MINI PROJECT

ELECTION STATISTICS

A.SAURABH

INDEX

1. PROJECT OBJECTIVE

2. EXPLORATORY DATA ANALYSIS

2.1. ENVIRONMENT SETUP AND DATA IMPORT

2.2 INSTALL NECESSARY PACKAGES

2.3 SETTING UP WORKING DIRECTORY

2.4 IMPORT DATASET

2.5 VARIABLE IDENTIFICATION

3. DATA IDENTIFICATION

4. CONCLUSION

5.SOURCE CODE

PROJECT OBJECTIVE

The objective of this project is to analyze the data and build a model accordingly. The data given is for state assembly elections in India. To build the model we must find the following:

1. Identify the important factors for a candidate to win the elections
2. Probability of a particular candidate winning
3. Classification: To classify the candidates as winning or not winning

The key objective is to explain the factors behind the 2017 outcome for state assembly.

Here we build a classification model and a continuous response model for the data we have acquired.

Classification model decides the winner of the elections and continuous response model determines the votes shares.

Since the state assembly elections happen only once in five years we require the data of previous winner and 2012 data. The incumbent effect comes into play when backdated data is used and here we know if someone who had won an election in 2012 contests again and has a higher probability of winning.

EXPLORATORY DATA ANALYSIS

A Typical Data exploration activity consists of the following steps:

1. Environment Set up and Data Import

2. Variable Identification

Install necessary Packages and Invoke Libraries

Use this section to install necessary packages and invoke associated

libraries. Having all the packages at the same places increases code

readability.

Set up working Directory

Setting a working directory on starting of the R session makes importing

and exporting data files and code files easier. Basically, working directory

is the location/ folder on the PC where you have the data, codes etc.

related to the project.

The location in which the file is stored is given by:

setwd("C:/Users/Saurabh/Downloads")

Please refer Appendix A for Source Code.

Import and Read the Dataset

The given dataset is in .xls format. Hence, the command ‘read.xls is

used for importing the file. The dataset are in both xls and xlsx format. There are four different datasets with the details of candidates individually and winners. The state chosen to run the regression model is Goa.

The list of datasets is given below;

1. LA 2017(Legislative assembly data 2017)
2. AE2012\_8913 (Data of winners from 2012)
3. Candidates\_2017 (Individual candidate data)
4. Candidates\_v2 (Winners )

PACKAGES INSTALLED

The following packages are installed to run the R script:

Xlsx: This package is used to run xlsx format file mainly for importing the data.

Dplyr: This package is to provide functions to manipulate data like using filter and mutate I in this case.

Readxl: The Readxl is used to import excel files into R.

Mass: Functions and datasets to support venables and ripley.

Car: Companion to applied Regression

Caret: The [caret](http://cran.r-project.org/web/packages/caret/index.html) package has several functions that attempt to streamline the model building and evaluation process. It is used for model training and parameter tuning.

The same packages given above are installed with the command install and called using library command.

VARIABLE IDENTIFICATION

The variables are the parameters given in the function on which the analysis is carried out. The parameters are called features.

The R function used for variable identification are as given below**:**

**Dim-** this function is used to find out the dimensions i.e. number of rows and columns in the dataset.

**Head**-this function is used to get the head values which are the starting values by default only 10 are displayed but it can be increased as per specification.

**Tail-** this function is used to read and focus only on the ending values or last few values by default only 10 are displayed but it can be increased as per specification.

Head and tail are used to check if the data provided in the particular dataset behaves similarly throughout the table or is random.

**View-**it gives the tabulated from of the data as shown in the dataset in the output window.

**Names-**this function gives only the names, or the parameters specified in the dataset arranged as a row.

**Summary-**it is a generic function used to give summaries of the objects in the dataset to be analyzed with different parameters included such as minimum, maximum, median and quantiles depending on the class of the variable individually for all the features. Result of model fitting functions.

**Str-**the function str displays the structure of an arbitrary R object. It is a diagnostic function and an alternate to the summary function which displays the class of the object and its name.

DATA IDENTIFICATION

There are two methods to be implemented on the data before running a model

1. Data cleaning
2. Data merging

DATA CLEANING

Here we have data from more than one dataset so we have to make them similar in terms to run a classification model. We convert the data to lower case and change the names to only the first names.

We eliminate the data which is of no use to us or is not of any significance.

We change the names of the columns and duplicate the columns such that the combined values are present.

DATA MERGING

We need to merge the data and add new sorted data here to make it unique we use the function to concatenate the first name of the candidate with the constituency and first name

of candidate with the party which makes it unique and easier to classify. Instead of eliminating few entries which leads to loss of data.

cand\_list\_clean is the dataset with sorted values in the dataset.

ads\_creation is the user defined function.

The names in the dataset signify the exact meaning as cand\_list\_clean is cleaned data and with the prefix merge is

combined or merged data.

INSIGHTS

Here we have used logistic regression for classification of the data to find winner or not.

In the summary of the model using logistic regression on using glm command, we get the list of coefficients, and on inspection we find that the total assets, and the party determine the winning factor and are significant and slightly less significant in the order is candidate sex. These are the factors which affect the state elections.

We choose the data depending on the

Data availability

Experience

Theories

Analysis

But here the source of data is from websites such as;

<http://myneta.info/goa2017/index.php?action=summary&subAction=candidates_analyzed&sort=candidate#summary>

<http://eci.nic.in/eci_main1/ElectionStatistics.aspx>

The data is in excel and raw format and we need data cleaning and merging to proceed with logistic regression.

More the number of independent variables we cannot make use of linear modelling.

There are five models being run here for the best and the most accurate prediction of the results.

Before performing the logistic regression, we

Here firstly we set up the regression model and then try linear or nonlinear forms.

Check R squared for overall fit.

Check individual coefficients if they are good then we proceed.

Significance of variables that is if the p-value is less than or equal to the **0.05 then it said that the variables are significant.**

For that, we need to check the estimate given in the coefficients. In the observations we find that most of the values are significant

Multicollinearity using VIF

VIF stands for variance inflation factor.

Estimate and predict

For picking the explanatory variables we choose on the basis of significance

Here we have employed backward selection to run regression on the dataset.

After trying all subsets we choose the best.

The r squared values determines the goodness of fit. i.e. explains the variance in the data set. The R squared values lie between zero and one. Here the value is 0.38.

This R squared says the 38% variance is explained by the model.

It suggests good fit. R squared gives overall accuracy and VIF=1/(1-R^2)

Fisher scoring iterations define the number of iterations required to estimate a model.

VIF gives the multicollinearity. The condition is that if VIF is high then the multicollinearity is high. The VIF keeps decreasing that and is close to one. Here the multicollinearity is low of most of the models. The values are close to 2.

The negative coefficient values suggest that there is a positive relationship among the variables. Since most of them have significant statistic the multi collinearity is low.

