```
> polling = read.csv("D:/Projects/Assam_custom_data.csv")
> str(polling)           #structure of the custommade Dataset
'data.frame':   373 obs. of  7 variables:
 $ Constituency   : chr  "Abhayapuri North" "Abhayapuri North" "Abhayapuri South (sc)" "Abhaya
puri South (sc)" ...
 $ Year           : int  2006 2011 2006 2011 2006 2011 2016 2006 2011 2016 ...
 $ TimesNowVMR    : int  11 21 NA 16 5 5 8 1 6 NA ...
 $ IndiatodayAxis : int  18 25 NA NA 15 NA NA 5 NA NA ...
 $ CountDifference: int  5 5 1 6 8 9 4 8 5 2 ...
 $ propBJP        : num  1 1 1 1 1 ...
 $ BJP            : int  1 1 1 1 1 1 1 1 1 1 ...
>                       #BJP is the dependent variable
>                       #TimesNowVMR,IndiaTodayAxis,CountDifference
>                       #and propBJP are Independent Variables
> summary(polling)      #to get a count on the number of missing values
 Constituency           Year       TimesNowVMR      IndiatodayAxis
 Length:373         Min.   :2006   Min.   :-28.000   Min.   :-33.0000
 Class :character   1st Qu.:2006   1st Qu.: -6.750   1st Qu.:-11.0000
 Mode  :character   Median :2011   Median :  1.000   Median : -2.0000
                    Mean   :2011   Mean   :  1.053   Mean   : -0.8964
                    3rd Qu.:2016   3rd Qu.:  7.750   3rd Qu.:  8.0000
                    Max.   :2016   Max.   : 34.000   Max.   : 30.0000
                                   NA's   :91        NA's   :180
 CountDifference      propBJP           BJP
 Min.   :-19.000   Min.   :0.0000   Min.   :0.0000
 1st Qu.: -6.000   1st Qu.:0.0000   1st Qu.:0.0000
 Median :  1.000   Median :0.6000   Median :1.0000
 Mean   : -1.319   Mean   :0.5117   Mean   :0.5013
 3rd Qu.:  3.000   3rd Qu.:1.0000   3rd Qu.:1.0000
 Max.   : 11.000   Max.   :1.0000   Max.   :1.0000
 NA's   :50        NA's   :1
> library("mice")       #to perform MULTIPLE IMPUTATION ON THE MISSING DATA

Attaching package: ‘mice’

The following object is masked from ‘package:stats’:

    filter

The following objects are masked from ‘package:base’:

    cbind, rbind

> simple=polling[c("TimesNowVMR","IndiatodayAxis","CountDifference","propBJP")]   #For multipl
e Imputation to be useful
>                                                                                 #we have to
find the missing variables
>                                                                                 #without usi
ng the outcome of BJP so we
>                                                                                 #limit our d
ataframe and create a new
>                                                                                 #dataframe t
o just 4 polling related
>                                                                                 #variables.
> summary(simple)       #now we observe we have smaller no of variables in total
  TimesNowVMR      IndiatodayAxis     CountDifference      propBJP
 Min.   :-28.000   Min.   :-33.0000   Min.   :-19.000   Min.   :0.0000
 1st Qu.: -6.750   1st Qu.:-11.0000   1st Qu.: -6.000   1st Qu.:0.0000
 Median :  1.000   Median : -2.0000   Median :  1.000   Median :0.6000
 Mean   :  1.053   Mean   : -0.8964   Mean   : -1.319   Mean   :0.5117
 3rd Qu.:  7.750   3rd Qu.:  8.0000   3rd Qu.:  3.000   3rd Qu.:1.0000
 Max.   : 34.000   Max.   : 30.0000   Max.   : 11.000   Max.   :1.0000
 NA's   :91        NA's   :180        NA's   :50        NA's   :1
>
> set.seed(100) # to get same value from Multiple Imputation we fix seed to a same value
> imputed= impute.me((simple))
Error in impute.me((simple)) : could not find function "impute.me"
> imputed= complete(mice(simple))

 iter imp variable
  1   1  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  1   2  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  1   3  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  1   4  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
```

```
  1   5  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  2   1  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  2   2  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  2   3  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  2   4  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  2   5  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  3   1  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  3   2  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  3   3  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  3   4  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  3   5  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  4   1  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  4   2  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  4   3  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  4   4  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  4   5  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  5   1  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  5   2  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  5   3  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  5   4  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
  5   5  TimesNowVMR  IndiatodayAxis  CountDifference  propBJP
> summary(imputed)
  TimesNowVMR      IndiatodayAxis     CountDifference      propBJP
 Min.   :-28.000   Min.   :-33.0000   Min.   :-19.000   Min.   :0.000
 1st Qu.: -6.000   1st Qu.:-12.0000   1st Qu.: -5.000   1st Qu.:0.000
 Median :  1.000   Median : -2.0000   Median :  1.000   Median :0.600
 Mean   :  1.043   Mean   : -0.6944   Mean   : -1.198   Mean   :0.513
 3rd Qu.:  9.000   3rd Qu.:  8.0000   3rd Qu.:  3.000   3rd Qu.:1.000
 Max.   : 34.000   Max.   : 30.0000   Max.   : 11.000   Max.   :1.000
> polling$TimesNowVMR=imputed$TimesNowVMR
> polling$IndiatodayAxis=imputed$IndiatodayAxis   #copying imputed data into the dataframe
> summary(polling)      #checking original dataframe for missing values
 Constituency            Year       TimesNowVMR      IndiatodayAxis     CountDifference      p
ropBJP            BJP
 Length:373         Min.   :2006   Min.   :-28.000   Min.   :-33.0000   Min.   :-19.000   Min.
