Assignment #1

David Caban

Bronsin Jabraeili

Scott Jue

Dr. Joe Wilck

Northwestern University

MSDS 411 – Unsupervised Learning Methods

April 01, 2023

Abstract

Surveyed data is often large in volume, has a great amount of variability; making it difficult to understand insightful patterns. This research paper focuses on the value dimensionality reduction provided on the Pew Research Center's political public opinion survey data. Two methods are used: principal component analysis (PCA) and exploratory factor analysis (EFA) in the R programming language. A thorough literature review will be conducted and presented, showing how other researchers have identified the value of dimensionality reduction provided when working with complex survey data. The common thread across researchers is that without techniques like PCA and EFA, the large volume of data is essentially useless. The methodology utilized consisted of reading in the data, performing exploratory data analysis, checking if there are missing data points, removing missing data, converting the data to be a matrix of binary indicators, performing both PCA and EFA in R, assessing the outputs of each technique, and reaching conclusions. After following the steps within the methodology section, the results reveal that EFA consisted of ~7 factors whereas PCA consisted of ~40 factors. EFA was clearly the better option to proceed with, being more intuitive to interpret 7 factors rather than 40 components. From there, various permutations were performed in R and the best one was utilizing 7 factors, rotation using varimax, and factoring method using principal factor (pa). This allowed for the final correlation matrix to be established which facilitated the analysis of any trends. Key, unique insights were generated across factors including: attitudes toward the current political environment, perceptions of discrimination, trust in the Federal Government, tax fairness, and attitudes toward religion. Overall, this study reveals the evolution from confusing, complex data to key strategic insights revealed following dimensionality reduction.

Keywords: *Principal Component Analysis, Factor Analysis, Unsupervised Learning, Dimensionality Reduction, Survey Data*

Introduction

The purpose of this assignment is to vividly demonstrate the utility of exploratory data analysis utilizing unsupervised learning methods such as PCA and EFA to aid in consulting political campaign strategies through data driven insights. However, these methods are not limited to garnering insights about the data through powerful visualizations, they also optimize working with the data by making it more interpretable at the lowest possible cost (dimensionality reduction) through the “components” and “factors” being recommended for selection. These points are especially paramount when working with massive, complex data sets such as the Pew Research Center survey data in this exercise (Cheng 2022). Pew Research Center is a non-partisan research organization that conducts regular surveys on a wide range of topics related to politics, social issues, and public opinion. The data set for this analysis includes 129 columns that are difficult to keep track of and 1,503 rows of records. When a topic comes to mind, one generally associates certain attributes with different categories. In this instance, when it comes to political affiliation, there may be known trends in regards to age, regions within the United States, education, and income brackets that typically align with different party values. However, there is no way to guarantee that the incoming assumptions are representative of the data unless the algorithms of interest in this experiment are run. The underlying architecture of the data, relativity amongst variables, and the efficiency of dimensionality reduction will be revealed in this study of PCA and EFA. As such, political survey analysis is an important tool for understanding the attitudes and opinions of the American public towards political issues.

Literature Review

Theiss-Morse et al. (2018) discusses the use of voter surveys and exit polls to examine various factors that influence voter behaviors in U.S. political elections. These factors include party affiliation, positions on current issues, sentiment about current candidates, and demographic characteristics. By using these factors to understand the current voter landscape, political campaigns can develop appropriate strategies to influence voters in their favor. However, survey data often has a high number of dimensions that can also suffer from multicollinearity. As a result, this creates several challenges for analyzing and modeling the data. Weng and Young (2017) have explored the effectiveness of several dimensionality reduction techniques such as PCA in the context of analyzing health insurance coverage survey data obtained from federal databases. Using a data set with reduced dimensions, Weng and Young (2017) were able to successfully create a model that identified regional differences for healthcare insurance coverage. As such, prior research shows that dimensionality reduction techniques can provide significant value when dealing with survey data in order to facilitate data modeling. Therefore, the analysis below will expand on research performed by Weng and Young, but in the context of political survey data provided by the Pew Research Center to identify the main differences between voter populations across the political spectrum.

Methodology

The approach taken is to first perform exploratory data analysis on the complex survey data consisting of ~194,000 cells within the data, convert everything into binary indicators, handling missing data, performing both PCA and EFA, comparing the two methods, and finally reaching conclusions about the data. The language used to perform these steps was R because of its plethora of versatile libraries and excellent visualization capabilities.

The pre-requisite for PCA and EFA is data preparation. There cannot be any missing values in traditional PCA and therefore it must be handled (Bailey 2018). There are several ways to impute the missing values but in this instance, the strategy implemented was to utilize the subset of complete records within the data (See Exhibit A in Appendix). After completing the data wrangling, the data was converted into a matrix consisting of binary indicators as PCA and EFA techniques can only be applied to numeric data.

The next step is to employ the various strategies to the refined data, analyze the outputs, and decide what is the best approach to implement. The first step was to create the correlation matrix, compute the eigenvalues which summarize the variability represented by each component, and visualize utilizing a scree plot (See Exhibit B in Appendix). The visualization reveals somewhere around 38-40 components is where the eigenvalues are no longer greater than or equal to one. The next approach was to visualize the differences in the numbers of components and factors when utilizing the different methods; more in depth detail of the outputs will be provided in the Results section subsequent to this section. In the Results section we will also uncover why EFA was favored in this analysis. The final methodology of the analysis was to compare using no rotation parameters with "varimax" rotation parameters as well as the number of components within the fa() function in the R code (See Exhibits E-G in Appendix).

Results

A comparison between PCA and EFA resulted in fewer factors being required for EFA (~7) than PCA components (~40). Therefore, EFA is the more efficient and less complex method. This is demonstrated in Exhibit C and makes the EFA results easier to interpret and explain to consulting clients. Seven factors with a varimax rotation and principal axis factoring were used to perform EFA because this was determined to be the best possible combination of parameters. Varimax was selected as it is a widely used rotation method and principal axis factoring was used, as opposed to other methods such as maximum likelihood, as it does not require distributional assumptions (Baglin 2014).

The seven factors measured respondent sentiment across a variety of categories. A review of the questions that were most heavily tied to each factor highlighted the topic or concept being measured. For example, the relevant questions related to PA2 measured respondent opinions of the pervasiveness of discrimination and included questions such as, “Please tell me how much discrimination there is against each of these groups in our society today. The factors are categorized in detail in Exhibit I.

