*note economic variables at state and national level were winsorized (capped at 5th and 95th percentiles)

*this is because covid created levels of these variables that were historically never seen (such as high unemployment rates)

*capping at previous historical high levels reduces outlier influence and overfitting

Forecasting Presidential Elections

Code

Perebibowei Azazi

21 May, 2022

Set up our file and download the needed packages

Hide

```
#set working directory to load file
setwd("C:/Users/coold/Desktop/geecodes")


#load dataset
fundamental.data<-read.csv("fundamental_data_election.csv",stringsAsFactors=F,header
=T)
```

#note data is at level of state x election year, which is what we need #it has 561 rows which is 51 (states plus DC) x 11 elections since 1980 including 2020 #variables in data set

Code

*note economic variables at state and national level were winsorized (capped at 5th and 95th percentiles)

*this is because covid created levels of these variables that were historically never seen (such as high unemployment rates)

*capping at previous historical high levels reduces outlier influence and overfitting

```r
#explore dataset
dim(fundamental.data); colnames(fundamental.data); head(fundamental.data)
```

```
## [1] 561  36
```

```
##  [1] "election_date"             "election_year"
##  [3] "democratic_vote_share_adj" "republican_vote_share_adj"
##  [5] "state"                     "state_name"
##  [7] "region"                    "electoral_votes"
##  [9] "PVI"                       "state_unemployment"
## [11] "state_house_price"         "state_med_income"
## [13] "state_personal_income"     "nat_gdp"
## [15] "nat_nonfarm_payroll"       "nat_ind_production"
## [17] "nat_personal_exp"          "nat_personal_inc"
## [19] "nat_price_index"           "stock_market"
## [21] "age_under_18"              "age_18_to_40"
## [23] "age_40_to_65"              "age_65_plus"
## [25] "gender_male"               "gender_female"
## [27] "race_white"                "race_black"
## [29] "educ_hs_or_less"           "educ_less_than_4_yrs"
## [31] "educ_ba_or_post_grad"      "nonurban"
## [33] "urban"                     "density"
## [35] "net_approval"              "incumbent"
```

```
##   election_date election_year democratic_vote_share_adj
## 1    1980-11-04          1980                 0.3270081
## 2    1984-11-06          1984                 0.3094409
## 3    1988-11-08          1988                 0.3783668
## 4    1992-11-03          1992                 0.4342574
## 5    1996-11-05          1996                 0.3957150
## 6    2000-11-07          2000                 0.3206305
##   republican_vote_share_adj state state_name region electoral_votes        PVI
## 1                 0.6729919    AK     Alaska   West               3 -0.1985722
## 2                 0.6905591    AK     Alaska   West               3 -0.2446591
## 3                 0.6216332    AK     Alaska   West               3 -0.2082752
## 4                 0.5657426    AK     Alaska   West               3 -0.1733584
## 5                 0.6042850    AK     Alaska   West               3 -0.1917490
## 6                 0.6793695    AK     Alaska   West               3 -0.2776207
##   state_unemployment state_house_price state_med_income state_personal_income
## 1                9.6       -0.02285655               NA            0.11258024
## 2                9.5        0.04419252               NA            0.05524476
## 3                8.2        0.15857913     -0.003911774            0.04613674
## 4                8.4        0.03406087      0.029301684            0.04065257
## 5                7.5        0.05039039      0.100617258            0.04010450
## 6                6.4        0.02109267      0.028231769            0.07388679
##       nat_gdp nat_nonfarm_payroll nat_ind_production nat_personal_exp
## 1 0.018649251         0.001921102        -0.01353780       0.08695548
## 2 0.008208628         0.044702041         0.05470637       0.03402432
## 3 0.013324958         0.031391479         0.02748743       0.04170927
## 4 0.010431647         0.009012337         0.03548821       0.02671888
## 5 0.010383820         0.023575206         0.05754997       0.02441489
## 6 0.006228896         0.016019499         0.02125280       0.02527845
##   nat_personal_inc nat_price_index stock_market age_under_18 age_18_to_40
## 1       0.01742324      0.10485133  0.479768786    0.3337604    0.4434330
## 2       0.06024577      0.04154303 -0.002724796    0.3261277    0.4599665
## 3       0.03716257      0.04246101 -0.042194093    0.3353670    0.4047260
## 4       0.03026867      0.03120464  0.153225806    0.3187151    0.3690193
## 5       0.03929598      0.03253090  0.328930818    0.3107793    0.3715097
## 6       0.04835777      0.03444181  0.107893388    0.3438298    0.3361665
##   age_40_to_65 age_65_plus gender_male gender_female race_white race_black
## 1    0.1906253  0.03218136   0.5244403     0.4755597  0.7816126 0.02895366
## 2    0.1854212  0.02848461   0.5304949     0.4695051  0.7598859 0.03885189
## 3    0.2177040  0.04220304   0.5085276     0.4914724  0.7435135 0.04036799
## 4    0.2605948  0.05167083   0.5079881     0.4920119  0.7433641 0.05436533
## 5    0.2777853  0.03992571   0.5182073     0.4817927  0.7973348 0.05019038
## 6    0.2705758  0.04942803   0.5066827     0.4933173  0.7472945 0.04158266
##   educ_hs_or_less educ_less_than_4_yrs educ_ba_or_post_grad  nonurban     urban
## 1       0.4616605            0.2359848           0.04672580 1.0000000 0.0000000
## 2       0.4439632            0.2288924           0.05806418 1.0000000 0.0000000
## 3       0.4345118            0.2280362           0.06653076 0.5462505 0.4537495
## 4       0.4195310            0.2474633           0.07484090 0.4999388 0.5000612
## 5       0.3288643            0.2629508           0.15960407 0.4975909 0.5024091
```

```
## 6       0.3204781           0.2381668           0.15231834 0.5181572 0.4818428
##     density net_approval incumbent
## 1 0.6466675    -18.13502         1
## 2 0.7332402    -21.82420        -1
## 3 0.8810425    -17.84319         0
## 4 0.8470510     21.79688        -1
## 5 1.0040763     19.63401         1
## 6 1.0967949     20.19180         0
```

