final project

2025-05-18

We began by using a random forest on 13 specific variables (protest_attend, religion, religiondevout, socioeconclass, incomeenough, race, education, age, leftrightscale, democracy_score, trust_score, economic_score, freedom_score) to determine the most important variables to predicting whether someone would attend a protest. We decided to choose these variables and filter out the others because they were the original list of predictors that we used to create the protest_attend column/variable [AMUNA double check]. We decided on using random forest because it was a supervised model and it was one that we were most familiar with. We also thought that ranger would work well with our large dataset that contains both categorical and numerical values.

RANDOM FOREST on the 13 chosen variables

```
library(ranger)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v readr     2.1.5
## v ggplot2   3.5.1     v stringr   1.5.1
## v lubridate 1.9.3     v tibble    3.2.1
## v purrr     1.0.2     v tidyr     1.3.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
set.seed(1)

protestData <- read.csv("stats0218_finaldata.csv")
```

```
protestData <- protestData |>
  mutate(across(c(protest_attend, religion, religiondevout, socioeconclass, incomeenough, race, educatio

protestDataClean13vars <- protestData |>
                select(protest_attend, religion, religiondevout, socioeconclass,
                       incomeenough, race, education, age, leftrightscale,
                       democracy_score, trust_score, economic_score, freedom_score) |>
  na.omit() |>
  mutate(across(c(protest_attend, religion, religiondevout, socioeconclass, incomeenough, race, educatio
#looking at data without NAs just to see which variables are most important


rf_NAs <- ranger(protest_attend ~ . - protest_label -protest_engagement_score1,
             data = protestDataClean13vars,
             importance = "impurity")
```

```
## Warning in terms.formula(f, data = data): 'varlist' has changed (from nvar=13)
## to new 15 after EncodeVars() -- should no longer happen!
```

```
rf_NAs
```

```
## Ranger result
##
## Call:
##  ranger(protest_attend ~ . - protest_label - protest_engagement_score1,      data = protestDataClean
##
## Type:                             Classification
## Number of trees:                  500
## Sample size:                      443
## Number of independent variables:  12
## Mtry:                             3
## Target node size:                 1
## Variable importance mode:         impurity
## Splitrule:                        gini
## OOB prediction error:             34.99 %
```

```
rf_NAs$confusion.matrix
```

```
##     predicted
## true   0   1
##    0 147  75
##    1  80 141
```

```
sort(rf_NAs$variable.importance)
```

```
##        religion  religiondevout    incomeenough       education            race
##        4.111215        7.884361        7.986362        9.120391        9.463214
##  socioeconclass  leftrightscale democracy_score   freedom_score             age
##       10.649797       15.915917       26.375429       29.302636       33.053659
##  economic_score     trust_score
##       33.129869       33.242460
```

```r
#prediction error: 35% , 65% accurate
#predicting 0s (68.1% accurate), predicting 1s (69% accurate)

table(protestData$protest_attend)
```

```
##
##   0   1
## 600 598
```

```r
#about same amount of 0s/1s
```

When we filter out all NAs, we find that the most important variables in order of most to least important are: trust_score, age, economic_score,freedom_score, democracy_score,leftrightscale, socioeconclass, race, education, religiondevout, incomeenough, religion.

Checking if there are NAs in the top 5 predictors from protestData

```r
colSums(is.na(protestData |>
                select(protest_attend, trust_score, age, economic_score,freedom_score, democracy_score)
```

```
##  protest_attend     trust_score             age  economic_score   freedom_score
##               2               0               0               0               2
## democracy_score
##               0
```