As illustrated in Exhibit F, PA1 and PA2 are the factors that explain the highest amount of variance in the data (45%). The remaining factors each explain between 8% and 12% of the variance. In regard to fit, the root mean square of the residuals was 0.04, the RMSEA index was 0.058 with 90% confidence intervals of 0.057 and 0.059, indicating that the model fits the data reasonably well.

The EFA highlighted 15 separate questions that delineated the factors. It was noted that these questions could be used to create a regression tree that could approximate where on a political spectrum a respondent’s ideology might fall with just a handful of questions. This increases the utility of the dimension reduction provided by EFA even further. The regression tree is included in Exhibit H**.** Conservative ideologies lie closer to 1 and liberal ideologies lie closer to 5.

Conclusion

PCA and EFA methods were compared to determine the appropriate technique for this type of analysis on political survey data. EFA was shown to be the more appropriate method and was successfully used to perform dimensionality reduction of the Pew Research Center's political public opinion survey data from March 20, 2019. By being able to isolate questions that relate to specific topics and themes, latent variables can be uncovered that may have been difficult to see prior to EFA dimensionality reduction techniques. The information obtained through this process can then be used to better understand the key questions in the survey, build new and more efficient questionnaires, better understand the opinions that relate to one's political ideology, and develop more effective campaign strategies.

References

Bailey, S. (2018). *Principal Component Analysis with Missing Data*. Medium.

Retrieved April 10, 2023, from

https://medium.com/@seb231/principal-component-analysis-with-missing-data-9e28f440ce93

Baglin, J. (2014) *Improving Your Exploratory Factor Analysis for Ordinal Data: A Demonstration Using FACTOR.* Practical Assessment, Research, and Evaluation: Vol. 19 , Article 5.

DOI: https://doi.org/10.7275/dsep-4220

Cheng, C. (2021). *Principal Component Analysis (PCA) Explained Visually with Zero Math*. Towards Data Science.

Retrieved April 10, 2023, from

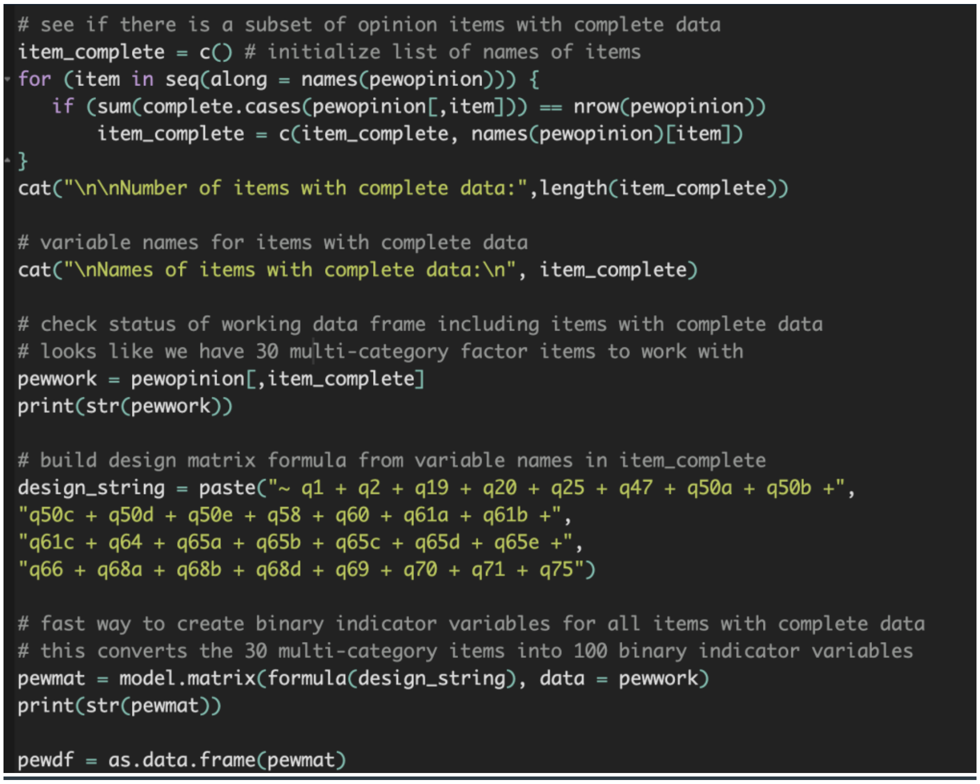
https://towardsdatascience.com/principal-component-analysis-pca-explained-visually-with-zero-math-1cbf392b9e7d

Theiss-Morse, Elizabeth, Michael W. Wagner, William H. Flanigan, and Nancy H. Zingale. 2018. *Political Behavior of the American Electorate*. Washington: CQ Press.

Weng, Jiaying, and Derek S. Young. 2017. “Some Dimension Reduction Strategies for the Analysis of Survey Data.” *Journal of Big Data* 4 (1). doi:10.1186/s40537-017-0103-6.