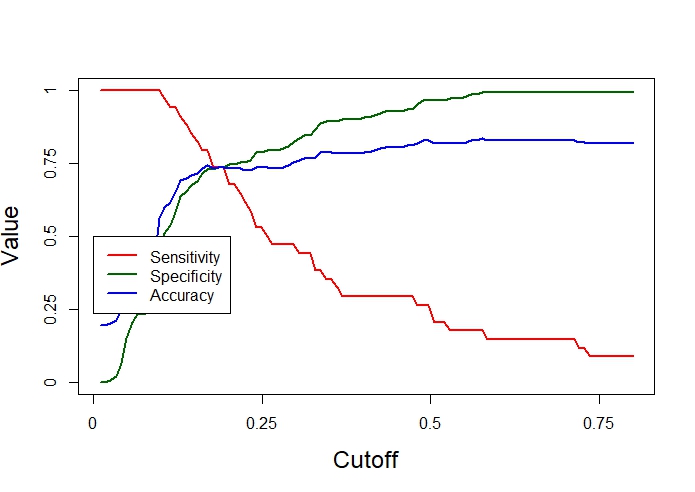
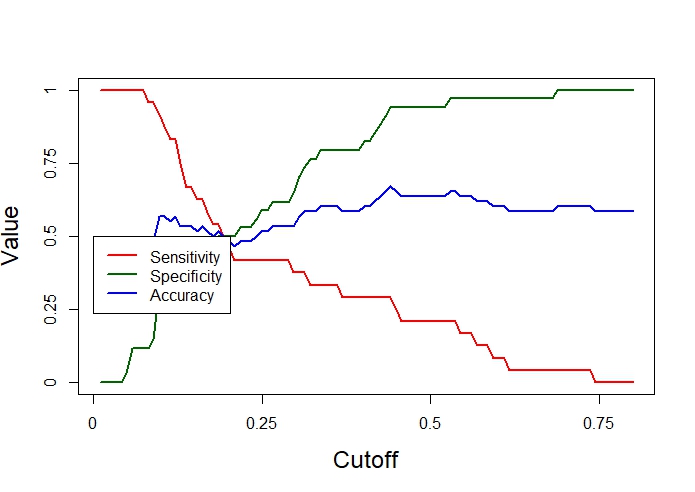
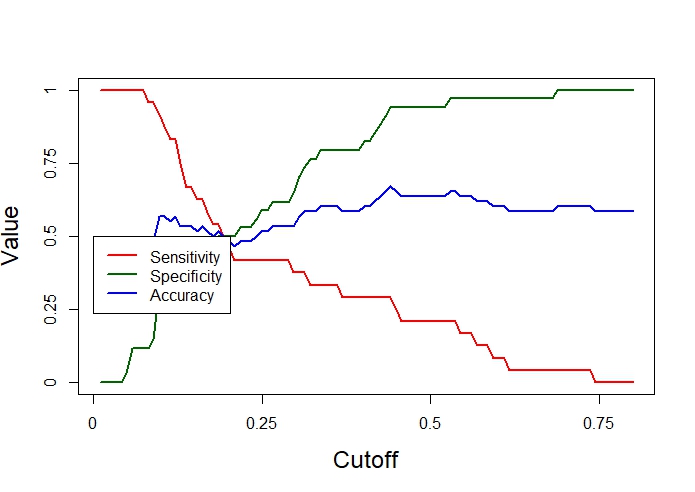
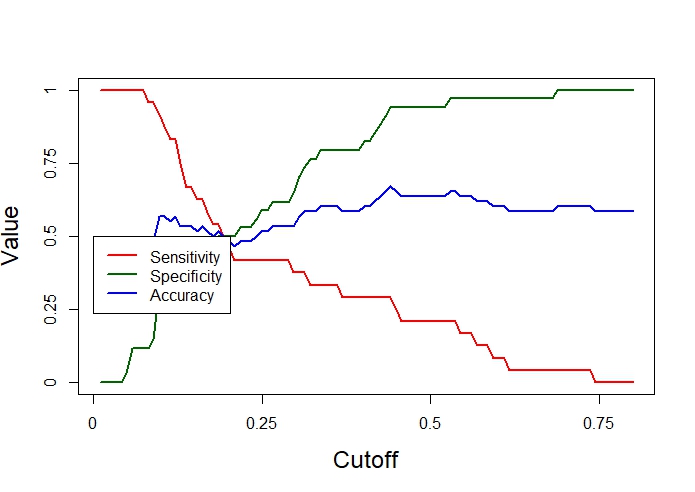
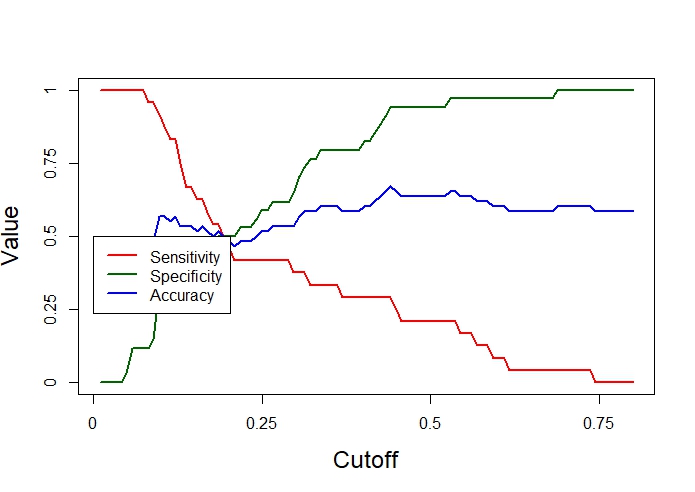
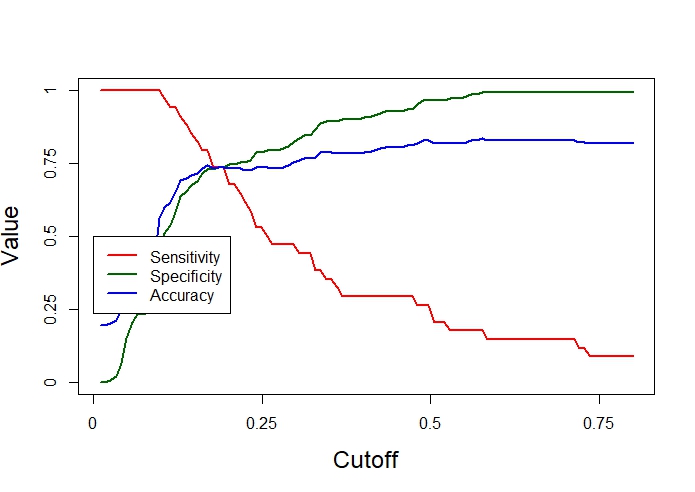
From the plots given below we determine the important factors like specificity, sensitivity and accuracy. The factors are obtained from the confusion matrix. It gives the number of data which are accurately classified as well as the data which is inaccurately classified.

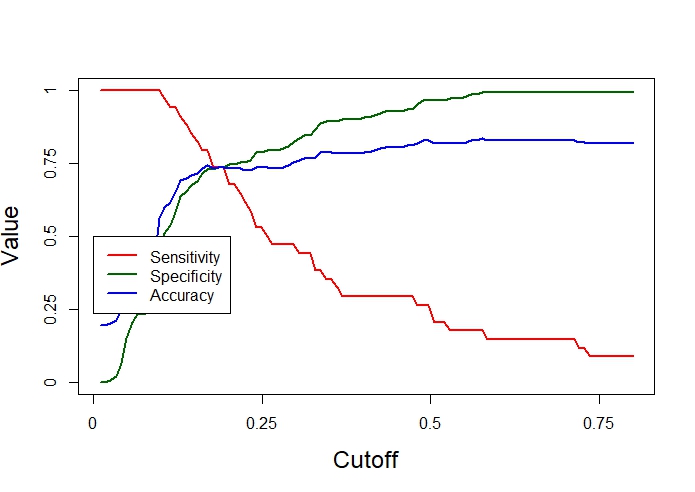
The vote shares are decided by simple linear regression.

Last step is to refine the model if possible so that it explains most of the values and gives the best possible fit.

PLOTS

The plots given below explain the factors sensitivity, specificity and accuracy for value v/s cutoff.





There are plots which show us the representation of value vs the cutoff and

the point where they coincide gives us the value. We get to pick a cutoff value which is convenient to give the best model.

Logistic regression does not classify the winner but specifies the probability. According to the model the value is at 20 % and optimum probability cutoff.

If we choose 0.5 as the cutoff the value is lower and if we choose 0.2 as the cutoff then we get better probabilities.