   :0.0000   Min.   :0.0000
 Class :character   1st Qu.:2006   1st Qu.: -6.000   1st Qu.:-12.0000   1st Qu.: -6.000   1st
Qu.:0.0000   1st Qu.:0.0000
 Mode  :character   Median :2011   Median :  1.000   Median : -2.0000   Median :  1.000   Medi
an :0.6000   Median :1.0000
                    Mean   :2011   Mean   :  1.043   Mean   : -0.6944   Mean   : -1.319   Mean
   :0.5117   Mean   :0.5013
                    3rd Qu.:2016   3rd Qu.:  9.000   3rd Qu.:  8.0000   3rd Qu.:  3.000   3rd
Qu.:1.0000   3rd Qu.:1.0000
                    Max.   :2016   Max.   : 34.000   Max.   : 30.0000   Max.   : 11.000   Max.
   :1.0000   Max.   :1.0000
                                                                        NA's   :50        NA's
   :1
> polling$CountDifference=imputed$CountDifference
> polling$propBJP=imputed$propBJP
> summary(polling)      #Now we have successfully eliminated all missing values from all inpde
pendent variables
 Constituency            Year       TimesNowVMR      IndiatodayAxis     CountDifference      p
ropBJP           BJP
 Length:373         Min.   :2006   Min.   :-28.000   Min.   :-33.0000   Min.   :-19.000   Min.
   :0.000   Min.   :0.0000
 Class :character   1st Qu.:2006   1st Qu.: -6.000   1st Qu.:-12.0000   1st Qu.: -5.000   1st
Qu.:0.000   1st Qu.:0.0000
 Mode  :character   Median :2011   Median :  1.000   Median : -2.0000   Median :  1.000   Medi
an :0.600   Median :1.0000
                    Mean   :2011   Mean   :  1.043   Mean   : -0.6944   Mean   : -1.198   Mean
   :0.513   Mean   :0.5013
                    3rd Qu.:2016   3rd Qu.:  9.000   3rd Qu.:  8.0000   3rd Qu.:  3.000   3rd
Qu.:1.000   3rd Qu.:1.0000
                    Max.   :2016   Max.   : 34.000   Max.   : 30.0000   Max.   : 11.000   Max.
   :1.000   Max.   :1.0000
> #BUILDING MODELS
> #We will split the data into a training and testing set.
> #Data from 2006 and 2011 will be used for training and data from 2016 will be used for testi
ng.
> Train=subset(polling,Year==2006|Year==2011)   #creating a dataframe having the statistics of
 only 2006,2011 data
> Test=subset(polling,Year==2016)   #creating a dataframe having the statistics of only 2016 d
ata
```

```
> #Understanding Prediction of our Baseline Model
>
> table(Train$BJP)

  0   1
125 130
> # in the 255 training observations it predicts in 125 AIUDF+Congress won a particular consti
tuency and in BJP 130 won so it always predicts victory of BJP even for a majority AIUDF+Congr
ess Alliance constituency
> # in the 255 training observations it predicts in 125 AIUDF+Congress won a particular consti
tuency and in BJP 130 won so it always predicts victory of BJP even for a majority AIUDF+Congr
ess Alliance constituency

> # accuracy on training set is 50.9% so we conclude it is a very weak model
> #CREATING A SMARTER BASELINE MODEL TO COMPARE WITH THE LOGISTIC REGRESSION MODEL
>
> table(sign(Train$TimesNowVMR))

 -1   0   1
113  14 128
> # It shows on 113 observations AIUDF alliance wins & BJP Alliance wins 128 and on 14 observa
tions it is not sure of outcome.