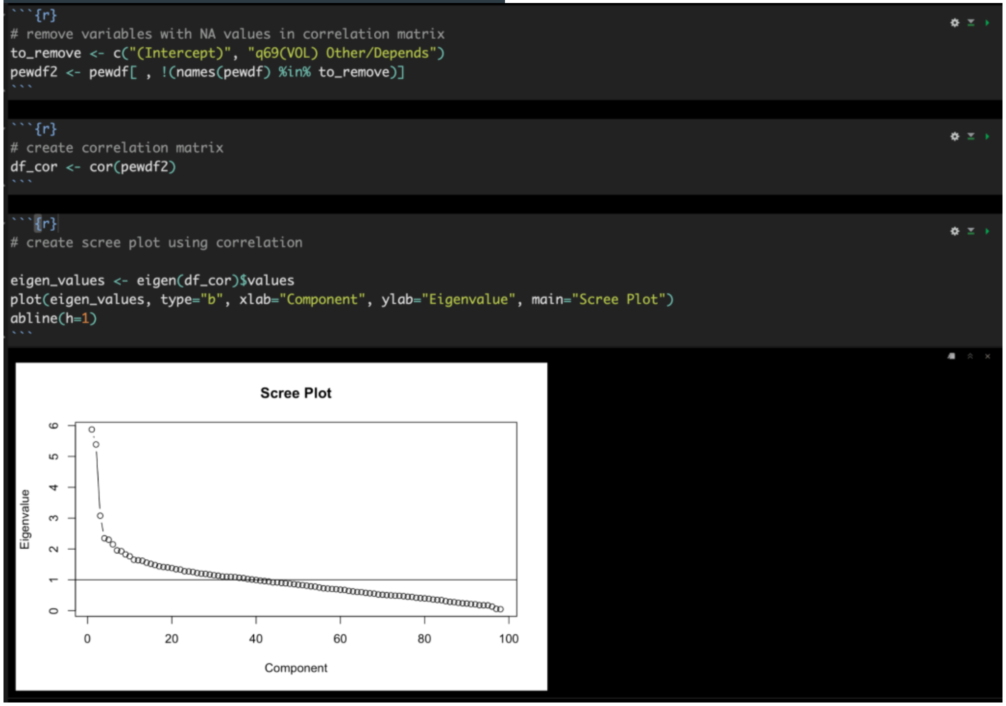
Appendix

Exhibit A - Data Preparation



```
# see if there is a subset of opinion items with complete data
item_complete = c() # initialize list of names of items
for (item in seq(along = names(pewopinion))) {
   if (sum(complete.cases(pewopinion[,item])) == nrow(pewopinion))
       item_complete = c(item_complete, names(pewopinion)[item])
}
cat("\n\nNumber of items with complete data:",length(item_complete))

# variable names for items with complete data
cat("\nNames of items with complete data:\n", item_complete)

# check status of working data frame including items with complete data
# looks like we have 30 multi-category factor items to work with
pewwork = pewopinion[,item_complete]
print(str(pewwork))

# build design matrix formula from variable names in item_complete
design_string = paste("~ q1 + q2 + q19 + q20 + q25 + q47 + q50a + q50b +",
"q50c + q50d + q50e + q58 + q60 + q61a + q61b +",
"q61c + q64 + q65a + q65b + q65c + q65d + q65e +",
"q66 + q68a + q68b + q68d + q69 + q70 + q71 + q75")

# fast way to create binary indicator variables for all items with complete data
# this converts the 30 multi-category items into 100 binary indicator variables
pewmat = model.matrix(formula(design_string), data = pewwork)
print(str(pewmat))

pewdf = as.data.frame(pewmat)
```

Exhibit B - Scree Plot of Eigenvalues

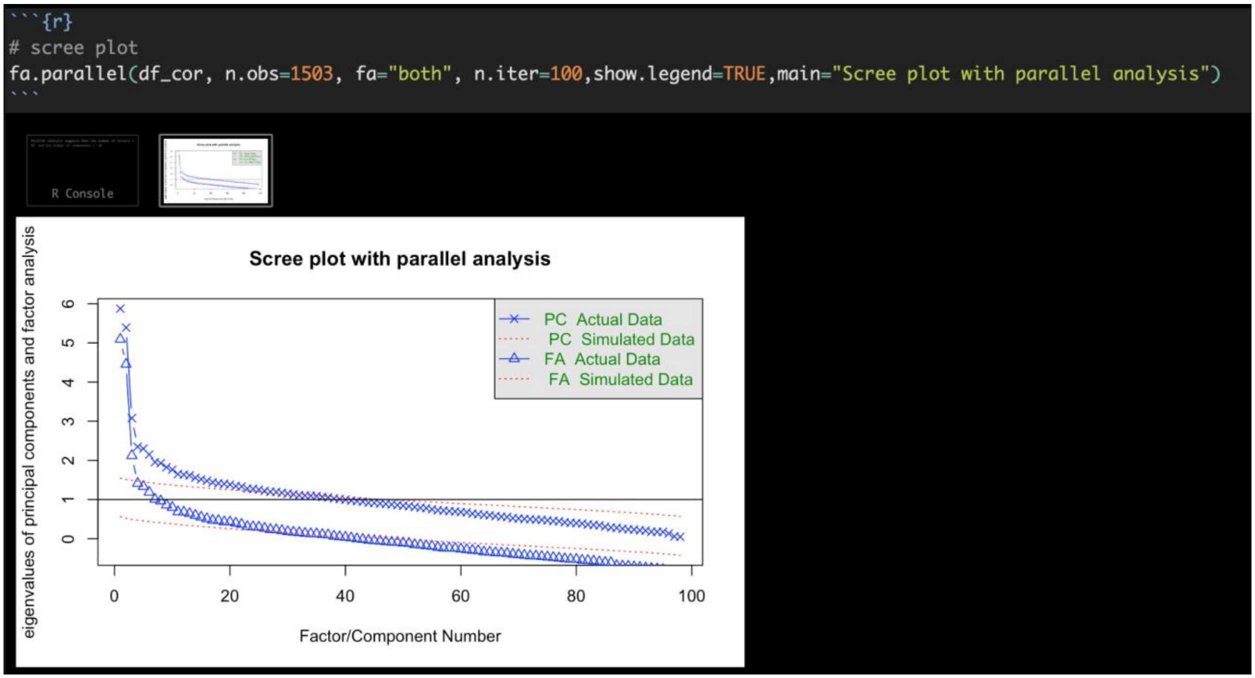


~~~~
```{r}
# remove variables with NA values in correlation matrix
to_remove <- c("(Intercept)", "q69(VOL) Other/Depends")
pewdf2 <- pewdf[ , !(names(pewdf) %in% to_remove)]
```