The variables protest_attend and freedom_score have NAs so we have to impute:

```r
set.seed(1)
library(mice)
```

```
##
## Attaching package: 'mice'

## The following object is masked from 'package:stats':
##
##     filter

## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```r
imputedData <- mice(protestData |> select(freedom_score, protest_attend),
                          method = "rf",
                          seed = 1)
```

```
##
##  iter imp variable
##   1   1  freedom_score  protest_attend
##   1   2  freedom_score  protest_attend
##   1   3  freedom_score  protest_attend
```

```
##   1   4  freedom_score  protest_attend
##   1   5  freedom_score  protest_attend
##   2   1  freedom_score  protest_attend
##   2   2  freedom_score  protest_attend
##   2   3  freedom_score  protest_attend
##   2   4  freedom_score  protest_attend
##   2   5  freedom_score  protest_attend
##   3   1  freedom_score  protest_attend
##   3   2  freedom_score  protest_attend
##   3   3  freedom_score  protest_attend
##   3   4  freedom_score  protest_attend
##   3   5  freedom_score  protest_attend
##   4   1  freedom_score  protest_attend
##   4   2  freedom_score  protest_attend
##   4   3  freedom_score  protest_attend
##   4   4  freedom_score  protest_attend
##   4   5  freedom_score  protest_attend
##   5   1  freedom_score  protest_attend
##   5   2  freedom_score  protest_attend
##   5   3  freedom_score  protest_attend
##   5   4  freedom_score  protest_attend
##   5   5  freedom_score  protest_attend
```

```
completeData <- complete(imputedData)

#fill in NAs with complete data
protestData$freedom_score <- completeData$freedom_score
protestData$protest_attend <- completeData$protest_attend


#Check to see if there are still NAs
protestDataClean13vars <- protestData |>
                select(protest_attend, religion, religiondevout, socioeconclass,
                       incomeenough, race, education, age, leftrightscale,
                       democracy_score, trust_score, economic_score, freedom_score) |>
  mutate(across(c(protest_attend, religion, religiondevout, socioeconclass, incomeenough, race, educati

colSums(is.na(protestDataClean13vars |>
                select(protest_attend, freedom_score, trust_score, economic_score, democracy_score, age
```

```
##  protest_attend   freedom_score     trust_score  economic_score democracy_score
##               0               0               0               0               0
##             age
##               0
```

Now that there are no more NAs in the original dataset, we will run random forest using the top 5 predictor variables

```
set.seed(1)
rf <- ranger(protest_attend ~ trust_score+age+economic_score+freedom_score+democracy_score,
             data = protestDataClean13vars,
             importance = "impurity")

rf
```

```
## Ranger result
##
## Call:
##  ranger(protest_attend ~ trust_score + age + economic_score +      freedom_score + democracy_score, d
##
## Type:                             Classification
## Number of trees:                  500
## Sample size:                      1200
## Number of independent variables:  5
## Mtry:                             2
## Target node size:                 1
## Variable importance mode:         impurity
## Splitrule:                        gini
## OOB prediction error:             34.92 %
```

```
rf$confusion.matrix
```

```
##     predicted
## true   0   1
##    0 379 223
##    1 196 402
```

We see that the prediction error is about the same when we ran random forest on the data that omitted the NAs. The model is slightly better at predicting 1s (63% right) vs predicting 0s (67% right).

Instead of imputing data using mice, what if we did something more naive such as imputing data using mean/median?

```
set.seed(1)
#reload original protestData
protestData <- read.csv("cleandata.csv")
protestData <- protestData |>
  mutate(across(c(protest_attend, religion, religiondevout, socioeconclass, incomeenough, race, educati

#select columns with NAs, find median, and update the columns in protestData
protestDataNoNA <- protestData |>
  select( freedom_score)

medianFreedomScore <- median(protestDataNoNA$freedom_score)

protestDataMedianAsNAs <- protestDataClean13vars |>
    mutate(freedom_score = case_when(is.na(freedom_score) ~ medianFreedomScore, TRUE ~ freedom_score))
  filter(!is.na(protest_attend)) #gets rid of 6 missing protest_attend NA values instead of calculating

rfMedians <- ranger(protest_attend ~ trust_score+age+economic_score+freedom_score+democracy_score,
             data = protestDataMedianAsNAs,
             importance = "impurity")

rfMedians
```

```
## Ranger result
##
## Call:
```

```
##  ranger(protest_attend ~ trust_score + age + economic_score +      freedom_score + democracy_score, d
##
## Type:                             Classification
## Number of trees:                  500
## Sample size:                      1200
## Number of independent variables:  5
## Mtry:                             2
## Target node size:                 1
## Variable importance mode:         impurity
## Splitrule:                        gini
## OOB prediction error:             34.92 %
```

```
rfMedians$confusion.matrix
```

```
##     predicted
## true   0   1
##    0 379 223
##    1 196 402
```

When imputing data with median values, the predictions are slightly worse (40.1% prediction error) when compared to models using imputed data.