>
> #comparison of smarter baseline model with the outcome of the training data
>
> table(Train$BJP,sign(Train$TimesNowVMR))

    -1  0  1
  0 74  7 44
  1 39  7 84
> #in 74 obsevations correct prediction of win of AIUDF
> #in 84 obsevations correct prediction of win of BJP
> #14 observations not sure
> # 44 observation where predicted BJP+ wins but AIUDF+ won
> # 39 observation where predicted AIUDF+ wins but BJP+ won
> # SO CONCLUSION IS THAT THIS MODEL IS BETTER THAN NAIVE BASELINE MODEL WHICH DID 125 ERRORS
IN COMPARISON TO(44+39)=83 INCORRECT PREDICTIONS IN SMARTER BASELINE METHOD
>
> # CREATING A LOGISTIC REGRESSION BASED MODEL
>
> #TO Compute MULTICOLLINEARITY(relationship between independent variables)
>
> cor(Train[c("TimesNowVMR","IndiatodayAxis","CountDifference","propBJP","BJP")])
                TimesNowVMR IndiatodayAxis CountDifference   propBJP       BJP
TimesNowVMR       1.0000000      0.1110427       0.2500535 0.2243130 0.2338315
IndiatodayAxis    0.1110427      1.0000000       0.3968285 0.1213623 0.1942504
CountDifference   0.2500535      0.3968285       1.0000000 0.2772816 0.2918842
propBJP           0.2243130      0.1213623       0.2772816 1.0000000 0.4426514
BJP               0.2338315      0.1942504       0.2918842 0.4426514 1.0000000
>
> #CREATING A LOGISTIC REGRESSION MODEL WITH ON;Y ONE VARIABLE
>
> #We choose a variable with highest correlation with dependent variable with BJP which is pro
pBJP in our case(0.4426514)
>
> mod1=glm(BJP~propBJP, data=Train,family="binomial")
> summary(mod1)

Call:
glm(formula = BJP ~ propBJP, family = "binomial", data = Train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6339  -0.7760   0.7816   0.7816   1.6415

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.0461     0.2129  -4.914 8.91e-07 ***
propBJP       2.0754     0.3064   6.773 1.26e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)
```

```
    Null deviance: 353.41  on 254  degrees of freedom
Residual deviance: 301.45  on 253  degrees of freedom
AIC: 305.45

Number of Fisher Scoring iterations: 4

> # since we just took dummy data in the dataset  the AIC strength value is high .Stars indica
te the significance of variable propBJP
>
> #PREDICTION OF mod1 model ON Training set
>
> pred1=predict(mod1,type="response")
> table(Train$BJP,pred1>=0.5) #threshold is 0.5 if >0.5 it is 1 (BJP wins) else 0 (AIUDF wins)

    FALSE TRUE
  0    87   38
  1    34   96
> # this model makes 34+38=72 mistakes so it is better than smarter baseline model which made
83 mistakes
>
> #TO IMPROVE FURTHER ON THIS MODEL WE CAN ADD ANOTHER VARIABLE
>
> #Idea is to use pair of dependent variables with least correlation  so we can use Indiatoday
Axis and TimesNowVMR as one pair(cor=0.1110427) and
> # propBJP and IndiatodayAxis as the second pair(cor=0.1213623)
>
> mod2=glm(BJP~propBJP+In, data=Train,family="binomial")

> mod2=glm(BJP~TimesNowVMR+IndiaTodayAxis, data=Train,family="binomial")
Error in eval(predvars, data, env) : object 'IndiaTodayAxis' not found
> mod2=glm(BJP~TimesNowVMR+IndiatodayAxis, data=Train,family="binomial")
> pred1=predict(mod2,type="response")
> table(Train$BJP,pred1>=0.5) #threshold is 0.5 if >0.5 it is 1 (BJP wins) else 0 (AIUDF wins)

    FALSE TRUE
  0    78   47
  1    48   82
> #SINCE it gives 48+47=95 errors which is more than 83 errors in one variable model was BETTE
R
> summary(mod2)

Call:
glm(formula = BJP ~ TimesNowVMR + IndiatodayAxis, family = "binomial",
    data = Train)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.992  -1.123   0.537   1.109   1.720

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)    0.012755   0.130981   0.097 0.922425
TimesNowVMR    0.039302   0.011528   3.409 0.000651 ***
IndiatodayAxis 0.027088   0.009712   2.789 0.005286 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 353.41  on 254  degrees of freedom
Residual deviance: 330.90  on 252  degrees of freedom
AIC: 336.9

Number of Fisher Scoring iterations: 4

> ##conclusion can also be verified as mod1 had 3+3=6 stars(more significant) but mod2 has onl
y 3+2=5 stars
>
> #computing outcome of simple baseline model on the testing set
>
> table(Test$BJP, sign(Test$TimesNowVMR))

    -1  0  1
  0 31  3 27
```

```
  1 21  2 34
> # Correctly predicted 31 times AIUDF wins
> # Correctly predicted 34 times BJP wins