SOURCE CODE

**#=======================================================================**

**#**

**# *Data Analysis – ELECTIONS LEGISLATIVE ASSEMBLY***

**#**

**#=======================================================================**

#Setting up working directory

> setwd("C:/Users/Saurabh/Downloads")

>

# necessary packages

> #library(xlsx)

> library(dplyr)

> library(readxl)

> library("MASS")

> library(car)

> library(caret)

>

# Creating a user defined function to combine the candidate name and abr/acn

> ads\_creation <- function(Dataset)

+ {

+ #Converting name to lower case

+ Dataset$CAND\_NAME <- tolower(Dataset$CAND\_NAME)

+ Dataset$first\_name <- gsub("([A-Za-z]+).\*", "\\1", Dataset$CAND\_NAME)

+

+ Dataset$AC\_NAME <- tolower(Dataset$AC\_NAME)

+ Dataset$NameConst <- paste0(Dataset$first\_name,":",Dataset$AC\_NAME)

+ Dataset$PARTYABBRE <- tolower(Dataset$PARTYABBRE)

+ Dataset$NameParty <- paste0(Dataset$first\_name,":",Dataset$PARTYABBRE)

+

+

+

+ cand\_merge1 <- merge(Dataset, cand\_list\_clean[,!names(cand\_list\_clean) %in% c("NameParty","first\_name")],

+ by = "NameConst", all.x = T)

+

+ cand\_merge2 <- merge(Dataset, cand\_list\_clean[,!names(cand\_list\_clean) %in% c("NameConst","first\_name")],

+ by = "NameParty", all.x = T)

+

+ cand\_merge\_tot <- rbind(cand\_merge1[,!names(cand\_merge1) %in% c("NameConst","first\_name","NameParty")],

+ cand\_merge2[,!names(cand\_merge2) %in% c("NameConst","first\_name","NameParty")])

+

+ cand\_merge\_tot <- cand\_merge\_tot[!duplicated(cand\_merge\_tot),]

+ cand\_merge\_tot <- cand\_merge\_tot[complete.cases(cand\_merge\_tot),]

+ cand\_merge\_tot\_1 <- cand\_merge\_tot[cand\_merge\_tot$AC\_NAME==cand\_merge\_tot$Constituency & cand\_merge\_tot$PARTYABBRE==cand\_merge\_tot$Party,]

+

+ cand\_merge\_tot\_2 <- cand\_merge\_tot[cand\_merge\_tot$AC\_NAME==cand\_merge\_tot$Constituency & cand\_merge\_tot$PARTYABBRE!=cand\_merge\_tot$Party,]

+ cand\_merge\_tot\_2 <- cand\_merge\_tot\_2[cand\_merge\_tot\_2$CAND\_NAME==cand\_merge\_tot\_2$Candidate,]

+

+ cand\_merge\_all <- rbind(cand\_merge\_tot\_1,cand\_merge\_tot\_2)

+ cand\_merge\_all$CAND\_NAME[duplicated(cand\_merge\_all$CAND\_NAME)]

+

+ #-----------------------------------------------------------------------------------------

+

+ cand\_merge <- merge(Dataset, cand\_list\_clean[,!names(cand\_list\_clean) %in% c("NameConst","first\_name","NameParty")],

+ by.x = "CAND\_NAME",by.y = "Candidate", all.x = T)

+ cand\_merge$Candidate <- NA

+ cand\_merge <- cand\_merge[,!names(cand\_merge) %in% c("NameConst","first\_name","NameParty")]

+

+ missing\_cand <- cand\_merge[cand\_merge$CAND\_NAME %in% cand\_merge$CAND\_NAME[!cand\_merge$CAND\_NAME %in% cand\_merge\_all$CAND\_NAME],]

+ missing\_cand <- missing\_cand[!is.na(missing\_cand$Education),]

+

+ cand\_merge\_all <- rbind(cand\_merge\_all,missing\_cand)

+ cand\_merge\_all <- cand\_merge\_all[,!names(cand\_merge\_all) == "Candidate"]

+ cand\_merge\_all <- cand\_merge\_all[complete.cases(cand\_merge\_all),]

+

+ #---------------------------------------------------------------------------------------

+

+ cand\_final <- cand\_merge\_all[,names(cand\_merge\_all) %in% c("YEAR","DIST\_NAME",

+ "AC\_NO", "AC\_NAME","AC\_TYPE","CAND\_NAME","CAND\_SEX",

+ "CAND\_AGE","PARTYABBRE","TOTVOTPOLL","POSITION",

+ "Criminal Case","Education","Total Assets","Liabilities")]

+

+ #---------------------------------------------------------------------------------------

+

+ cand\_final\_1 <- cand\_final %>%

+ rowwise() %>%

+ mutate(total\_assets = as.numeric(gsub(",", "", strsplit(`Total Assets`, "\\s+")[[1]][2])),

+ liabilities = as.numeric(gsub(",", "", strsplit(Liabilities, "\\s+")[[1]][2])))

+

+ cand\_final\_1 <- cand\_final\_1[,!names(cand\_final\_1) %in% c("Total Assets","Liabilities")]

+

+

+ #---------------------------------------------------------------------------------------

+ factor\_data <- data.frame(sapply(cand\_final\_1[,c("DIST\_NAME","AC\_TYPE","CAND\_NAME","CAND\_SEX","PARTYABBRE","Education")], FUN = as.factor))

+

+ names(cand\_final\_1)[colnames(cand\_final\_1) %in% c("DIST\_NAME","AC\_TYPE","CAND\_NAME","CAND\_SEX","PARTYABBRE","Education")] <- c("DIST\_NAME.x","AC\_TYPE.x","CAND\_NAME.x","CAND\_SEX.x","PARTYABBRE.x","Education.x")

+ cand\_final\_2 <- cbind(cand\_final\_1,factor\_data)

+

+ cand\_final\_2

+ }

>

> #-----------------------------------------------------------------------------------------

> #Reading the candidates details

> cand\_list <- read\_excel("Candidates\_v2.xlsx")

> names(cand\_list)[colnames(cand\_list)=="Candidate∇"] <- "Candidate"

> cand\_list\_clean <- cand\_list[complete.cases(cand\_list),]

> cand\_list\_clean$Candidate <- tolower(cand\_list\_clean$Candidate)

>

> cand\_list\_clean$first\_name <- gsub("([A-Za-z]+).\*", "\\1", cand\_list\_clean$Candidate)

> cand\_list\_clean$Constituency <- tolower(cand\_list\_clean$Constituency)

> cand\_list\_clean$NameConst <- paste0(cand\_list\_clean$first\_name,":",cand\_list\_clean$Constituency)

>

> cand\_list\_clean$Party <- tolower(cand\_list\_clean$Party)

> cand\_list\_clean$NameParty <- paste0(cand\_list\_clean$first\_name,":",cand\_list\_clean$Party)

>

> #----------------------------------------------------------------------------------------

>

>

> #reading the election results from 2012

> cand\_res\_2012 <- read\_excel("AE2012\_8913.xls", sheet = 1)

>

> #filtering for Goa

> cand\_res\_2012\_goa <- cand\_res\_2012 %>% filter(ST\_NAME == "Goa")

> cand\_train <- ads\_creation(cand\_res\_2012\_goa)

> must\_convert<-sapply(cand\_train,is.factor)

> M2<-data.frame(sapply(cand\_train[,must\_convert],unclass))

>

> train\_out<-cbind(cand\_train[,!names(cand\_train)%in% c("DIST\_NAME.x","AC\_TYPE.x","CAND\_NAME.x","CAND\_SEX.x","PARTYABBRE.x","Education.x","DIST\_NAME","AC\_TYPE","CAND\_NAME","CAND\_SEX","PARTYABBRE","Education")],M2)

> mapping<-cbind(cand\_train[,names(cand\_train)%in% c("DIST\_NAME.x","AC\_TYPE.x","CAND\_NAME.x","CAND\_SEX.x","PARTYABBRE.x","Education.x")],M2)

>

> #removing totvotpoll becuase this is also a dependant variable

> train\_out <- train\_out[,!names(train\_out) %in% c("AC\_NAME","YEAR","TOTVOTPOLL")]

>

> train\_out$POSITION <- if\_else(train\_out$POSITION==1,1,0)

>

> mapping\_dist <- mapping[,c("DIST\_NAME.x","DIST\_NAME")]

> mapping\_dist <- mapping\_dist[!duplicated(mapping\_dist),]

>

> mapping\_ac <- mapping[,c("AC\_TYPE.x","AC\_TYPE")]

> mapping\_ac <- mapping\_ac[!duplicated(mapping\_ac),]

>

> mapping\_name <- mapping[,c("CAND\_NAME.x","CAND\_NAME")]

> mapping\_name <- mapping\_name[!duplicated(mapping\_name),]

>

> mapping\_sex <- mapping[,c("CAND\_SEX.x","CAND\_SEX")]

> mapping\_sex <- mapping\_sex[!duplicated(mapping\_sex),]

>

> mapping\_party <- mapping[,c("PARTYABBRE.x","PARTYABBRE")]

> mapping\_party <- mapping\_party[!duplicated(mapping\_party),]

>

> mapping\_edu <- mapping[,c("Education.x","Education")]

> mapping\_edu <- mapping\_edu[!duplicated(mapping\_edu),]

> #----------------------------------------------------------------------------------------

> set.seed(100)

> model\_1 = glm(POSITION ~ ., data = train\_out, family = "binomial")

> summary(model\_1) #AIC 4150.1....31 coeff..nullDev 5699.5...resDev 4102.1

Call:

glm(formula = POSITION ~ ., family = "binomial", data = train\_out)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.8583 -0.6421 -0.3958 -0.1350 2.3171

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -5.980e+00 3.361e+00 -1.779 0.075204 .