Training/testing RF with the imputed data from mice/complete since its slightly better

```
#reloading protestData/protestDataClean13vars with imputed NA data
protestData <- read.csv("stats0218_finaldata.csv")

protestData <- protestData |>
  mutate(across(c(protest_attend, religion, religiondevout, socioeconclass, incomeenough, race, educatio

protestData$freedom_score <- completeData$freedom_score
protestData$protest_attend <- completeData$protest_attend

protestDataClean13vars <- protestData |>
                select(protest_attend, religion, religiondevout, socioeconclass,
                       incomeenough, race, education, age, leftrightscale,
                       democracy_score, trust_score, economic_score, freedom_score) |>
  mutate(across(c(protest_attend, religion, religiondevout, socioeconclass, incomeenough, race, educatio

colSums(is.na(protestData |>
                select(protest_attend, freedom_score, trust_score, economic_score, democracy_score, age)
```

```
##  protest_attend   freedom_score     trust_score  economic_score democracy_score
##               0               0               0               0               0
##             age
##               0
```

```
#training/testing
set.seed(2)
split <- sample(1:nrow(protestDataClean13vars), 0.5*nrow(protestDataClean13vars))
train <- protestDataClean13vars[split,]
test <- protestDataClean13vars[-split, ]
```

```
rfTrain <- ranger(protest_attend ~ trust_score+age+economic_score+freedom_score+democracy_score,
             data = train,
             importance = "impurity")

rfTrainPreds <- predict(rfTrain, data = test)
table(rfTrainPreds$predictions, test$protest_attend)
```

```
##
##       0   1
##   0 178  94
##   1 128 200
```

From the table we see that: Predicting 0s: 65.4% accuracy Predicting 1s: 60.9% Overall accuracy: 63%

This tells us that the model has an overall accuracy of 63%, but is slightly better at predicting 0s (when someone does not attend the protest). An overall accuracy of 63% is better than average, but not by much. Instead of limiting our model to the 13 variables, our group decided to try using the entire dataset to see if other predictor variables might be better.

