Stanford STATS 202

PANSS Score and Flagged Assessments Predictor

Class Project for STATS202

Singh, Mandeep

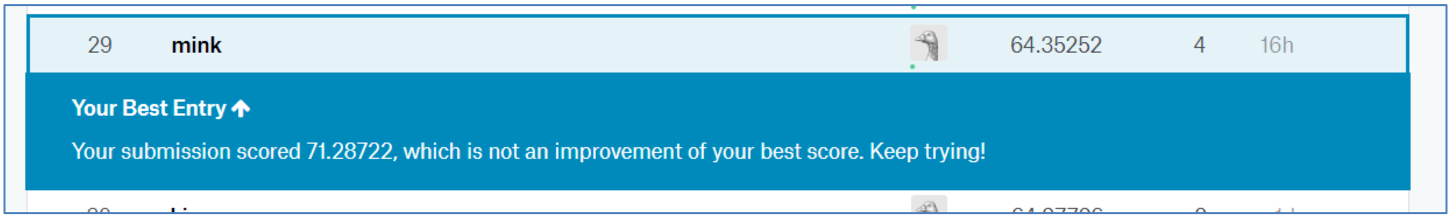
msingh13@stanford.edu

Aug 14, 2019

TABLE OF CONTENTS:

Kaggle Status Objective 3

Predict the 18th week PANSS score



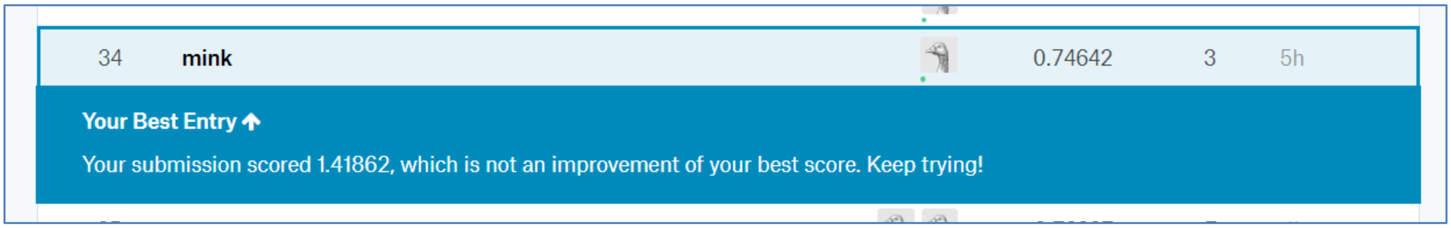
29	mink	64.35252	4	16h

Your Best Entry ↑

Your submission scored 71.28722, which is not an improvement of your best score. Keep trying!

Kaggle status objective 4

Predict flagged assessments



34	mink	0.74642	3	5h

Your Best Entry ↑

Your submission scored 1.41862, which is not an improvement of your best score. Keep trying!

OBJECTIVE 1: In this objective we have to demonstrate with the given data whether there is treatment effect on schizophrenia. We are given "Treatment" and "Control" group based on the PANSS score of a given patient on a "VisitDay". From the data using Program 1, we obtained average of PANSS score per day for Treatment & Control group. We then plot average PANSS score per visit day for both the groups as show in **FIG1a**. We see that at VisitDay 0 the average of PANSS score for Control and Treatment group is same. As the number of visit days increases we see the average PANSS score for both the treatment and control group decreases and comes closer to minimum PANSS score we can obtain. The average minimum PANSS score observed in the plot is 36. This data indicates there is treatment effect. **FIG1b** was obtained for LeadStatus of Passed to rule out data incorrectness or outliers. We don't see significant difference in the plot. There were few outlier observations in **FIG1a** which we don't see in **FIG1b**.

FIG2-FIG6 are the density functions for PANSS_Score, Total P Score, Total N Score, Total G Score and Composite Score respectively per treatment group. The plots are normally distributed with skewedness observed in PANSS_Score, Total P Score and Total G Score plots. As FIG2-FIG6 are observed to be normally distributed, it indicates the data is typical continua and the measurement is amenable to parametric statistical treatment [1].