AC\_NO 5.988e-02 3.894e-02 1.538 0.124132

CAND\_AGE 2.448e-04 2.064e-02 0.012 0.990536

`Criminal Case` -1.370e-01 3.012e-01 -0.455 0.649288

total\_assets 1.623e-08 4.865e-09 3.335 0.000852 \*\*\*

liabilities -5.004e-09 1.074e-08 -0.466 0.641203

DIST\_NAME -1.306e+00 9.062e-01 -1.441 0.149677

AC\_TYPE 1.496e+00 1.324e+00 1.130 0.258516

CAND\_NAME -3.736e-03 3.962e-03 -0.943 0.345665

CAND\_SEX 2.836e+00 1.415e+00 2.005 0.044990 \*

PARTYABBRE -2.559e-01 6.451e-02 -3.966 7.31e-05 \*\*\*

Education -9.686e-02 7.650e-02 -1.266 0.205471

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 191.48 on 192 degrees of freedom

Residual deviance: 152.23 on 181 degrees of freedom

AIC: 176.23

Number of Fisher Scoring iterations: 5

>

> # Stepwise selection

> model\_2<- stepAIC(model\_1, direction="both")

Start: AIC=176.23

POSITION ~ AC\_NO + CAND\_AGE + `Criminal Case` + total\_assets +

liabilities + DIST\_NAME + AC\_TYPE + CAND\_NAME + CAND\_SEX +

PARTYABBRE + Education

Df Deviance AIC

- CAND\_AGE 1 152.24 174.24

- `Criminal Case` 1 152.45 174.45

- liabilities 1 152.46 174.46

- CAND\_NAME 1 153.13 175.13

- AC\_TYPE 1 153.31 175.31

- Education 1 153.90 175.90

<none> 152.24 176.24

- DIST\_NAME 1 154.34 176.34

- AC\_NO 1 154.64 176.64

- CAND\_SEX 1 157.84 179.84

- total\_assets 1 165.24 187.24

- PARTYABBRE 1 171.31 193.31

Step: AIC=174.23

POSITION ~ AC\_NO + `Criminal Case` + total\_assets + liabilities +

DIST\_NAME + AC\_TYPE + CAND\_NAME + CAND\_SEX + PARTYABBRE +

Education

Df Deviance AIC

- `Criminal Case` 1 152.45 172.45

- liabilities 1 152.47 172.47

- CAND\_NAME 1 153.14 173.14

- AC\_TYPE 1 153.31 173.31

- Education 1 153.93 173.93

<none> 152.24 174.24

- DIST\_NAME 1 154.34 174.34

- AC\_NO 1 154.65 174.65

+ CAND\_AGE 1 152.24 176.24

- CAND\_SEX 1 157.84 177.84

- total\_assets 1 165.85 185.85

- PARTYABBRE 1 171.34 191.34

Step: AIC=172.45

POSITION ~ AC\_NO + total\_assets + liabilities + DIST\_NAME + AC\_TYPE +

CAND\_NAME + CAND\_SEX + PARTYABBRE + Education

Df Deviance AIC

- liabilities 1 152.73 170.73

- CAND\_NAME 1 153.26 171.26

- AC\_TYPE 1 153.48 171.48

- Education 1 154.09 172.09

<none> 152.45 172.45

- DIST\_NAME 1 154.59 172.59

- AC\_NO 1 154.91 172.91

+ `Criminal Case` 1 152.24 174.24

+ CAND\_AGE 1 152.45 174.45

- CAND\_SEX 1 157.84 175.84

- total\_assets 1 165.91 183.91

- PARTYABBRE 1 171.35 189.35

Step: AIC=170.73

POSITION ~ AC\_NO + total\_assets + DIST\_NAME + AC\_TYPE + CAND\_NAME +

CAND\_SEX + PARTYABBRE + Education

Df Deviance AIC

- CAND\_NAME 1 153.62 169.62

- AC\_TYPE 1 153.71 169.71

- Education 1 154.44 170.44

<none> 152.73 170.73

- DIST\_NAME 1 154.77 170.77

- AC\_NO 1 155.03 171.03

+ liabilities 1 152.45 172.45

+ `Criminal Case` 1 152.47 172.47

+ CAND\_AGE 1 152.70 172.70

- CAND\_SEX 1 157.89 173.89

- PARTYABBRE 1 171.80 187.80

- total\_assets 1 172.53 188.53

Step: AIC=169.62

POSITION ~ AC\_NO + total\_assets + DIST\_NAME + AC\_TYPE + CAND\_SEX +

PARTYABBRE + Education

Df Deviance AIC

- AC\_TYPE 1 154.35 168.35

- DIST\_NAME 1 155.18 169.18

- AC\_NO 1 155.54 169.54

<none> 153.62 169.62

- Education 1 155.66 169.66

+ CAND\_NAME 1 152.73 170.73

+ liabilities 1 153.26 171.26

+ `Criminal Case` 1 153.47 171.47

+ CAND\_AGE 1 153.61 171.61

- CAND\_SEX 1 158.38 172.38

- PARTYABBRE 1 172.45 186.45

- total\_assets 1 173.02 187.02

Step: AIC=168.35

POSITION ~ AC\_NO + total\_assets + DIST\_NAME + CAND\_SEX + PARTYABBRE +

Education

Df Deviance AIC

- DIST\_NAME 1 155.64 167.64

- AC\_NO 1 155.82 167.82

- Education 1 156.22 168.22

<none> 154.35 168.35

+ AC\_TYPE 1 153.62 169.62

+ CAND\_NAME 1 153.71 169.71

+ liabilities 1 154.05 170.05

+ `Criminal Case` 1 154.23 170.23

+ CAND\_AGE 1 154.34 170.34

- CAND\_SEX 1 159.13 171.13

- PARTYABBRE 1 172.62 184.62

- total\_assets 1 173.22 185.22

Step: AIC=167.64

POSITION ~ AC\_NO + total\_assets + CAND\_SEX + PARTYABBRE + Education

Df Deviance AIC

- AC\_NO 1 155.84 165.84

- Education 1 157.01 167.01

<none> 155.64 167.64

+ DIST\_NAME 1 154.35 168.35

+ AC\_TYPE 1 155.18 169.18

+ CAND\_NAME 1 155.33 169.33

+ liabilities 1 155.43 169.43

+ `Criminal Case` 1 155.46 169.46

+ CAND\_AGE 1 155.62 169.62

- CAND\_SEX 1 160.29 170.29

- PARTYABBRE 1 173.39 183.39

- total\_assets 1 174.97 184.97

Step: AIC=165.84

POSITION ~ total\_assets + CAND\_SEX + PARTYABBRE + Education

Df Deviance AIC

- Education 1 157.13 165.13

<none> 155.84 165.84

+ CAND\_NAME 1 155.48 167.48

+ AC\_TYPE 1 155.55 167.55

+ AC\_NO 1 155.64 167.64

+ `Criminal Case` 1 155.67 167.67

+ liabilities 1 155.68 167.68

+ DIST\_NAME 1 155.82 167.82

+ CAND\_AGE 1 155.84 167.84

- CAND\_SEX 1 160.70 168.70

- PARTYABBRE 1 173.51 181.51

- total\_assets 1 175.21 183.21

Step: AIC=165.13

POSITION ~ total\_assets + CAND\_SEX + PARTYABBRE

Df Deviance AIC

<none> 157.13 165.13

+ Education 1 155.84 165.84

+ CAND\_NAME 1 156.55 166.55

+ liabilities 1 156.86 166.86

+ AC\_TYPE 1 156.87 166.87

+ AC\_NO 1 157.01 167.01

+ `Criminal Case` 1 157.02 167.02

+ DIST\_NAME 1 157.11 167.11

+ CAND\_AGE 1 157.12 167.12

- CAND\_SEX 1 162.47 168.47

- PARTYABBRE 1 174.23 180.23

- total\_assets 1 175.36 181.36

>

> summary(model\_2)