Hide

```
#date to partition train and test data sets
test.election.date<-2020
train<-fundamental.data[fundamental.data$election_year<test.election.date,]
test<-fundamental.data[fundamental.data$election_year==test.election.date,]
```

Hide

```
#train mixed effects model; specify fixed effects predictors and random intercepts f
or error simulations
#fixed effects specified like a regular linear regression in R as in lm function
#random effects specified using a different notation; random intercepts denoted by
 (1|group_variable)
#for national error use election_year, regional error use region, & state error is r
esidual (since model is at state level)
fundamental.model<-lmer(republican_vote_share_adj~PVI+state_unemployment+nat_gdp+rac
e_white+
                          (1|election_year)+(1|region),
                        data=train)
```

Hide

```
#extract random effects variances from model fit
vote.share.var<-as.data.frame(VarCorr(fundamental.model))
vote.share.var
```

```
##            grp        var1 var2         vcov      sdcor
## 1 election_year (Intercept) <NA> 0.0017108971 0.04136299
## 2        region (Intercept) <NA> 0.0001570505 0.01253198
## 3      Residual        <NA> <NA> 0.0012618683 0.03552279
```

Hide

```
national.error.var<-matrix(vote.share.var$vcov[vote.share.var$grp=="election_year"])
regional.error.var<-matrix(vote.share.var$vcov[vote.share.var$grp=="region"])
state.error.var<-matrix(vote.share.var$vcov[vote.share.var$grp=="Residual"])
```

Hide

```
#set constants for simulation
B<-1000 #number of times to simulate election
n.years<-length(unique(train$election_year))
n.states<-length(unique(train$state))
n.regions<-length(unique(train$region))
df<-n.years-1 #set df for student t simulations as number of elections minus 1
state.region.mapping<-unique(fundamental.data[,c("state","region")])
electoral.votes<-unique(fundamental.data[,c("state","electoral_votes")])
```

Hide

```
#simulate election B times

#repeat each row of test dataset B times to create expanded data across all simulati
ons
#note here each columns is therefore one simulation (rows are states + DC)
test.sim<-test[rep(1:nrow(test),times=B),]
dim(test.sim)
```

```
## [1] 51000    36
```

Hide

```
#create matrix of test set predictions; note flag for re.form=NA to set random effec
ts to zero for fixed effects prediction
fund.pred<-predict(fundamental.model,newdata=test.sim,re.form=NA)
pred.rep.share<-matrix(fund.pred,ncol=B)
dim(pred.rep.share); pred.rep.share[1:5,1:5]
```

```
## [1]   51 1000
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 0.5372330 0.5372330 0.5372330 0.5372330 0.5372330
## [2,] 0.5981711 0.5981711 0.5981711 0.5981711 0.5981711
## [3,] 0.6014883 0.6014883 0.6014883 0.6014883 0.6014883
## [4,] 0.4996734 0.4996734 0.4996734 0.4996734 0.4996734
## [5,] 0.3141839 0.3141839 0.3141839 0.3141839 0.3141839
```

Hide

```
#simulate random errors at national level
national.error<-rep(rmvt(n=B,sigma=national.error.var,df=df),each=n.states)
national.error<-matrix(national.error,ncol=B)
dim(national.error); national.error[1:5,1:5]
```

```
## [1]   51 1000
```

```
##             [,1]       [,2]        [,3]       [,4]       [,5]
## [1,] -0.04747549 0.03905997 0.004522059 0.03948534 0.01179659
## [2,] -0.04747549 0.03905997 0.004522059 0.03948534 0.01179659
## [3,] -0.04747549 0.03905997 0.004522059 0.03948534 0.01179659
## [4,] -0.04747549 0.03905997 0.004522059 0.03948534 0.01179659
## [5,] -0.04747549 0.03905997 0.004522059 0.03948534 0.01179659
```