```{r}
# create correlation matrix
df_cor <- cor(pewdf2)
```

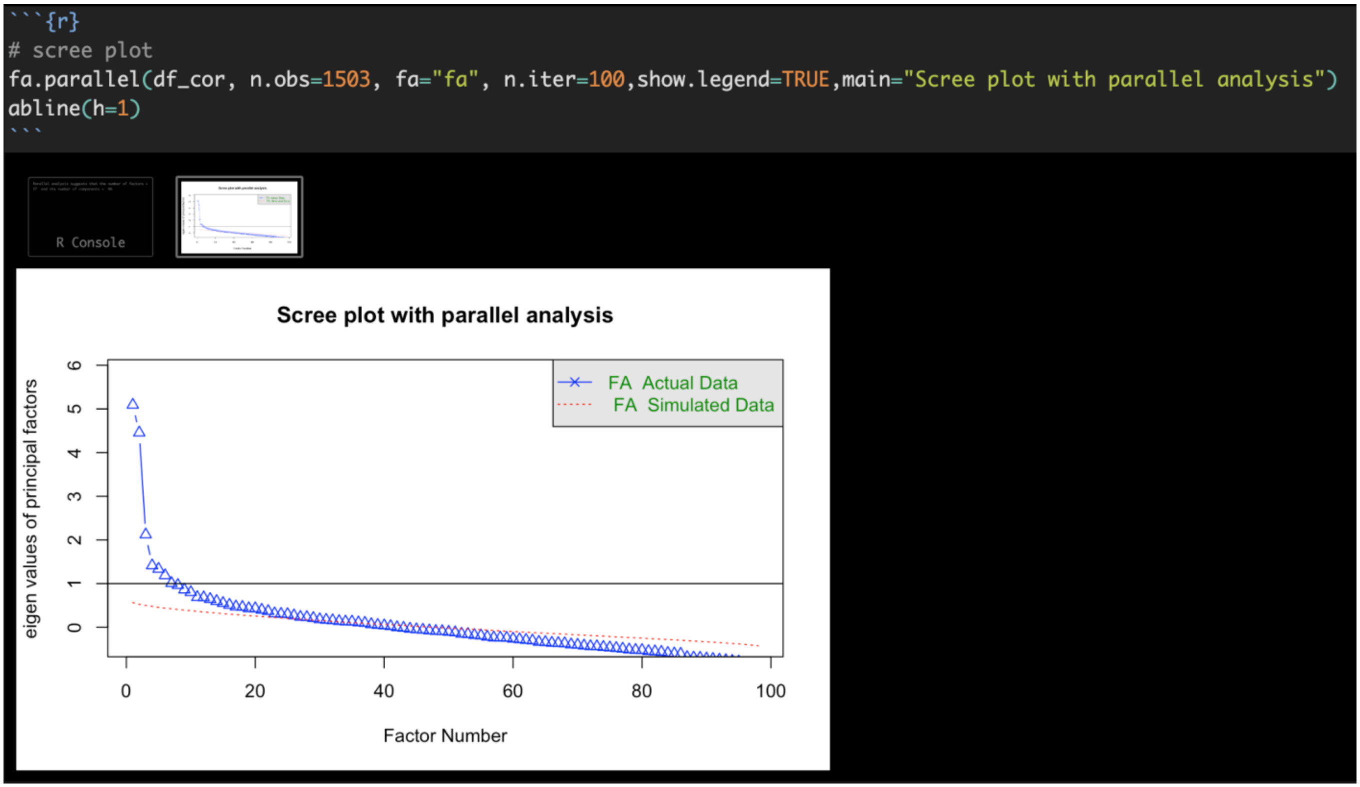
```{r}
# create scree plot using correlation

eigen_values <- eigen(df_cor)$values
plot(eigen_values, type="b", xlab="Component", ylab="Eigenvalue", main="Scree Plot")
abline(h=1)
```
~~~~

Exhibit C - Scree Plot comparing PCA vs EFA



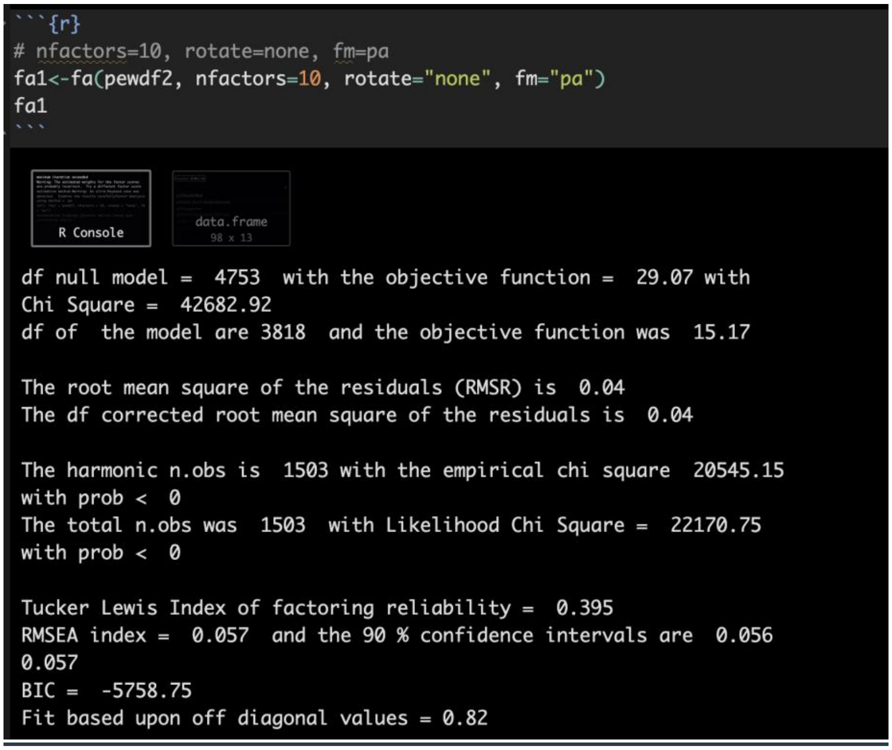
```{r}
# scree plot
fa.parallel(df_cor, n.obs=1503, fa="both", n.iter=100,show.legend=TRUE,main="Scree plot with parallel analysis")
```

Exhibit D - Scree Plot of EFA



```{r}
# scree plot
fa.parallel(df_cor, n.obs=1503, fa="fa", n.iter=100,show.legend=TRUE,main="Scree plot with parallel analysis")
abline(h=1)
```