All current RF models aren't amazing. Decided to try using all variables from original data

```
protestData <- read.csv("stats0218_finaldata.csv")

#taking out NAs to see most important variables
protestDataAllVars <- protestData |>
  mutate(protest_attend = as.factor(protest_attend)) |>
  filter(!is.na(protest_attend)) |>
  na.omit()

rfAll <- ranger(protest_attend ~ . - protest_label - protest_engagement_score1,
             data = protestDataAllVars,
             importance = "impurity")

sort(rfAll$variable.importance)
```

```
##                      numinves                        idenpa
##                  0.0000000000                0.0000000000
##                       totcuot                         digit
##                  0.0000000000                0.0000000000
##                      P31INN_B                     P31INN_F
##                  0.0000000000                0.0000000000
##                      P31INN_6                         P47ST
##                  0.0000000000                0.0000000000
##                       P57ST_F                      P57ST_G
##                  0.0000000000                0.0000000000
##                            S9                    S13INN_A_C
##                  0.0000000000                0.0000000000
##                    S13INN_A_2                    S13INN_A_3
##                  0.0000000000                0.0000000000
##                        S14M_E                        S14M_G
##                  0.0000000000                0.0000000000
##                        S14M_H                        S14M_I
##                  0.0000000000                0.0000000000
```

```
##                S14M_J                S15_A
##          0.0000000000          0.0000000000
##                S15_B                  S16
##          0.0000000000          0.0000000000
##                S20_A                S20_C
##          0.0000000000          0.0000000000
##                S20_D                S20_F
##          0.0000000000          0.0000000000
##                S20_G                S20_H
##          0.0000000000          0.0000000000
##                S20_I                S20_J
##          0.0000000000          0.0000000000
##                S22_B               P33N_D
##          0.0000000000          0.0005777778
##              S13INN_A_1               P18N_H
##          0.0017647059          0.0020000000
##                P23ST_D               S14M_D
##          0.0026666667          0.0026666667
##                P57ST_H             S13INN_A_D
##          0.0030000000          0.0031704261
##               P31INN_5                S20_K
##          0.0034285714          0.0035555556
##                S14M_B               P57ST_A
##          0.0036666667          0.0037686275
##     marketeconomygood               P24ST_A
##          0.0040181818          0.0043111111
##                P24ST_C                 P19N
##          0.0043589744          0.0048000000
##               P38CSN_F          trustinpolice
##          0.0053333333          0.0057000000
##               P31INN_A                P36ST
##          0.0057142857          0.0064000000
##       trustinjudiciary                   S8
##          0.0064285714          0.0064444444
##                P57ST_B               P18N_D
##          0.0066666667          0.0068000000
##          governedbyfew               P18N_E
##          0.0071428571          0.0072000000
##                P25N_A             P37STCS_E
##          0.0072380952          0.0072666667
##               P38CSN_C               P56N_B
##          0.0074666667          0.0075753968
##                   S11               S14M_F
##          0.0080980392          0.0081696970
##                P25N_C               P57ST_E
##          0.0083804196          0.0084461538
##               P31INN_4                fampart
##          0.0084784689          0.0086363636
##       trustinchurch               P44ST_D
##          0.0089392713          0.0092222222
##                P53N_D             P31INN_3
##          0.0093405573          0.0094424242
##                 P48ST             P31INN_1
##          0.0095897436          0.0096666667
```

```
##                        reg                      P56N_C
##               0.0100586821                0.0101522914
##                     P53N_A                      P53N_C
##               0.0103333333                0.0104000000
##                  P38CSN_B                     perpart
##               0.0106848485                0.0109489621
##                      P46N                     P24ST_B
##               0.0109578947                0.0111372549
##             socioeconclass                      P59ST
##               0.0114927536                0.0115474747
##                     P62N_1          democracypreference
##               0.0121264069                0.0122469697
##           householdheademp                     totrech
##               0.0126000000                0.0126810967
##                  P33ST_B                    P37STCS_C
##               0.0126835165                0.0128253968
##                  P31INN_E                   P43STGBS_B
##               0.0129697320                0.0133739130
##                 education                    P30INN_E
##               0.0134133401                0.0134285714
##          trustintlcompanies                   S13INN_A_B
##               0.0135272727                0.0141030303
##                 S13INN_A_A                    P31INN_C
##               0.0144571429                0.0146424242
##               trustinbanks       corruptionlast2years
##               0.0148000000                0.0149500000
##           chanceswuoriginf                    P38CSN_G
##               0.0150303030                0.0153333333
##                         S3                      P63ST
##               0.0155076923                0.0158787879
##                  P34WVSNA                     P62ST_2
##               0.0163564103                0.0164042347
##            freedomofspeech                       codif
##               0.0165654528                0.0167658730
##              freedom_group                     P23ST_F
##               0.0169019079                0.0169563492
##               P37WVSSTCS_D                    P37CSN_F
##               0.0171731602                0.0172203175
##                  P30INN_F                          S6
##               0.0174666667                0.0174777778
##                     S14M_C                      P32INN
##               0.0175344029                0.0179380952
##                  P38CSN_A                      P33N_C
##               0.0182612368                0.0191399209
##                        S12                      S22_A
##               0.0192195804                0.0192301587
##                  P29STIN_B                     P23ST_A
##               0.0192663781                0.0197818182
##                  P31INN_D                         S19
##               0.0199314650                0.0200437922
##             trustinpeople                     P23ST_E
##               0.0201346032                0.0202500000
##                     P25N_B                     P13ST_I
##               0.0204498965                0.0205381903
```

```
##                       P30INN_B                        numcasa
##                    0.0205754386                   0.0206314204
##             equalityofmenwomenf                        totrevi
##                    0.0208432794                   0.0211273726
##            trustinprintedpress                         tamciud
##                    0.0216226238                   0.0220151961
##         trustinpoliticalparties                          S20_B
##                    0.0220588235                   0.0225332158
##         publicopinionexpression                        P18ST_I
##                    0.0230317460                   0.0231010101
##                          P58ST                  religiondevout
##                    0.0235533800                   0.0239311422
## countrypast12monthseconsituation                      REEDUC_2
##                    0.0240213564                   0.0244399711
##                       P53N_B                         P28STIN
##                    0.0244746108                   0.0245797908
##                            S4                          P18N_F
##                    0.0249119127                   0.0249658730
##                       P23ST_C              freedomchancetogetjob
##                    0.0251819005                   0.0253878915
##              trustinarmedforces                         totperd
##                    0.0254612277                   0.0255144928
##                       P30INN_D                justfairwealthdistf
##                    0.0255914297                   0.0257633766
##                    trustinradio                        P24ST_E
##                    0.0259560583                   0.0267936508