Call:

glm(formula = POSITION ~ total\_assets + CAND\_SEX + PARTYABBRE,

family = "binomial", data = train\_out)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.9464 -0.6187 -0.4280 -0.1722 2.4246

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -5.355e+00 2.698e+00 -1.985 0.047179 \*

total\_assets 1.282e-08 3.210e-09 3.993 6.52e-05 \*\*\*

CAND\_SEX 2.526e+00 1.321e+00 1.912 0.055858 .

PARTYABBRE -2.297e-01 6.040e-02 -3.803 0.000143 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 191.48 on 192 degrees of freedom

Residual deviance: 157.13 on 189 degrees of freedom

AIC: 165.13

Number of Fisher Scoring iterations: 5

>

> # Removing multicollinearity through VIF check

> vif(model\_2)

total\_assets CAND\_SEX PARTYABBRE

1.321650 1.254261 1.060962

>

> #Excluding cand.sex

> model\_3<- glm(formula = POSITION ~ total\_assets+PARTYABBRE , family = "binomial", data = train\_out)

>

> summary(model\_3)

Call:

glm(formula = POSITION ~ total\_assets + PARTYABBRE, family = "binomial",

data = train\_out)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.7331 -0.6130 -0.4395 -0.2031 2.4110

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -2.989e-01 4.135e-01 -0.723 0.469794

total\_assets 9.813e-09 2.697e-09 3.639 0.000274 \*\*\*

PARTYABBRE -2.237e-01 5.868e-02 -3.812 0.000138 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 191.48 on 192 degrees of freedom

Residual deviance: 162.47 on 190 degrees of freedom

AIC: 168.47

Number of Fisher Scoring iterations: 5

>

> vif(model\_3)

total\_assets PARTYABBRE

1.056807 1.056807

>

> #-----------------------------------------------------------------------------------------

> #Reading the candidates details for 2017

> cand\_list <- read\_excel("Candidates\_2017.xlsx")

> names(cand\_list)[colnames(cand\_list)=="Candidate∇"] <- "Candidate"

> cand\_list\_clean <- cand\_list[complete.cases(cand\_list),]

> cand\_list\_clean$Candidate <- tolower(cand\_list\_clean$Candidate)

>

> cand\_list\_clean$first\_name <- gsub("([A-Za-z]+).\*", "\\1", cand\_list\_clean$Candidate)

> cand\_list\_clean$Constituency <- tolower(cand\_list\_clean$Constituency)

> cand\_list\_clean$NameConst <- paste0(cand\_list\_clean$first\_name,":",cand\_list\_clean$Constituency)

>

> cand\_list\_clean$Party <- tolower(cand\_list\_clean$Party)

> cand\_list\_clean$NameParty <- paste0(cand\_list\_clean$first\_name,":",cand\_list\_clean$Party)

>

> #---------------------------------------------------------------------------------------

>

> cand\_res\_2017 <- read\_excel("LA 2017.xls", sheet = 1)

> names(cand\_res\_2017)[colnames(cand\_res\_2017) == "TOTALVALIDVOTESPOLLED"] <- "TOTVOTPOLL"

>

> #filtering for Goa

> cand\_res\_2017\_goa <- cand\_res\_2017 %>% filter(ST\_NAME == "Goa")

> cand\_test <- ads\_creation(cand\_res\_2017\_goa)

> must\_convert<-sapply(cand\_test,is.factor)

> M2<-cand\_test[,!must\_convert]

>

> test\_out<- merge(M2,mapping\_ac,by = "AC\_TYPE.x",all.x = T)

> test\_out<- merge(test\_out,mapping\_dist,by = "DIST\_NAME.x", all.x = T)

> test\_out<- merge(test\_out,mapping\_edu,by = "Education.x", all.x = T)

> test\_out<- merge(test\_out,mapping\_name,by = "CAND\_NAME.x", all.x = T)

> test\_out<- merge(test\_out,mapping\_party,by = "PARTYABBRE.x", all.x = T)

> test\_out<- merge(test\_out,mapping\_sex,by = "CAND\_SEX.x", all.x = T)

>

> test\_out <- test\_out[,!names(test\_out)%in% c("DIST\_NAME.x","AC\_TYPE.x","CAND\_NAME.x","CAND\_SEX.x","PARTYABBRE.x","Education.x")]

> test\_out <- test\_out[!is.na(test\_out$PARTYABBRE),]

>

>

> #removing totvotpoll becuase this is also a dependant variable

> test\_out <- test\_out[,!names(test\_out) %in% c("AC\_NAME","YEAR","TOTVOTPOLL")]

>

> test\_out$POSITION <- if\_else(test\_out$POSITION==1,1,0)

>

>

> test\_pred = predict(model\_3, type = "response",

+ newdata = test\_out[,!names(test\_out)%in% "POSITION"])

>

>

> # Let's see the summary

>

> summary(test\_pred)

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

0.02548 0.09065 0.12614 0.20180 0.28270 0.98772 1

>

> test\_out$prob <- test\_pred

> View(test\_out)

> # Let's use the probability cutoff of 50%.

>

> test\_pred\_churn <- factor(ifelse(test\_pred >= 0.50, "Yes", "No"))

> test\_actual\_churn <- factor(ifelse(test\_out$POSITION==1,"Yes","No"))

>

>

> table(test\_actual\_churn,test\_pred\_churn)

test\_pred\_churn

test\_actual\_churn No Yes

No 136 5

Yes 25 9

>

>

> #######################################################################

> test\_pred\_churn <- factor(ifelse(test\_pred >= 0.40, "Yes", "No"))

>

> library(e1071)

>

> test\_conf <- confusionMatrix(test\_pred\_churn, test\_actual\_churn, positive = "Yes")

> test\_conf

Confusion Matrix and Statistics

Reference

Prediction No Yes

No 127 24

Yes 14 10

Accuracy : 0.7829

95% CI : (0.7144, 0.8415)

No Information Rate : 0.8057

P-Value [Acc > NIR] : 0.8065

Kappa : 0.2193

Mcnemar's Test P-Value : 0.1443

Sensitivity : 0.29412

Specificity : 0.90071

Pos Pred Value : 0.41667

Neg Pred Value : 0.84106

Prevalence : 0.19429

Detection Rate : 0.05714

Detection Prevalence : 0.13714

Balanced Accuracy : 0.59741

'Positive' Class : Yes

> #######################################################################

>

> #########################################################################################

> # Let's Choose the cutoff value.

> #

>

> # Let's find out the optimal probalility cutoff

>

> perform\_fn <- function(cutoff)

+ {

+ predicted\_churn <- factor(ifelse(test\_pred >= cutoff, "Yes", "No"))

+ conf <- confusionMatrix(predicted\_churn, test\_actual\_churn, positive = "Yes")

+ acc <- conf$overall[1]

+ sens <- conf$byClass[1]

+ spec <- conf$byClass[2]

+ out <- t(as.matrix(c(sens, spec, acc)))

+ colnames(out) <- c("sensitivity", "specificity", "accuracy")

+ return(out)

+ }

>

> # Creating cutoff values from 0.003575 to 0.812100 for plotting and initiallizing a matrix of 100 X 3.