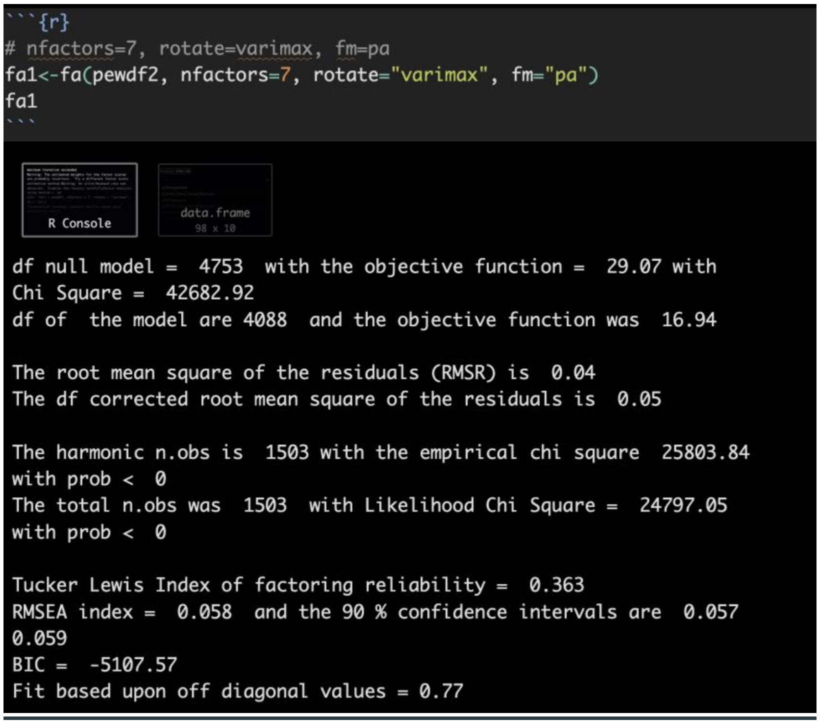
Exhibit E - EFA where nfactors = 10, rotate = “none”, fm = “pa”



~~~
```{r}
# nfactors=10, rotate=none, fm=pa
fa1<-fa(pewdf2, nfactors=10, rotate="none", fm="pa")
fa1
```
~~~

```
df null model =  4753  with the objective function =  29.07 with
Chi Square =  42682.92
df of  the model are 3818  and the objective function was  15.17

The root mean square of the residuals (RMSR) is  0.04
The df corrected root mean square of the residuals is  0.04

The harmonic n.obs is  1503 with the empirical chi square  20545.15
with prob <  0
The total n.obs was  1503  with Likelihood Chi Square =  22170.75
with prob <  0

Tucker Lewis Index of factoring reliability =  0.395
RMSEA index =  0.057  and the 90 % confidence intervals are  0.056
0.057
BIC =  -5758.75
Fit based upon off diagonal values = 0.82
```

Exhibit F - EFA where nfactors = 7, rotate = "varimax", fm= "pa"

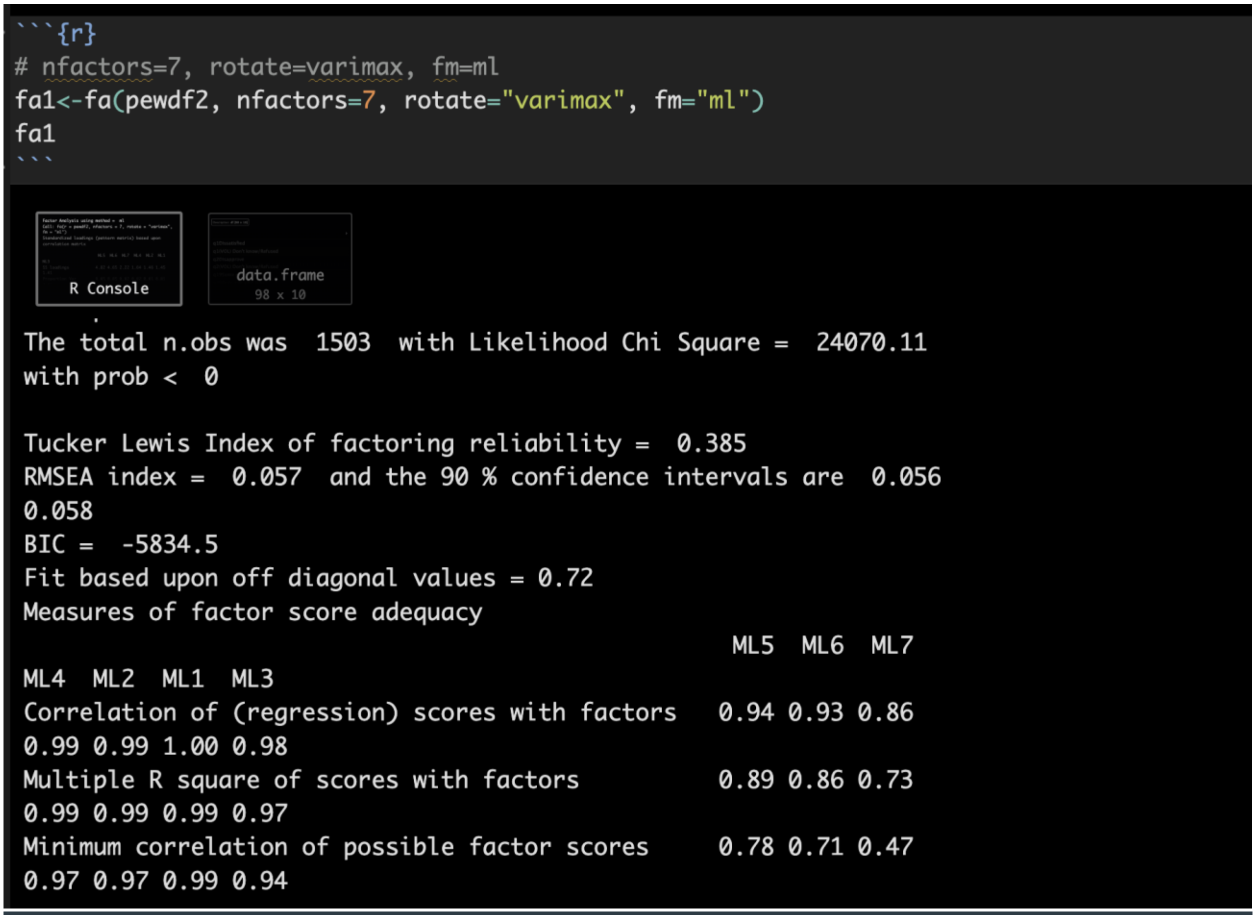


~~~
```{r}
# nfactors=7, rotate=varimax, fm=pa
fa1<-fa(pewdf2, nfactors=7, rotate="varimax", fm="pa")
fa1
```
~~~

```
df null model =  4753  with the objective function =  29.07 with
Chi Square =  42682.92
df of  the model are 4088  and the objective function was  16.94

The root mean square of the residuals (RMSR) is  0.04
The df corrected root mean square of the residuals is  0.05

The harmonic n.obs is  1503 with the empirical chi square  25803.84
with prob <  0
The total n.obs was  1503  with Likelihood Chi Square =  24797.05
with prob <  0

Tucker Lewis Index of factoring reliability =  0.363
RMSEA index =  0.058  and the 90 % confidence intervals are  0.057
0.059
BIC =  -5107.57
Fit based upon off diagonal values = 0.77
```