##                          reedad                        P24ST_D
##                    0.0269660710                   0.0269730769
##                       P57ST_D                        P42STGBS
##                    0.0274434703                   0.0277056581
##                       P26SDN_A                        P62N_14
##                    0.0295101859                   0.0301187653
##                         P18N_G                           S21A
##                    0.0308751338                   0.0310303030
##                          P61ST                  democracybest
##                    0.0317825108                   0.0322359527
##                        mesreal                   robotjobtake
##                    0.0323172653                   0.0336381788
##                           P52N                          P21ST
##                    0.0353293743                   0.0355735749
##          countryeconomicsituation          protectionagainstcrimef
##                    0.0362154220                   0.0362177489
##                       P14ST_A                           P51N
##                    0.0365528822                   0.0366169172
##                         P56N_D                       REEDUC_3
##                    0.0366404417                   0.0381769825
##                       P18STM_C                        P34WVSNB
##                    0.0382268477                   0.0382947372
##                          S20_E                socioecon_label
##                    0.0386143597                   0.0387938451
##                         P56N_A                          P35NB
##                    0.0399148815                   0.0399305116
##                       P26SDN_C                trustinsocmedia
##                    0.0401741924                   0.0407428986
```

```
##           freedomprivateproperty                        P62ST_5
##                     0.0418709716                   0.0420846280
##                        P26SDN_F                         codigo
##                     0.0424934188                   0.0433423687
##                  trustintelevision                        P23ST_B
##                     0.0434883550                   0.0440729247
##                   countryproblems                        P62N_13
##                     0.0448053985                   0.0448266224
##                          P62N_12                          S18_A
##                     0.0450380120                   0.0450770094
##                          P31INN_2                       P37WVSCS_G
##                     0.0452211980                   0.0457564305
##                              P55N                       P26SDN_E
##                     0.0461076479                   0.0482007361
##                              sexo                       P38CSN_D
##                     0.0484415508                   0.0492314233
##                economysatisfaction                        P57ST_C
##                     0.0495110863                   0.0521516872
##                          superven                  democracy_group
##                     0.0526344988                   0.0539822299
##                          supervvi                           P22ST
##                     0.0545620618                   0.0547530844
##                           P62N_9                  socialsecurityf
##                     0.0552662963                   0.0553739285
##                        freedomenv                            ini
##                     0.0560021632                   0.0560915920
##                              race              freedompoliticalpart
##                     0.0567993162                   0.0576482906
##                           P62ST_8                       P18STM_B
##                     0.0579807768                   0.0603376179
##                           P41ST_L                            P39N
##                     0.0605852184                   0.0618633630
##                           P27SDN                       codsuper
##                     0.0620038256                   0.0621948496
##                  incomedistfairness                         S14M_A
##                     0.0627719048                   0.0634438874
##                           P20STM                  trustincongress
##                     0.0644937238                   0.0645837503
##               democracysatisfaction                         P35NA
##                     0.0646397733                   0.0655414396
##                              S10                            S24
##                     0.0665674068                   0.0684324844
##                        P37WVSCS_H                         P33N_A
##                     0.0685298186                   0.0699270297
##                              S21B                       P26SDN_B
##                     0.0709574942                   0.0710974868
##                               wt                  freedomofreligion
##                     0.0720617835                   0.0751592487
##                  trustintradeunions                       P43STGBS_A
##                     0.0754735794                   0.0755767577
##                               fin                  countryprogress
##                     0.0756554881                   0.0779960287
##                              dura                        P62N_15
##                     0.0783821451                   0.0786935719
```

```
##                         P26SDN_D                         P62ST_10
##                      0.0794152015                      0.0808169386
##                         religion                          P62ST_3
##                      0.0809341104                      0.0823856203
## countrynext12monthseconsituation                          P41ST_B
##                      0.0841247271                      0.0877734289
##                         P30INN_C                         P29STIN_A
##                      0.0903348127                      0.0904329242
##                         P37CSN_B  familynext12monthseconsituation
##                      0.0916595194                      0.0921953632
##                interestinpolitics                           comdist
##                      0.0946613193                      0.0965814823
##                      incomeenough                             S18_B
##                      0.0979529049                      0.1056223932
##                           P62ST_4                           P62N_6
##                      0.1115703317                      0.1121519616
##                   democracy_score                     freedom_score
##                      0.1130598039                      0.1130951928
##                               S17                           P62ST_7
##                      0.1146647418                      0.1147704975
##                           P62N_17                          P62ST_11
##                      0.1177864278                      0.1224804145
##                            ciudad                          P37CSN_A
##                      0.1308662916                      0.1331721160
##         trustinnationalgovernment                    leftrightscale
##                      0.1342564068                      0.1362344158
##                  workforcommunity                  lifesatisfaction
##                      0.1367042208                      0.1379907079
##                 protestinsocmedia                               age
##                      0.1453468858                      0.1471552540
##                           diareal            presleadershipapproval
##                      0.1567998106                      0.1586766006
##                   opinionplatform                          P38CSN_E
##                      0.1603345342                      0.1631742518
##                           P44ST_C               trustinnatcompanies
##                      0.1643166108                      0.1647915502
##                                 X                       trust_group
##                      0.1678413981                      0.1764879987
##                           P62N_16                       protestagree
##                      0.2066839820                      0.2140897747
##                     nonauthdemonst                               P3N
##                      0.2217776523                      0.2757320301
##                          numentre              trustinelectoralinst
##                      0.2817905185                      0.2976324862
##                       trust_score                    economic_score
##                      0.3037161766                      0.3883560820
##                           P45ST_A                 talkoftenpolitics
##                      0.4307306536                      0.5194219696
##                     economic_group                       authdemonst
##                      0.6297522123                      0.6823904683
```