>

> # Summary of test probability

>

> summary(test\_pred)

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

0.02548 0.09065 0.12614 0.20180 0.28270 0.98772 1

>

> s = seq(.01,.80,length=100)

>

> OUT = matrix(0,100,3)

>

>

> for(i in 1:100)

+ {

+ OUT[i,] = perform\_fn(s[i])

+ }

Warning messages:

1: In confusionMatrix.default(predicted\_churn, test\_actual\_churn, positive = "Yes") :

Levels are not in the same order for reference and data. Refactoring data to match.

2: In confusionMatrix.default(predicted\_churn, test\_actual\_churn, positive = "Yes") :

Levels are not in the same order for reference and data. Refactoring data to match.

>

>

> plot(s, OUT[,1],xlab="Cutoff",ylab="Value",cex.lab=1.5,cex.axis=1.5,ylim=c(0,1),type="l",lwd=2,axes=FALSE,col=2)

> axis(1,seq(0,1,length=5),seq(0,1,length=5),cex.lab=1.5)

> axis(2,seq(0,1,length=5),seq(0,1,length=5),cex.lab=1.5)

> lines(s,OUT[,2],col="darkgreen",lwd=2)

> lines(s,OUT[,3],col=4,lwd=2)

> box()

> legend(0,.50,col=c(2,"darkgreen",4,"darkred"),lwd=c(2,2,2,2),c("Sensitivity","Specificity","Accuracy"))

>

>

> cutoff <- s[which(abs(OUT[,1]-OUT[,2])<0.01)]

>

>

> # Let's choose a cutoff value of 0.3132 for final model

>

> test\_cutoff\_churn <- factor(ifelse(test\_pred >=0.2, "Yes", "No"))

>

> conf\_final <- confusionMatrix(test\_cutoff\_churn, test\_actual\_churn, positive = "Yes")

>

> acc <- conf\_final$overall[1]

>

> sens <- conf\_final$byClass[1]

>

> spec <- conf\_final$byClass[2]

>

> acc

Accuracy

0.7314286

>

> sens

Sensitivity

0.6764706

>

> spec

Specificity

0.7446809

>

> View(test\_out)

>

>

>

> #----------------------------------------------------------------------------------------

> #---------------------------This code is for vote share------------------------------------

> #----------------------------------------------------------------------------------------

>

> # Lets load the library in which stepAIC function exists

> # install.packages("MASS")

> library(MASS)

>

> elec\_sum\_2012 <- read\_excel("AE2012\_8913.xls", sheet = 2)

>

> elec\_sum\_2012\_goa <- elec\_sum\_2012 %>% filter(ST\_NAME == "Goa")

>

> elec\_sum\_2012\_goa <- elec\_sum\_2012\_goa[,names(elec\_sum\_2012\_goa) %in% c("AC\_NO","TOT\_VOTERS","TOT\_ELECTORS","POLL\_PERCENT")]

>

> must\_convert<-sapply(cand\_train,is.factor)

> M2<-data.frame(sapply(cand\_train[,must\_convert],unclass))

> mapping<-cbind(cand\_train[,names(cand\_train)%in% c("DIST\_NAME.x","AC\_TYPE.x","CAND\_NAME.x","CAND\_SEX.x","PARTYABBRE.x","Education.x")],M2)

>

> train\_out<-cbind(cand\_train[,!names(cand\_train)%in% c("DIST\_NAME.x","AC\_TYPE.x","CAND\_NAME.x","CAND\_SEX.x","PARTYABBRE.x","Education.x","DIST\_NAME","AC\_TYPE","CAND\_NAME","CAND\_SEX","PARTYABBRE","Education")],M2)

>

> #removing totvotpoll becuase this is also a dependant variable

> train\_out <- train\_out[,!names(train\_out) %in% c("AC\_NAME","YEAR")]

>

> train\_out$POSITION <- if\_else(train\_out$POSITION==1,1,0)

>

> elect\_mer <- merge(train\_out,elec\_sum\_2012\_goa,by="AC\_NO")

> elect\_mer$Vote\_share <- elect\_mer$TOTVOTPOLL/elect\_mer$TOT\_VOTERS

>

> train <- elect\_mer[,!names(elect\_mer) %in% c("TOT\_ELECTORS","POLL\_PERCENT","TOTVOTPOLL","TOT\_VOTERS","POSITION")]

>

> # Build model 1 containing all variables

> model\_1 <-lm(Vote\_share~.,data=train)

> summary(model\_1)

Call:

lm(formula = Vote\_share ~ ., data = train)

Residuals:

Min 1Q Median 3Q Max

-0.34273 -0.14407 -0.06628 0.14833 0.49093

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 4.129e-02 1.803e-01 0.229 0.819098

AC\_NO 4.302e-03 2.560e-03 1.680 0.094627 .

CAND\_AGE -7.086e-04 1.404e-03 -0.505 0.614406

`Criminal Case` 2.947e-02 2.244e-02 1.313 0.190724

total\_assets 1.381e-09 3.402e-10 4.059 7.34e-05 \*\*\*

liabilities -2.914e-10 8.493e-10 -0.343 0.731914

DIST\_NAME -9.252e-02 5.763e-02 -1.605 0.110144

AC\_TYPE 3.589e-02 8.710e-02 0.412 0.680804

CAND\_NAME -1.097e-04 2.622e-04 -0.419 0.676021

CAND\_SEX 1.297e-01 6.850e-02 1.893 0.059977 .

PARTYABBRE -1.333e-02 3.547e-03 -3.758 0.000231 \*\*\*

Education -6.904e-04 4.956e-03 -0.139 0.889364

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1948 on 181 degrees of freedom

Multiple R-squared: 0.2471, Adjusted R-squared: 0.2014

F-statistic: 5.402 on 11 and 181 DF, p-value: 2.102e-07

> #######

>

> step <- stepAIC(model\_1, direction="both")

