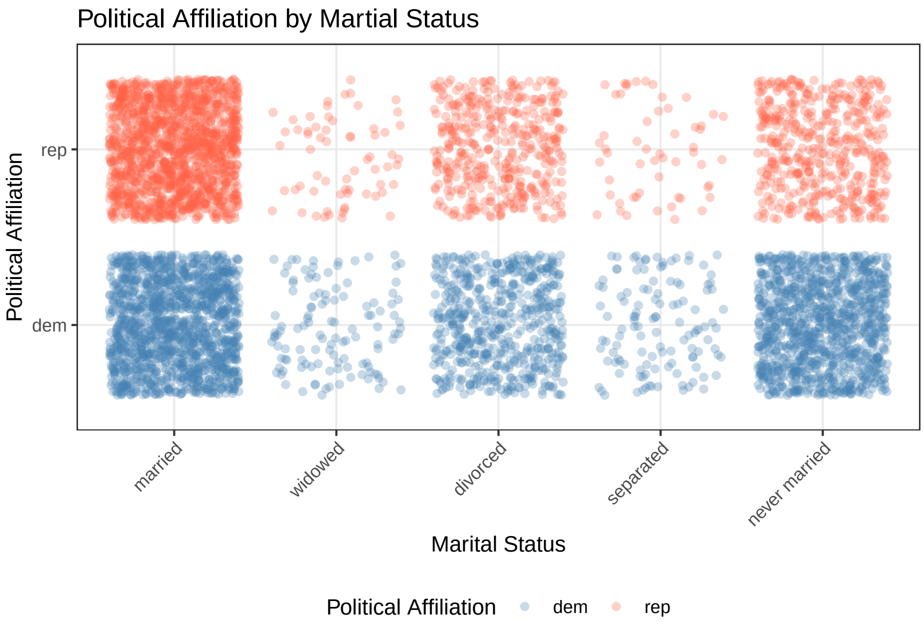


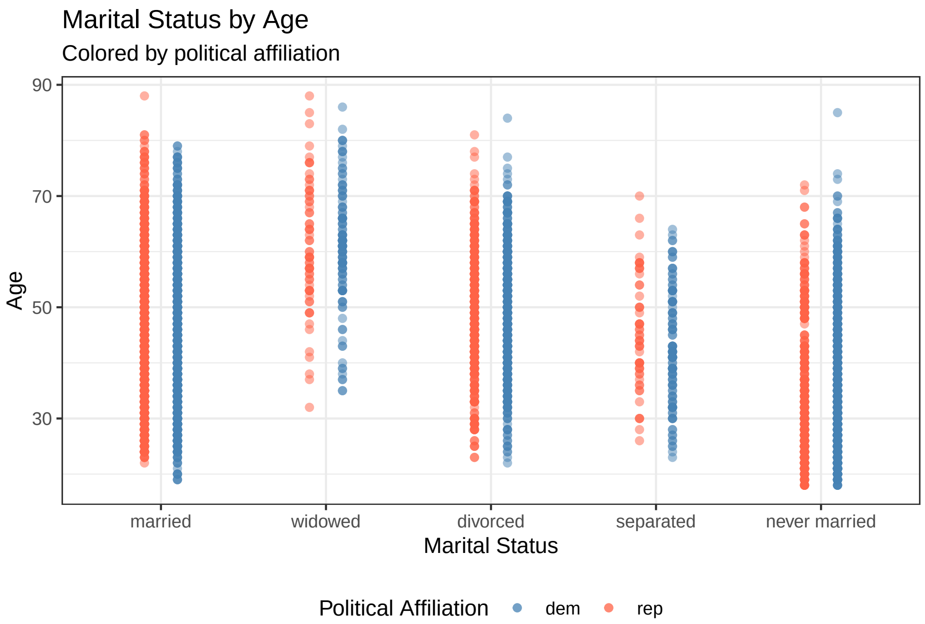


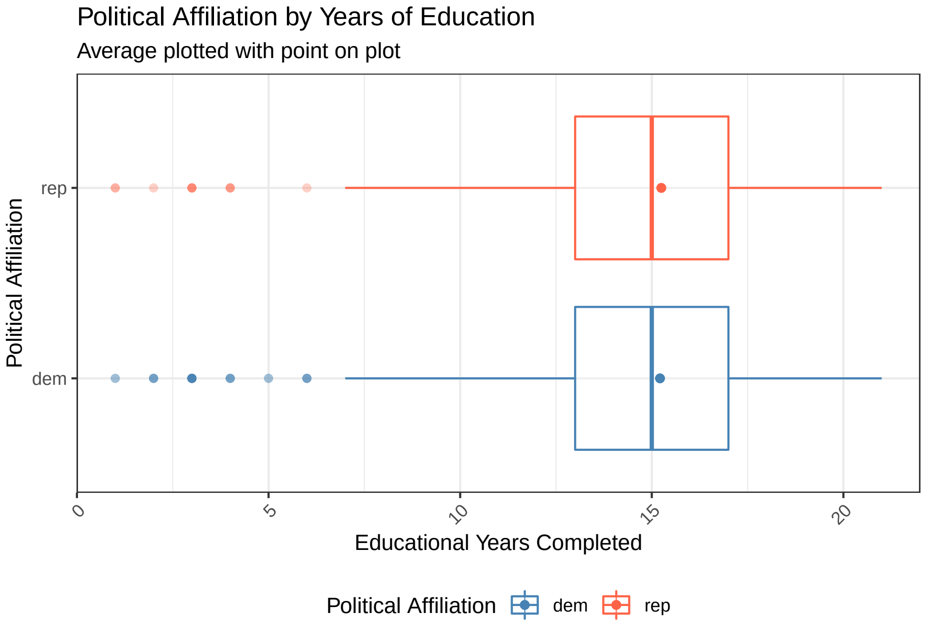


It is also relevant to look at other variables such as race, marital status, and education as factors related to political party affiliation. Most notably, there is a much larger proportion of white republicans than democrats. We can see this in the first plot in the following plots:


Political Affiliation by Race
Political Affiliation
rep
dem
white
black
other
Race
Political Affiliation
dem
rep


Political Affiliation by Martial Status
Political Affiliation
rep
dem
married
widowed
divorced
separated
never married
Marital Status
Political Affiliation
dem
rep





After completing these exploratory analyses, it is clear that while there are some weak relationships within many variables, we will likely need all of these variables to make models which have good predictive power. None of the predictors appear to have an extremely strong relationship with political party affiliation, and so we will need to use many of them for our models to perform well.