> # Incorrectly predicted 27 times BJP wins when AIUDF won
> # Incorrectly predicted 2 times AIUDF wins when BJP won
> # NOT sure about 21 observations
>
> #computing outcome of mod2 2 Variable model  on the testing set
>
> TestPrediction=predict(mod2,newdata=Test,type="response")
> table(Test$BJP, TestPrediction>=0.5)

    FALSE TRUE
  0    36   25
  1    23   34
> #mistakes=23+25=48 errors
>
> #PULLING OUT WHAT THE MISTAKES WERE
>
> subset(Test, TestPrediction>=0.5 & BJP==0)
       Constituency Year TimesNowVMR IndiatodayAxis CountDifference    propBJP BJP
16          Baghbar 2016          34             -2              -5 0.30769231   0
42           Behali 2016          10             -8              -2 0.40000000   0
84           Chenga 2016          -2              8              -8 0.00000000   0
108          Dholai 2016          10             -7              -4 0.00000000   0
114       Dibrugarh 2016          -2              8              -2 0.00000000   0
143   Goalpara East 2016           0             22              -8 0.00000000   0
171           Hojai 2016          12             21               4 0.00000000   0
207  Karimgan South 2016           5             18               3 0.66666667   0
210 Karimganj North 2016           2              8               3 1.00000000   0
213       Katigorah 2016         -10             18               5 0.00000000   0
216     Katlicherra 2016          -8             22              -6 1.00000000   0
237         Lumding 2016           1             21               3 1.00000000   0
261           Moran 2016           1              0               2 1.00000000   0
264        Morigaon 2016           5              8               3 0.00000000   0
270         Nalbari 2016           6             -4              -8 0.00000000   0
292     Patharkandi 2016          11            -11               3 0.08333333   0
298       Rangapara 2016          12            -14              -4 1.00000000   0
315        Samaguri 2016           9             -9               3 0.00000000   0
321      Sarupathar 2016          19             -2              -4 0.00000000   0
327           Sidli 2016          14              0               1 0.50000000   0
336           Sonai 2016          10              3              -8 0.00000000   0
348        Tamulpur 2016           2              0               1 1.00000000   0
354          Tezpur 2016          19             24               5 0.00000000   0
360       Tingkhong 2016           6             21               1 0.08333333   0
366         Titabar 2016           6             22               1 1.00000000   0
> #THESE ARE THE 25 MISTAKES WHEN AIUDF WON BUT PREDICTION WAS BJP
>
> subset(Test, TestPrediction<=0.5 & BJP==1)
      Constituency Year TimesNowVMR IndiatodayAxis CountDifference   propBJP BJP
27       Barhampur 2016         -16              8               4 1.0000000   1
78   Chapaguri(st) 2016          -5             -7               2 1.0000000   1
126     Doom Dooma 2016         -13              8               4 1.0000000   1
180       Jaleswar 2016          -8            -11              -5 1.0000000   1
198       Kaliabor 2016          -5             -3              -5 0.0000000   1
201      Kaliagaon 2016          -5             -1               3 0.0000000   1
204       Kamalpur 2016          -6            -13               5 0.0000000   1
222 Kokrajhar East 2016          -5             -2              -2 0.0000000   1
225     Laharighat 2016          -5             -4              -2 0.0000000   1
228         Lahowal 2016           3            -16              -7 1.0000000   1
234        Lakhipur 2016         -16            -16               8 1.0000000   1
243          Majbat 2016         -13            -13               9 1.0000000   1
249  Mangaldoi(sc) 2016           1            -24              -8 1.0000000   1
255     Margherita 2016           1            -16              -6 1.0000000   1
275         Nawgong 2016          -8              0             -13 1.0000000   1
281      Palasbari 2016         -19             21               1 0.0000000   1
289  Patacharkuchi 2016         -19             -8               5 0.0000000   1
301         Rangiya 2016          10            -16              -8 1.0000000   1
324       Sibsagar 2016           6            -14              -4 0.3333333   1