Start: AIC=-619.77

Vote\_share ~ AC\_NO + CAND\_AGE + `Criminal Case` + total\_assets +

liabilities + DIST\_NAME + AC\_TYPE + CAND\_NAME + CAND\_SEX +

PARTYABBRE + Education

Df Sum of Sq RSS AIC

- Education 1 0.00074 6.8702 -621.75

- liabilities 1 0.00447 6.8739 -621.65

- AC\_TYPE 1 0.00644 6.8759 -621.59

- CAND\_NAME 1 0.00665 6.8761 -621.59

- CAND\_AGE 1 0.00967 6.8791 -621.50

- `Criminal Case` 1 0.06547 6.9349 -619.94

<none> 6.8694 -619.77

- DIST\_NAME 1 0.09782 6.9672 -619.04

- AC\_NO 1 0.10715 6.9766 -618.79

- CAND\_SEX 1 0.13598 7.0054 -617.99

- PARTYABBRE 1 0.53601 7.4054 -607.27

- total\_assets 1 0.62516 7.4946 -604.96

Step: AIC=-621.75

Vote\_share ~ AC\_NO + CAND\_AGE + `Criminal Case` + total\_assets +

liabilities + DIST\_NAME + AC\_TYPE + CAND\_NAME + CAND\_SEX +

PARTYABBRE

Df Sum of Sq RSS AIC

- liabilities 1 0.00484 6.8750 -623.62

- AC\_TYPE 1 0.00636 6.8765 -623.57

- CAND\_NAME 1 0.00688 6.8770 -623.56

- CAND\_AGE 1 0.01020 6.8804 -623.47

- `Criminal Case` 1 0.06618 6.9363 -621.90

<none> 6.8702 -621.75

- DIST\_NAME 1 0.09726 6.9674 -621.04

- AC\_NO 1 0.10688 6.9770 -620.77

- CAND\_SEX 1 0.14224 7.0124 -619.80

+ Education 1 0.00074 6.8694 -619.77

- PARTYABBRE 1 0.53606 7.4062 -609.25

- total\_assets 1 0.62479 7.4950 -606.95

Step: AIC=-623.62

Vote\_share ~ AC\_NO + CAND\_AGE + `Criminal Case` + total\_assets +

DIST\_NAME + AC\_TYPE + CAND\_NAME + CAND\_SEX + PARTYABBRE

Df Sum of Sq RSS AIC

- AC\_TYPE 1 0.00604 6.8811 -625.45

- CAND\_NAME 1 0.00815 6.8832 -625.39

- CAND\_AGE 1 0.00820 6.8832 -625.39

- `Criminal Case` 1 0.06340 6.9384 -623.84

<none> 6.8750 -623.62

- DIST\_NAME 1 0.09556 6.9706 -622.95

- AC\_NO 1 0.10374 6.9788 -622.73

- CAND\_SEX 1 0.13817 7.0132 -621.78

+ liabilities 1 0.00484 6.8702 -621.75

+ Education 1 0.00111 6.8739 -621.65

- PARTYABBRE 1 0.54321 7.4182 -610.94

- total\_assets 1 1.01399 7.8890 -599.06

Step: AIC=-625.45

Vote\_share ~ AC\_NO + CAND\_AGE + `Criminal Case` + total\_assets +

DIST\_NAME + CAND\_NAME + CAND\_SEX + PARTYABBRE

Df Sum of Sq RSS AIC

- CAND\_NAME 1 0.00626 6.8873 -627.27

- CAND\_AGE 1 0.00796 6.8890 -627.22

- `Criminal Case` 1 0.06318 6.9442 -625.68

<none> 6.8811 -625.45

- DIST\_NAME 1 0.09004 6.9711 -624.94

- AC\_NO 1 0.09798 6.9790 -624.72

+ AC\_TYPE 1 0.00604 6.8750 -623.62

+ liabilities 1 0.00452 6.8765 -623.57

- CAND\_SEX 1 0.13983 7.0209 -623.56

+ Education 1 0.00100 6.8800 -623.47

- PARTYABBRE 1 0.53751 7.4186 -612.93

- total\_assets 1 1.00815 7.8892 -601.06

Step: AIC=-627.27

Vote\_share ~ AC\_NO + CAND\_AGE + `Criminal Case` + total\_assets +

DIST\_NAME + CAND\_SEX + PARTYABBRE

Df Sum of Sq RSS AIC

- CAND\_AGE 1 0.00854 6.8959 -629.03

- `Criminal Case` 1 0.06710 6.9544 -627.40

<none> 6.8873 -627.27

- DIST\_NAME 1 0.08472 6.9720 -626.91

- AC\_NO 1 0.09406 6.9814 -626.65

+ CAND\_NAME 1 0.00626 6.8811 -625.45

+ liabilities 1 0.00565 6.8817 -625.43

- CAND\_SEX 1 0.13912 7.0264 -625.41

+ AC\_TYPE 1 0.00415 6.8832 -625.39

+ Education 1 0.00133 6.8860 -625.31

- PARTYABBRE 1 0.53824 7.4256 -614.75

- total\_assets 1 1.00963 7.8969 -602.87

Step: AIC=-629.03

Vote\_share ~ AC\_NO + `Criminal Case` + total\_assets + DIST\_NAME +

CAND\_SEX + PARTYABBRE

Df Sum of Sq RSS AIC

<none> 6.8959 -629.03

- `Criminal Case` 1 0.07791 6.9738 -628.86

- DIST\_NAME 1 0.08738 6.9832 -628.60

- AC\_NO 1 0.09960 6.9955 -628.26

- CAND\_SEX 1 0.13463 7.0305 -627.30

+ CAND\_AGE 1 0.00854 6.8873 -627.27

+ CAND\_NAME 1 0.00685 6.8890 -627.22

+ AC\_TYPE 1 0.00387 6.8920 -627.14

+ liabilities 1 0.00349 6.8924 -627.13

+ Education 1 0.00186 6.8940 -627.08

- PARTYABBRE 1 0.52970 7.4256 -616.75

- total\_assets 1 1.02324 7.9191 -604.33

>

>

> step

Call:

lm(formula = Vote\_share ~ AC\_NO + `Criminal Case` + total\_assets +

DIST\_NAME + CAND\_SEX + PARTYABBRE, data = train)

Coefficients:

(Intercept) AC\_NO `Criminal Case` total\_assets DIST\_NAME

3.279e-02 3.878e-03 3.139e-02 1.265e-09 -8.400e-02

CAND\_SEX PARTYABBRE

1.263e-01 -1.304e-02

>

> # Let's execute this model here,

> model\_2 <- lm(formula = Vote\_share ~ AC\_NO + `Criminal Case` + total\_assets +

+ DIST\_NAME + CAND\_SEX + PARTYABBRE, data = train)

> # Let us look at the summary of the model

> summary(model\_2)

Call:

lm(formula = Vote\_share ~ AC\_NO + `Criminal Case` + total\_assets +

DIST\_NAME + CAND\_SEX + PARTYABBRE, data = train)

Residuals:

Min 1Q Median 3Q Max

-0.35514 -0.13365 -0.07192 0.14161 0.48534

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 3.279e-02 1.411e-01 0.232 0.816446

AC\_NO 3.878e-03 2.366e-03 1.639 0.102894

`Criminal Case` 3.139e-02 2.165e-02 1.450 0.148833

total\_assets 1.265e-09 2.408e-10 5.254 4.06e-07 \*\*\*

DIST\_NAME -8.400e-02 5.472e-02 -1.535 0.126426

CAND\_SEX 1.263e-01 6.629e-02 1.906 0.058243 .

PARTYABBRE -1.304e-02 3.450e-03 -3.780 0.000211 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1925 on 186 degrees of freedom

Multiple R-squared: 0.2442, Adjusted R-squared: 0.2199

F-statistic: 10.02 on 6 and 186 DF, p-value: 1.391e-09

>

>

> vif(model\_2)

AC\_NO `Criminal Case` total\_assets DIST\_NAME CAND\_SEX PARTYABBRE

3.827790 1.118231 1.214644 3.830835 1.123777 1.014189

>

> # Let's execute this model here,

> model\_3 <- lm(formula = Vote\_share ~ AC\_NO + `Criminal Case` + total\_assets +

+ + CAND\_SEX + PARTYABBRE, data = train)

> # Let us look at the summary of the model

> summary(model\_3)

Call:

lm(formula = Vote\_share ~ AC\_NO + `Criminal Case` + total\_assets +

+CAND\_SEX + PARTYABBRE, data = train)

Residuals:

Min 1Q Median 3Q Max

-0.35411 -0.13275 -0.08057 0.15273 0.49801

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -2.482e-02 1.365e-01 -0.182 0.855904

AC\_NO 7.587e-04 1.217e-03 0.624 0.533671

`Criminal Case` 2.905e-02 2.168e-02 1.340 0.181742

total\_assets 1.308e-09 2.400e-10 5.450 1.57e-07 \*\*\*

CAND\_SEX 1.259e-01 6.653e-02 1.892 0.060049 .

PARTYABBRE -1.318e-02 3.461e-03 -3.808 0.000189 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1932 on 187 degrees of freedom

Multiple R-squared: 0.2347, Adjusted R-squared: 0.2142

F-statistic: 11.47 on 5 and 187 DF, p-value: 1.145e-09

>

>

> vif(model\_3)

AC\_NO `Criminal Case` total\_assets CAND\_SEX PARTYABBRE

1.005043 1.112724 1.198097 1.123755 1.013464

>

> # Let's execute this model here,

> model\_4 <- lm(formula = Vote\_share ~ `Criminal Case` + total\_assets +

+ + CAND\_SEX + PARTYABBRE, data = train)

> # Let us look at the summary of the model

> summary(model\_4)

Call:

lm(formula = Vote\_share ~ `Criminal Case` + total\_assets + +CAND\_SEX +

PARTYABBRE, data = train)

Residuals:

Min 1Q Median 3Q Max

-0.34042 -0.12801 -0.08541 0.15737 0.50010

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -1.399e-02 1.352e-01 -0.104 0.91765

`Criminal Case` 2.835e-02 2.161e-02 1.312 0.19112

total\_assets 1.312e-09 2.395e-10 5.479 1.36e-07 \*\*\*

CAND\_SEX 1.282e-01 6.632e-02 1.932 0.05481 .