We also examined two classification model methods for accuracy in predicting partyid based on some of our 16 predictors. Linear disriminany analysis (LDA) appeare to perform better job correctly classifying Democrats than Republicans based on these 6 predictors, as there was an equal amount of Republicans incorrectly predicted to those correctly predicted. Our logistic regression model with all 16 predictors also appears to better classify Democrats than Republicans, but not by much, with an overall training error rate of about 18%. These results suggest that the current classification models we have used throughout this course may not be successful in predicting partyid with high accuracy on their own. We hope to leverage these methods through model stacking in our Methods and Results section.

```
#taking a look at how LDA could perform on our dataset with just a couple of variables
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```
set.seed(2020)
mlda <- lda(partyid ~ race + age + year + hompop + income + wrkgovt, data = gss_subset)
```

```
## Warning in lda.default(x, grouping, ...): groups ind other DK are empty
```

```
mlda_pred <- predict(mlda)
conf_mlda <- table(mlda_pred$class,gss_subset$partyid)
conf_mlda
```

```
##
##          dem  ind  rep other   DK
##   dem   1651    0  464     0    0
##   ind      0    0    0     0    0
##   rep   1665    0 2020     0    0
##   other    0    0    0     0    0
##   DK       0    0    0     0    0
```

```
#taking a look at how logistic regression could perform on our dataset with just a couple of variables
simple_logreg<-glm(partyid ~ race + age + year + hompop + income + wrkgovt, data = gss_subset, family=
summary(simple_logreg)
```

```
##
## Call:
## glm(formula = partyid ~ race + age + year + hompop + income +
##     wrkgovt, family = "binomial", data = gss_subset)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6308  -1.1936  -0.3684   1.1004   2.6449
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     19.645497  11.633733   1.689   0.0913 .
## raceblack       -2.726179   0.128901 -21.149  < 2e-16 ***
## raceother       -1.149020   0.107074 -10.731  < 2e-16 ***
## age              0.005456   0.002224   2.453   0.0142 *
## year            -0.009976   0.005788  -1.724   0.0848 .
## hompop           0.089863   0.022250   4.039 5.37e-05 ***
## income.L       -0.764534   0.344701  -2.218   0.0266 *
## income.Q        0.592449   0.389611   1.521   0.1284
## income.C        0.294515   0.366015   0.805   0.4210
## income^4        0.103443   0.390148   0.265   0.7909
## income^5       -0.363236   0.366427  -0.991   0.3215
## income^6        0.259640   0.394253   0.659   0.5102
## income^7       -0.088836   0.383146  -0.232   0.8166
## income^8        0.025585   0.398678   0.064   0.9488
## income^9       -0.752271   0.421740  -1.784   0.0745 .
## income^10       0.567971   0.408693   1.390   0.1646
## income^11       0.627012   0.481088   1.303   0.1925
## wrkgovtprivate  0.081194   0.073943   1.098   0.2722
## wrkgovtDK      -0.122282   0.280065  -0.437   0.6624
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7920.7  on 5799  degrees of freedom
## Residual deviance: 6939.2  on 5781  degrees of freedom
## AIC: 6977.2
##
## Number of Fisher Scoring iterations: 5
```

```r
full_logreg<-glm(partyid ~ ., data = gss_subset, family= "binomial")
summary(full_logreg)
```

```
##
## Call:
## glm(formula = partyid ~ ., family = "binomial", data = gss_subset)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8081  -1.0644  -0.3376   1.0318   2.7773
##
## Coefficients:
##                            Estimate Std. Error z value Pr(>|z|)
## (Intercept)               9.934e+00  1.208e+01   0.823  0.41076
## year                     -4.422e-03  5.979e-03  -0.740  0.45952
## age                      -7.696e-04  2.665e-03  -0.289  0.77276
## sexfemale                -3.593e-01  6.220e-02  -5.777 7.62e-09 ***
## collegedegree             9.412e-02  1.191e-01   0.790  0.42930
## hompop                   -4.604e-03  2.723e-02  -0.169  0.86576
## hours                     2.927e-03  2.150e-03   1.361  0.17347
## income.L                 -7.907e-01  3.574e-01  -2.212  0.02695 *
## income.Q                  4.297e-01  4.032e-01   1.066  0.28649
## income.C                  7.981e-02  3.755e-01   0.213  0.83167
## income^4                  6.986e-02  3.996e-01   0.175  0.86120
## income^5                 -4.738e-01  3.753e-01  -1.263  0.20669
## income^6                  2.337e-01  4.002e-01   0.584  0.55930
## income^7                 -1.640e-01  3.927e-01  -0.418  0.67619
## income^8                  4.602e-02  4.068e-01   0.113  0.90993
## income^9                 -8.015e-01  4.272e-01  -1.876  0.06064 .