330        Silchar 2016           0            -29              -4 0.1000000   1
351           Teok 2016           4             -7               2 1.0000000   1
363       Tinsukia 2016          -8            -14             -15 0.1428571   1
372      Udharbond 2016           2             -4               1 1.0000000   1
> #THESE ARE THE 23 MISTAKES WHEN BJP WON BUT PREDICTION WAS AIUDF
```

```
>
> #TESTING THE MODEL mod1 WHICH HAD LEAST TRAINING ERRORS OF ALL MODELS
>
> TestPrediction=predict(mod1,newdata=Test,type="response")
> table(Test$BJP, TestPrediction>=0.5)

    FALSE TRUE
  0    43   18
  1    18   39
> #mistakes=18+18=36 errors
> #PULLING OUT WHAT THE MISTAKES WERE
> subset(Test, TestPrediction>=0.5 & BJP==0)
                Constituency Year TimesNowVMR IndiatodayAxis CountDifference   propBJP BJP
24                 Barchalla 2016          -5              0               6 0.6666667   0
154                 Golaghat 2016          -4             -8              -2 1.0000000   0
157                Golakganj 2016          -7            -11              -4 1.0000000   0
160 Gossaigaon Kokrajhar West 2016         -15            -30              -8 1.0000000   0
183                Jalukbari 2016          -2            -10              -8 1.0000000   0
189                    Jania 2016           9            -33               4 1.0000000   0
207           Karimgan South 2016           5             18               3 0.6666667   0
210          Karimganj North 2016           2              8               3 1.0000000   0
216              Katlicherra 2016          -8             22              -6 1.0000000   0
237                  Lumding 2016           1             21               3 1.0000000   0
240                 Mahmara 2016          -4             -7               2 1.0000000   0
261                   Moran 2016           1              0               2 1.0000000   0
267               Naharkatia 2016         -10              0               4 0.6250000   0
278                   Nazira 2016           0            -15              -2 1.0000000   0
298                Rangapara 2016          12            -14              -4 1.0000000   0
345                  Sorbhog 2016           4             -7               2 1.0000000   0
348                 Tamulpur 2016           2              0               1 1.0000000   0
366                  Titabar 2016           6             22               1 1.0000000   0
> #THESE ARE THE 18 MISTAKES WHEN AIUDF WON BUT PREDICTION WAS BJP
>
> subset(Test, TestPrediction<=0.5 & BJP==1)
      Constituency Year TimesNowVMR IndiatodayAxis CountDifference    propBJP BJP
168           Hajo 2016          12              0               3 0.00000000   1
177   Jagiroad(sc) 2016          34             -2              -6 0.14285714   1
192      Jonai(st) 2016          10              2               2 0.00000000   1
195         Jorhat 2016          12             30               1 0.00000000   1
198        Kaliabor 2016          -5             -3              -5 0.00000000   1
201       Kaliagaon 2016          -5             -1               3 0.00000000   1
204        Kamalpur 2016          -6            -13               5 0.00000000   1
222 Kokrajhar East 2016          -5             -2              -2 0.00000000   1
225     Laharighat 2016          -5             -4              -2 0.00000000   1
281       Palasbari 2016         -19             21               1 0.00000000   1
284          Panery 2016           1             16               1 0.00000000   1
289  Patacharkuchi 2016         -19             -8               5 0.00000000   1
309          Sadiya 2016          -4              7               3 0.09090909   1
312  Salmara South 2016         -11             21               5 0.00000000   1
324        Sibsagar 2016           6            -14              -4 0.33333333   1
330         Silchar 2016           0            -29              -4 0.10000000   1
357          Thowra 2016          10             -4               2 0.00000000   1
363        Tinsukia 2016          -8            -14             -15 0.14285714   1
> #THESE ARE THE 18 MISTAKES WHEN BJP WON BUT PREDICTION WAS AIUDF
>
> #SO WE CONCLUDE THAT FOR OUR DATASET THE ONE VARIABLE MODEL "mod1" IS THE BEST MODEL FOR TES
T SET ELECTION PREDICTION
>
> #THANKYOU.
```