PARTYABBRE -1.321e-02 3.455e-03 -3.822 0.00018 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1929 on 188 degrees of freedom

Multiple R-squared: 0.2331, Adjusted R-squared: 0.2168

F-statistic: 14.28 on 4 and 188 DF, p-value: 3.361e-10

>

>

> vif(model\_4)

`Criminal Case` total\_assets CAND\_SEX PARTYABBRE

1.109739 1.197165 1.120323 1.013340

>

> # Let's execute this model here,

> model\_5 <- lm(formula = Vote\_share ~ total\_assets +

+ + CAND\_SEX + PARTYABBRE, data = train)

> # Let us look at the summary of the model

> summary(model\_5)

Call:

lm(formula = Vote\_share ~ total\_assets + +CAND\_SEX + PARTYABBRE,

data = train)

Residuals:

Min 1Q Median 3Q Max

-0.33106 -0.12771 -0.07763 0.15295 0.49200

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -3.099e-02 1.348e-01 -0.230 0.818

total\_assets 1.403e-09 2.297e-10 6.108 5.63e-09 \*\*\*

CAND\_SEX 1.411e-01 6.570e-02 2.148 0.033 \*

PARTYABBRE -1.370e-02 3.442e-03 -3.979 9.84e-05 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1933 on 189 degrees of freedom

Multiple R-squared: 0.2261, Adjusted R-squared: 0.2138

F-statistic: 18.4 on 3 and 189 DF, p-value: 1.62e-10

>

>

> vif(model\_5)

total\_assets CAND\_SEX PARTYABBRE

1.097094 1.095509 1.001514

>

> #------------------------------------------------------------------------------------

> elec\_sum\_2017 <- read\_excel("LA 2017.xls", sheet = 2)

>

> elec\_sum\_2017\_goa <- elec\_sum\_2017 %>% filter(ST\_NAME == "Goa")

>

> elec\_sum\_2017\_goa <- elec\_sum\_2017\_goa[,names(elec\_sum\_2017\_goa) %in% c("AC\_NO","TOTAL VOTES POLLED")]

>

> #removing totvotpoll becuase this is also a dependant variable

> must\_convert<-sapply(cand\_test,is.factor)

> M2<-cand\_test[,!must\_convert]

>

> test\_out<- merge(M2,mapping\_ac,by = "AC\_TYPE.x",all.x = T)

> test\_out<- merge(test\_out,mapping\_dist,by = "DIST\_NAME.x", all.x = T)

> test\_out<- merge(test\_out,mapping\_edu,by = "Education.x", all.x = T)

> test\_out<- merge(test\_out,mapping\_name,by = "CAND\_NAME.x", all.x = T)

> test\_out<- merge(test\_out,mapping\_party,by = "PARTYABBRE.x", all.x = T)

> test\_out<- merge(test\_out,mapping\_sex,by = "CAND\_SEX.x", all.x = T)

>

> test\_out <- test\_out[,!names(test\_out)%in% c("DIST\_NAME.x","AC\_TYPE.x","CAND\_NAME.x","CAND\_SEX.x","PARTYABBRE.x","Education.x")]

> test\_out <- test\_out[!is.na(test\_out$PARTYABBRE),]

>

> test\_out$POSITION <- if\_else(test\_out$POSITION==1,1,0)

>

> elect\_mer\_17 <- merge(test\_out,elec\_sum\_2017\_goa,by="AC\_NO")

> elect\_mer\_17$Vote\_share <- elect\_mer\_17$TOTVOTPOLL/elect\_mer\_17$`TOTAL VOTES POLLED`

>

> test <- elect\_mer\_17[,!names(elect\_mer\_17) %in% c("TOTAL VOTES POLLED","POSITION","YEAR","AC\_NAME","TOTVOTPOLL")]

> test <- test[!is.na(test$total\_assets),]

> Predict\_1 <- predict(model\_5,test[,!names(test) %in% "Vote\_share"])

> test$test\_vote\_share <- Predict\_1

>

> # Now, we need to test the r square between actual and predicted sales.

> r <- cor(test$Vote\_share,test$test\_vote\_share)

> rsquared <- cor(test$Vote\_share,test$test\_vote\_share)^2

> rsquared

[1] 0.3810812

EXPLORATORY ANALYSIS

summary(cand\_list)

Sno Candidate∇ Constituency Party Criminal Case

Min. : 1 Length:426 Length:426 Length:426 Min. :0.0000

1st Qu.: 54 Class :character Class :character Class :character 1st Qu.:0.0000

Median :107 Mode :character Mode :character Mode :character Median :0.0000

Mean :107 Mean :0.2535

3rd Qu.:160 3rd Qu.:0.0000

Max. :213 Max. :4.0000

NA's :213 NA's :213

Education Total Assets Liabilities

Length:426 Length:426 Length:426

Class :character Class :character Class :character

Mode :character Mode :character Mode :character

> names(cand\_list)

[1] "Sno" "Candidate∇" "Constituency" "Party" "Criminal Case"

[6] "Education" "Total Assets" "Liabilities"

> dim(cand\_list)

[1] 426 8

summary(cand\_res\_2017)

ST\_CODE ST\_NAME MONTH YEAR DIST\_NAME

Length:7942 Length:7942 Min. :3 Min. :2017 Length:7942

Class :character Class :character 1st Qu.:3 1st Qu.:2017 Class :character

Mode :character Mode :character Median :3 Median :2017 Mode :character

Mean :3 Mean :2017

3rd Qu.:3 3rd Qu.:2017

Max. :3 Max. :2017

AC\_NO AC\_NAME AC\_TYPE CAND\_NAME CAND\_SEX

Min. : 1.0 Length:7942 Length:7942 Length:7942 Length:7942

1st Qu.: 43.0 Class :character Class :character Class :character Class :character

Median :112.5 Mode :character Mode :character Mode :character Mode :character

Mean :153.3

3rd Qu.:260.0

Max. :403.0

CAND\_CATEGORY CAND\_AGE PARTYABBRE TOTALVALIDVOTESPOLLED POSITION

Length:7942 Length:7942 Length:7942 Min. : 20 Min. : 1.000

Class :character Class :character Class :character 1st Qu.: 465 1st Qu.: 3.000

Mode :character Mode :character Mode :character Median : 1026 Median : 6.000

Mean : 14009 Mean : 7.019

3rd Qu.: 9606 3rd Qu.:10.000

Max. :262741 Max. :30.000

> names(cand\_res\_2017)

[1] "ST\_CODE" "ST\_NAME" "MONTH"

[4] "YEAR" "DIST\_NAME" "AC\_NO"

[7] "AC\_NAME" "AC\_TYPE" "CAND\_NAME"

[10] "CAND\_SEX" "CAND\_CATEGORY" "CAND\_AGE"

[13] "PARTYABBRE" "TOTALVALIDVOTESPOLLED" "POSITION"

> dim(cand\_res\_2017)

[1] 7942 15

summary(elec\_sum\_2012)

ST\_CODE ST\_NAME AC\_NO AC\_NAME MONTH

Length:940 Length:940 Min. : 1.0 Length:940 Min. : 1.000

Class :character Class :character 1st Qu.: 34.0 Class :character 1st Qu.: 1.000

Mode :character Mode :character Median : 78.0 Mode :character Median : 1.000

Mean :119.6 Mean : 3.926

3rd Qu.:175.2 3rd Qu.:12.000

Max. :403.0 Max. :12.000

YEAR TOT\_VOTERS TOT\_ELECTORS POLL\_PERCENT

Min. :2012 Min. : 8050 Min. : 17409 Min. :40.90

1st Qu.:2012 1st Qu.: 84573 1st Qu.:119299 1st Qu.:59.81

Median :2012 Median :151841 Median :216225 Median :67.70

Mean :2012 Mean :135251 Mean :209699 Mean :68.28

3rd Qu.:2012 3rd Qu.:184180 3rd Qu.:305798 3rd Qu.:75.96

Max. :2012 Max. :333657 Max. :676637 Max. :96.02

> names(elec\_sum\_2012)

[1] "ST\_CODE" "ST\_NAME" "AC\_NO" "AC\_NAME" "MONTH" "YEAR"

[7] "TOT\_VOTERS" "TOT\_ELECTORS" "POLL\_PERCENT"

> dim(elec\_sum\_2012)

[1] 940 9