## income^10                 5.947e-01  4.161e-01   1.429  0.15291
## income^11                7.482e-01  4.893e-01   1.529  0.12623
## classworking class       -1.070e-01  1.901e-01  -0.563  0.57349
## classmiddle class         1.601e-01  1.975e-01   0.811  0.41762
## classupper class         -1.058e-01  2.650e-01  -0.399  0.68984
## classDK                   2.152e-01  7.134e-01   0.302  0.76295
## finrelabelow average     -6.919e-02  1.816e-01  -0.381  0.70317
## finrelaaverage           -2.840e-02  1.818e-01  -0.156  0.87582
## finrelaabove average      1.231e-01  1.899e-01   0.648  0.51679
## finrelafar above average  1.670e-01  2.650e-01   0.630  0.52869
## finrelaDK                 6.747e-02  5.946e-01   0.113  0.90965
## wrkgovtprivate           -3.290e-02  7.750e-02  -0.425  0.67115
## wrkgovtDK                -2.093e-01  2.892e-01  -0.724  0.46937
## maritalwidowed           -3.838e-01  1.827e-01  -2.101  0.03563 *
## maritaldivorced          -2.530e-01  8.915e-02  -2.838  0.00454 **
## maritalseparated         -3.928e-01  1.896e-01  -2.072  0.03826 *
## maritalnever married     -7.027e-01  8.675e-02  -8.100 5.50e-16 ***
## educ1                    -2.062e+00  1.625e+00  -1.269  0.20445
## educ2                    -1.405e+00  1.313e+00  -1.070  0.28460
## educ3                    -6.373e-01  1.426e+00  -0.447  0.65488
## educ4                    -1.326e+01  2.235e+02  -0.059  0.95268
## educ5                    -1.386e+00  1.610e+00  -0.861  0.38939
## educ6                    -1.733e+00  1.234e+00  -1.404  0.16042
## educ7                    -1.908e+00  1.415e+00  -1.349  0.17744
## educ8                    -5.753e-01  1.192e+00  -0.483  0.62942
## educ9                    -1.161e+00  1.181e+00  -0.982  0.32588
## educ10                   -5.666e-01  1.173e+00  -0.483  0.62911
## educ11                   -6.585e-01  1.164e+00  -0.566  0.57151
## educ12                   -6.432e-01  1.151e+00  -0.559  0.57636
## educ13                   -4.646e-01  1.154e+00  -0.402  0.68732
## educ14                   -7.047e-01  1.154e+00  -0.611  0.54125
## educ15                   -7.206e-01  1.157e+00  -0.623  0.53359
## educ16                   -7.899e-01  1.157e+00  -0.682  0.49497
## educ17                   -1.173e+00  1.164e+00  -1.008  0.31363
## educ18                   -1.231e+00  1.162e+00  -1.060  0.28912
## educ19                   -1.532e+00  1.171e+00  -1.307  0.19110
## educ20                   -1.647e+00  1.166e+00  -1.413  0.15779
## raceblack                -2.621e+00  1.313e-01 -19.957  < 2e-16 ***
## raceother                -1.070e+00  1.110e-01  -9.637  < 2e-16 ***
## incom16below average      1.976e-01  1.251e-01   1.581  0.11398
## incom16average            2.729e-01  1.204e-01   2.267  0.02337 *
## incom16above average      2.685e-01  1.324e-01   2.028  0.04255 *
## incom16far above average  1.607e-01  2.191e-01   0.733  0.46344
## weight                    1.421e-01  6.028e-02   2.357  0.01843 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7920.7  on 5799  degrees of freedom
## Residual deviance: 6701.7  on 5740  degrees of freedom
## AIC: 6821.7
##
## Number of Fisher Scoring iterations: 11
```

```r
probs<-predict(full_logreg, gss_subset, type = "response")
preds<-ifelse(probs >=.5, 1, 0)
conf_log <- table(preds, gss_subset$partyid)
conf_log
```

```
##
## preds  dem  ind  rep other   DK
##     0 2255    0  808     0    0
##     1 1061    0 1676     0    0
```

```r
n <- length(gss_subset$partyid)
false_pos <- conf_log[1,2]
false_neg <- conf_log[2,1]
error <- 1/n *(false_pos + false_neg)
error
```

```
## [1] 0.182931
```

Results

Discussion

Code Appendix

References

Bray, Andrew, Chester Ismay, Evgeni Chasnovski, Ben Baumer, and Mine Cetinkaya-Rundel. 2020. *Infer: Tidy Statistical Inference*. https://CRAN.R-project.org/package=infer.