	PA1	PA2	PA6	PA5	PA4	PA3	PA7
SS Loadings	4.56	3.99	2.33	2.29	2.27	1.99	1.42
Proportion Var	0.05	0.04	0.02	0.02	0.02	0.02	0.01
Cumulative Var	0.05	0.09	0.11	0.13	0.16	0.18	0.19
Proportion Explained	0.24	0.21	0.12	0.12	0.12	0.11	0.08
Cumulative Proportion	0.24	0.45	0.58	0.70	0.82	0.92	1.00

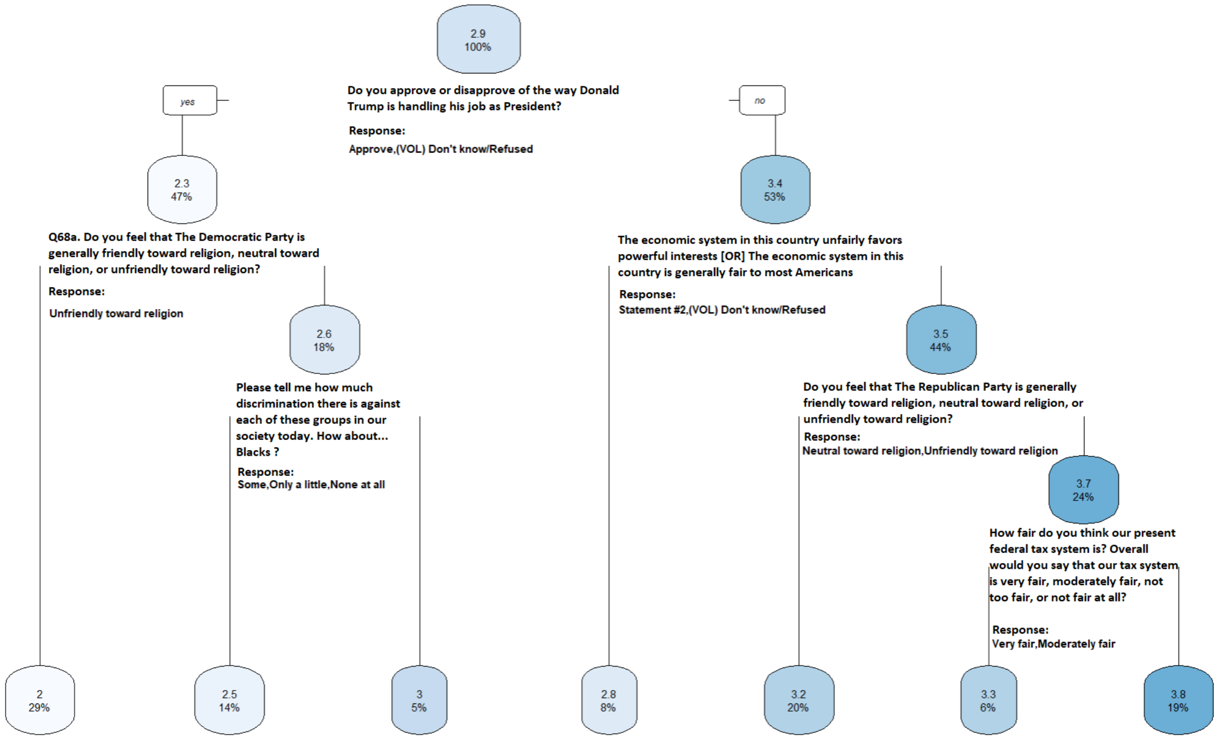
Exhibit G - EFA where nfactors = 7, rotate = "varimax", fm= "ml"

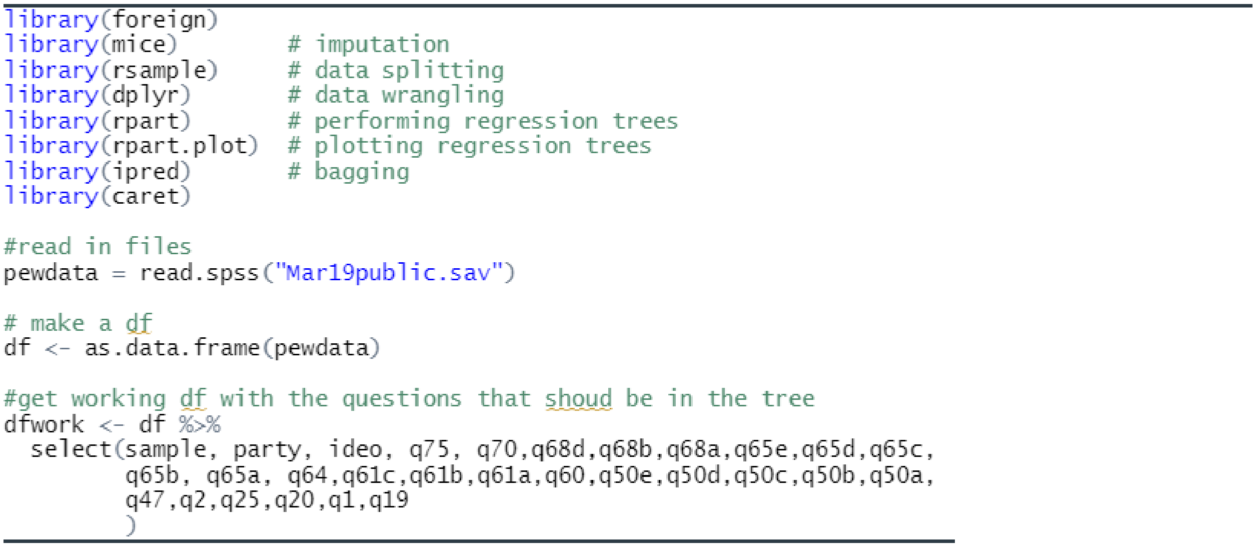


~~~
```{r}
# nfactors=7, rotate=varimax, fm=ml
fa1<-fa(pewdf2, nfactors=7, rotate="varimax", fm="ml")
fa1
```
~~~