Top 10 most important variables in order of most to least important: high_engagement, authdemonst, protestinsocmedia, talkoftenpolitics, protestagree, workforcommunity, nonauthdemonst , interestinpolitics, protestagree

Imputing missing data #“‘{r} colSums(is.na(protestData |> select(protest_engagement_score1, high_engagement, authdemonst, protestinsocmedia, talkoftenpolitics, opinionplatform, workforcommunity, nonauthdemonst, interestinpolitics, protestagree, protest_attend)))

library(mice)

imputedDataAllVars <- mice(protestData |> select(protest_engagement_score1, high_engagement, authdemonst, protestinsocmedia, talkoftenpolitics, opinionplatform, workforcommunity, nonauthdemonst, interestinpolitics, protestagree, protest_attend), method = “rf”, seed = 1)

completeDataAllVars <- complete(imputedDataAllVars)

#fill in NAs with complete data protestData$protest_engagement_score1 <- -completeDataAllVars$protest_engagement_score
protestData$high_engagement <- -completeDataAllVars$high_engagement protestData$authdemonst <- -completeDataAllVars$authdemonst protestData$protestinsocmedia <- -completeDataAllVars$protestinsocmedia protestData$talkoftenpolitics <- -completeDataAllVars$talkoftenpolitics protestData$opinionplatform <- -completeDataAllVars$opinionplatform protestData$workforcommunity <- -completeDataAllVars$workforcommunity protestData$nonauthdemonst <- -completeDataAllVars$nonauthdemonst protestData$interestinpolitics <- -completeDataAllVars$interestinpolitics protestData$protestagree <- -completeDataAllVars$protestagree protestData$protest_attend <- -completeData$protest_attend

#Check to see if there are still NAs colSums(is.na(protestData |> select(protest_attend, protest_engagement_score1, high_engagement, authdemonst, protestinsocmedia, talkoftenpolitics, opinionplatform, workforcommunity, nonauthdemonst, interestinpolitics, protestagree)))

#“‘ No more NAs moving onto RF with top ten variables instead

#“‘{r} rfTopTen <- ranger(protest_attend ~ protest_engagement_score1 + high_engagement + authdemonst + protestinsocmedia + talkoftenpolitics + opinionplatform + workforcommunity + nonauthdemonst + interestinpolitics + protestagree , data = protestData, importance = “impurity”)

rfTopTen rfTopTen$confusion.matrix #“‘ Almost perfect prediction (0.03% error) Gets 1s almost perfectly (9599 out of 9600 correct) and 0s (9601 out of 9605 correct)