Hide

```
#simulate random errors at regional level
#draw each region B times
region.error<-rmvt(n=n.regions*B,sigma=regional.error.var)
region.error<-matrix(region.error,ncol=B)
dim(region.error); region.error[,1:5]
```

```
## [1]    4 1000
```

```
##              [,1]        [,2]         [,3]       [,4]        [,5]
## [1,] -0.006986805  0.07214395  0.007851699 0.02505114 -0.02017292
## [2,]  0.011147960 -0.02858174  0.004441297 0.01653318  0.02043338
## [3,]  0.016583524 -0.07271940 -0.012936375 0.01710879 -0.03293066
## [4,]  0.272587575  0.01003481  0.037868202 0.04496404 -0.02318820
```

Hide

```
#replicate regional simulations so that states in same region get same regional erro
r
#this creates correlation among states within a region
head(state.region.mapping)
```

```
##    state region
## 1     AK   West
## 12    AL  South
## 23    AR  South
## 34    AZ   West
## 45    CA   West
## 56    CO   West
```

Hide

```
region.state.index<-as.numeric(factor(state.region.mapping$region))
region.error<-region.error[region.state.index,]
dim(region.error); data.frame(state.region.mapping,region.error)[1:5,1:5]
```

```
## [1]   51 1000
```

```
##    state region         X1          X2          X3
## 1     AK   West 0.27258758  0.01003481  0.03786820
## 12    AL  South 0.01658352 -0.07271940 -0.01293638
## 23    AR  South 0.01658352 -0.07271940 -0.01293638
## 34    AZ   West 0.27258758  0.01003481  0.03786820
## 45    CA   West 0.27258758  0.01003481  0.03786820
```

Hide

```
#simulate random errors at state level
#draw each state B times
state.error<-rmvt(n=n.states*B,sigma=state.error.var,df=df)
state.error<-matrix(state.error,ncol=B)
dim(state.error); state.error[1:5,1:5]
```

```
## [1]   51 1000
```

```
##             [,1]         [,2]        [,3]         [,4]        [,5]
## [1,] -0.04444723  0.008499783 -0.03092532  0.021363178  0.03576373
## [2,] -0.01123826 -0.009019765 -0.02456859  0.003665310 -0.01432047
## [3,] -0.02365247  0.018027452 -0.02117929 -0.028365470 -0.01551072
## [4,]  0.02876351  0.034922990 -0.05265242 -0.007307963 -0.01682597
## [5,] -0.04549260 -0.040233949 -0.01339289  0.040846123 -0.03918309
```

Hide

```
#calculate vote shares by simulation
#add three simulated errors to predicted republican vote shares
pred.rep.share<-pred.rep.share+national.error+region.error+state.error
```

Hide

```
#calculate win probabilities for republicans by state across simulations
rep.state.win.prob<-apply(pred.rep.share,1,function(x) sum(x>0.5)/length(x))
rep.state.win.prob<-data.frame("state"=electoral.votes$state,"rep_win_prob"=rep.stat
e.win.prob)
rep.state.win.prob
```

```
##    state rep_win_prob
## 1     AK        0.710
## 2     AL        0.899
## 3     AR        0.906
## 4     AZ        0.503
## 5     CA        0.025
## 6     CO        0.226
## 7     CT        0.084
## 8     DC        0.009
## 9     DE        0.090
## 10    FL        0.339
## 11    GA        0.441
## 12    HI        0.020
## 13    IA        0.551
## 14    ID        0.957
## 15    IL        0.066
## 16    IN        0.806
## 17    KS        0.870
## 18    KY        0.928
## 19    LA        0.768
## 20    MA        0.029
## 21    MD        0.035
## 22    ME        0.231
## 23    MI        0.270
## 24    MN        0.280
## 25    MO        0.787
## 26    MS        0.685
## 27    MT        0.851
## 28    NC        0.397
## 29    ND        0.957
## 30    NE        0.914
## 31    NH        0.337
## 32    NJ        0.063
## 33    NM        0.132
## 34    NV        0.207
## 35    NY        0.034
## 36    OH        0.457
## 37    OK        0.954
## 38    OR        0.129
## 39    PA        0.282
## 40    RI        0.055
## 41    SC        0.662
## 42    SD        0.930
## 43    TN        0.884
## 44    TX        0.621
## 45    UT        0.942
## 46    VA        0.201
## 47    VT        0.032
```

```
## 48    WA        0.073
## 49    WI        0.351
## 50    WV        0.961
## 51    WY        0.979
```

Hide

```
#calculate electoral votes that republicans win by state in each election
rep.EV<-apply(pred.rep.share,2,function(x) sum(electoral.votes$electoral_votes[x>0.5
]))
length(rep.EV)
```

```
## [1] 1000
```

Hide

```
#calculate probability that each party wins the election
winner<-ifelse(rep.EV>=270,"R","D")
prop.table(table(winner))
```

```
## winner
##     D     R
## 0.755 0.245
```