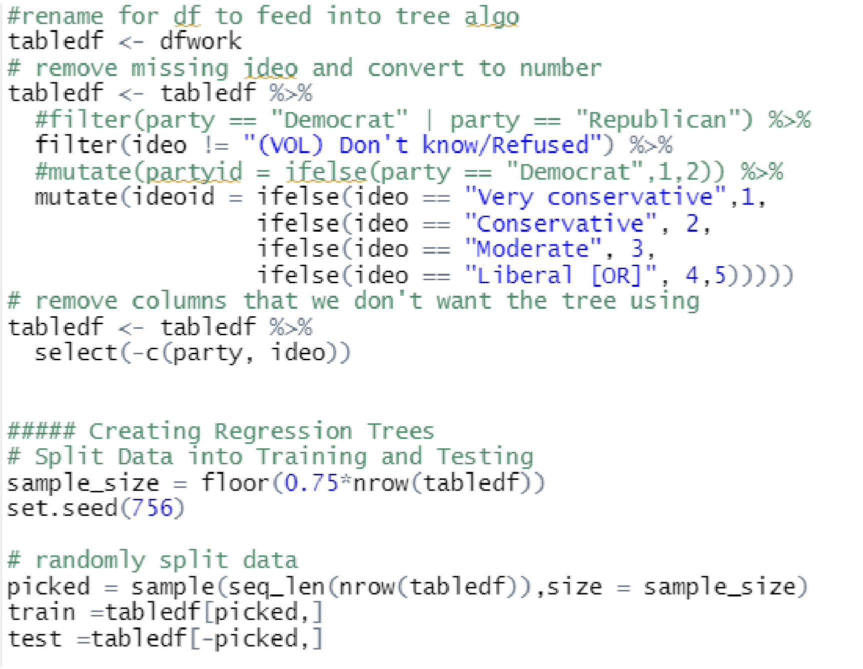
```
The total n.obs was  1503  with Likelihood Chi Square =  24070.11
with prob <  0

Tucker Lewis Index of factoring reliability =  0.385
RMSEA index =  0.057  and the 90 % confidence intervals are  0.056
0.058
BIC =  -5834.5
Fit based upon off diagonal values = 0.72
Measures of factor score adequacy
                                                  ML5  ML6  ML7
ML4  ML2  ML1  ML3
Correlation of (regression) scores with factors   0.94 0.93 0.86
0.99 0.99 1.00 0.98
Multiple R square of scores with factors          0.89 0.86 0.73
0.99 0.99 0.99 0.97
Minimum correlation of possible factor scores     0.78 0.71 0.47
0.97 0.97 0.99 0.94
```

Exhibit H - Regression Tree

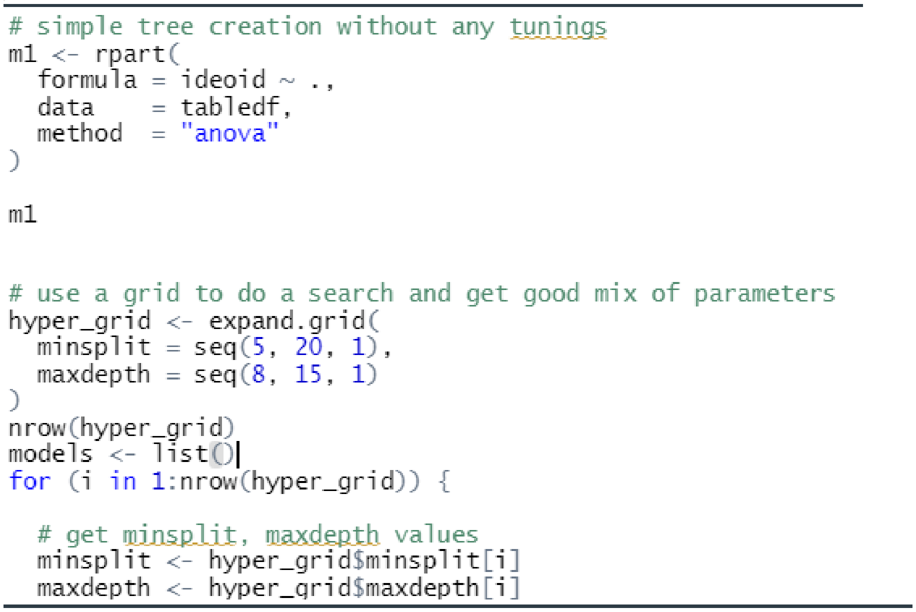

2.9
100%
yes
no
Do you approve or disapprove of the way Donald Trump is handling his job as President?
Response:
Approve,(VOL) Don't know/Refused
2.3
47%
Q68a. Do you feel that The Democratic Party is generally friendly toward religion, neutral toward religion, or unfriendly toward religion?
Response:
Unfriendly toward religion
2.6
18%
Please tell me how much discrimination there is against each of these groups in our society today. How about... Blacks ?
Response:
Some,Only a little,None at all
3.4
53%
The economic system in this country unfairly favors powerful interests [OR] The economic system in this country is generally fair to most Americans
Response:
Statement #2,(VOL) Don't know/Refused
3.5
44%
Do you feel that The Republican Party is generally friendly toward religion, neutral toward religion, or unfriendly toward religion?
Response:
Neutral toward religion,Unfriendly toward religion
3.7
24%
How fair do you think our present federal tax system is? Overall would you say that our tax system is very fair, moderately fair, not too fair, or not fair at all?
Response:
Very fair,Moderately fair
2
29%
2.5
14%
3
5%
2.8
8%
3.2
20%
3.3
6%
3.8
19%



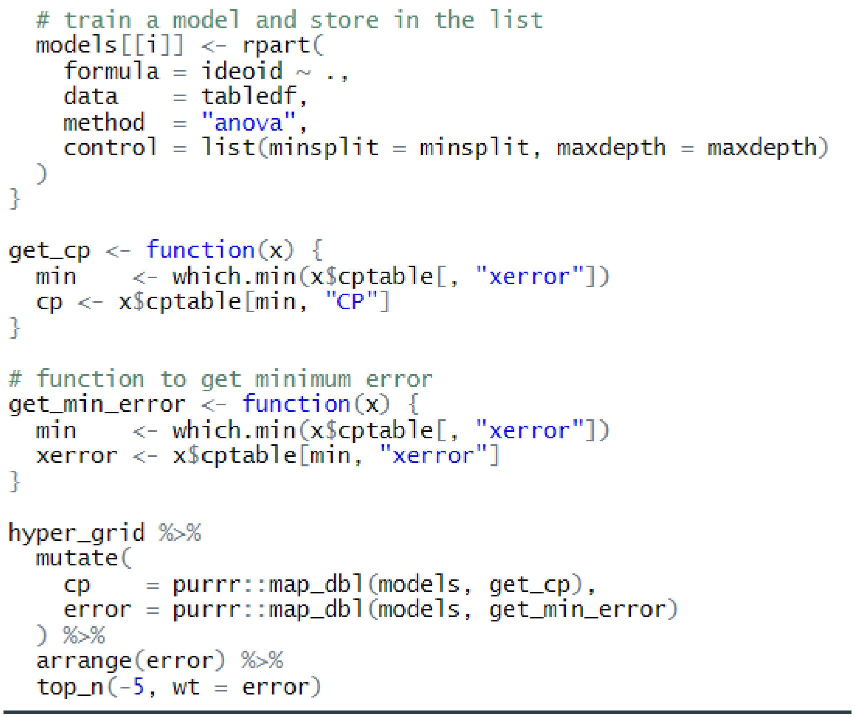
```
library(foreign)
library(mice)          # imputation
library(rsample)       # data splitting
library(dplyr)         # data wrangling
library(rpart)         # performing regression trees
library(rpart.plot)    # plotting regression trees
library(ipred)         # bagging
library(caret)

#read in files
pewdata = read.spss("Mar19public.sav")

# make a df
df <- as.data.frame(pewdata)

#get working df with the questions that shoud be in the tree
dfwork <- df %>%
  select(sample, party, ideo, q75, q70,q68d,q68b,q68a,q65e,q65d,q65c,
         q65b, q65a, q64,q61c,q61b,q61a,q60,q50e,q50d,q50c,q50b,q50a,
         q47,q2,q25,q20,q1,q19
         )
```



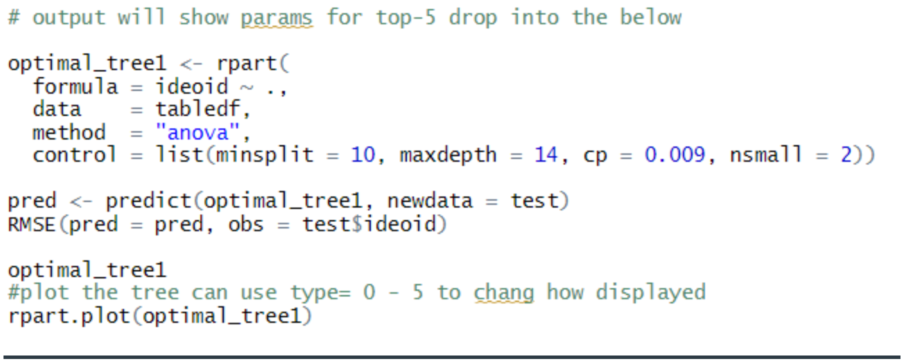
```
#rename for df to feed into tree algo
tabledf <- dfwork
# remove missing ideo and convert to number
tabledf <- tabledf %>%
  #filter(party == "Democrat" | party == "Republican") %>%
  filter(ideo != "(VOL) Don't know/Refused") %>%
  #mutate(partyid = ifelse(party == "Democrat",1,2)) %>%
  mutate(ideoid = ifelse(ideo == "Very conservative",1,
                  ifelse(ideo == "Conservative", 2,
                  ifelse(ideo == "Moderate", 3,
                  ifelse(ideo == "Liberal [OR]", 4,5)))))
# remove columns that we don't want the tree using
tabledf <- tabledf %>%
  select(-c(party, ideo))


##### Creating Regression Trees
# Split Data into Training and Testing
sample_size = floor(0.75*nrow(tabledf))
set.seed(756)

# randomly split data
picked = sample(seq_len(nrow(tabledf)),size = sample_size)
train =tabledf[picked,]
test =tabledf[-picked,]
```



```
# simple tree creation without any tunings
m1 <- rpart(
  formula = ideoid ~ .,
  data    = tabledf,
  method  = "anova"
)

m1


# use a grid to do a search and get good mix of parameters
hyper_grid <- expand.grid(
  minsplit = seq(5, 20, 1),
  maxdepth = seq(8, 15, 1)
)
nrow(hyper_grid)
models <- list()
for (i in 1:nrow(hyper_grid)) {

  # get minsplit, maxdepth values
  minsplit <- hyper_grid$minsplit[i]
  maxdepth <- hyper_grid$maxdepth[i]
```



```
  # train a model and store in the list
  models[[i]] <- rpart(
    formula = ideoid ~ .,
    data    = tabledf,
    method  = "anova",
    control = list(minsplit = minsplit, maxdepth = maxdepth)
  )
}

get_cp <- function(x) {
  min    <- which.min(x$cptable[, "xerror"])
  cp <- x$cptable[min, "CP"]
}

# function to get minimum error
get_min_error <- function(x) {
  min    <- which.min(x$cptable[, "xerror"])
  xerror <- x$cptable[min, "xerror"]
}

hyper_grid %>%
  mutate(
    cp    = purrr::map_dbl(models, get_cp),
    error = purrr::map_dbl(models, get_min_error)
  ) %>%
  arrange(error) %>%
  top_n(-5, wt = error)
```



```
# output will show params for top-5 drop into the below

optimal_tree1 <- rpart(
  formula = ideoid ~ .,
  data    = tabledf,
  method  = "anova",
  control = list(minsplit = 10, maxdepth = 14, cp = 0.009, nsmall = 2))

pred <- predict(optimal_tree1, newdata = test)
RMSE(pred = pred, obs = test$ideoid)

optimal_tree1
#plot the tree can use type= 0 - 5 to chang how displayed
rpart.plot(optimal_tree1)
```

Exhibit I

PA1: Measures satisfaction with the current political and economic environment

PA2/7: Measure opinions relating to the pervasiveness of discrimination

PA3: Captures a cross-section of respondents that do not find the country's tax system fair and categorize themselves as "angry" with the Federal Government

PA4: Measures perceptions of the Democratic parties attitudes toward religion

PA5: Measures trust in the Federal Government

PA6: Measures perceptions of the United States tax system