HYPOTHESIS TESTING

Alternative hypothesis is there is treatment effect i.e. PANSS Total score from "Treatment" group reflects treatment effect. Null hypothesis is there is no treatment effect evident from the "Treatment" group data. For this we create Average PANSS Total from "Treatment" group per VisitDay and Average PANSS Total from "Control" group per VisitDay collected in "Treatment_and_Control_PANSS_Total_data.csv" file and we regress PANSS Total from Treatment with PANSS Total from Control using **PROGRAM 3.** We obtain **TABLE 1**. From the data in **TABLE 1**, the p-value is much less than 0.05 so we reject NULL hypothesis which means data from "Treatment" group is not by simply chance. It should be noted that patient is randomly categorized to "Control" or "Treatment" group by the patient evaluator. This indicates treatment effect with statistical proof.

OBJECTIVE 2:

Patient data at day 0 was filtered from all the studies A-D. There were duplicate entries found that were removed.

There are 30 variables contributing to PANSS Total score. These 30 variables can be summarized to 3 categories namely PANSS Positive Scale, PANSS Negative Scale and PANSS General Psychopathology Scale. As stakeholders wants to understand the distribution prior to the treatment so any experimental outcome variables can be excluded for e.g. TxGroup can be excluded. LeadStatus may or may not be considered, it will give information about how much successful is the study on each patient. We will exclude LeadStatus as it will not contribute in understanding different types of patient, it only tells about the level of assessment. Country variable will give demographic information. Site ID & RaterID can be excluded as it does not tell about type of patient with schizophrenia.

With **OBJECTIVE 2 PROGRAM 1** for day 0 visit all PANSS positive scores were added, all PANSS negative scores were added & all General scores were added. **K-means clustering algorithm** was selected for categorization. For optimal cluster number, **NbClust()** function was used which runs 30 indices on the input data to recommend optimal cluster number. The output of NbClust() is summarized in **TABLE2.** 12 indices recommend cluster size of 2 while 3 indices recommend cluster size of 3.

K-MEANS CLUSTERING

Kmeans clustering algorithm was run with cluster number as 2 and nstart as 20. Few iterations were done by varying start from 1 to 50 and total within cluster sum of squares were observed which we should minimize. **OBJECTIVE2 FIG1** plot is

obtained with 2 cluster numbers. From this figure, it can be concluded that the clusters are separated at a line with TOTAL G Score of ~ 45. Half of patients lie above the TotalGScore of ~45 and the other half are below. 2D plot of same data is plotted as shown in **OBJECTIVE 2 FIG2** for better interpretation.
As some of the indices in NBClust() reported cluster size of 3, **FIG 3 & 4** was obtained for k-means with cluster size of 3. From FIG3 we can see the data is divided into 3 clusters with Total G scores 16-30(blue), 30-50(red) and 50(green).

HIERARCHICAL CLUSTERING

As the number of optimal clusters is not clear with k-means so hierarchical clustering technique is explored. Hierarchical clustering does not require that we commit to a particular choice of K. Complete linkage plot is obtained for 3 data columns namely Total P Score, Total N Score and Total G Score. From **FIG5** we can cut tree with 3 groups by visual inspection.
With k-means, as it is vague what optimal number of groups is, it purely depends upon what inference we want to bring from the number groups and accordingly try to analyze the k-means cluster output with that many number of groups.

<u>OBJECTIVE3</u>

Failed Approach:

We have to predict total PANSS score for each patient in study E at the 18th week. For clarification as the data contains VisitDay so we will assume the prediction is expected at the end of 18th week i.e. at the end of 125th day (the VisitDay starts from 0 so 126 - 0).

Parameters to exclude for model creation (Data Transformations):

For model simplification the SiteID & AssessmentID can be omitted from model creation as this does not affect PANSS score. RaterID may play some role in total PANSS score and highlight evaluations done by particular rater or evaluator but considering this component to be small so we will exclude it for simplicity. The 'Study' variable may or may not be excluded.
We will have to create continent variable as in the data for which we have to predict contains 'UK' which is not in the input data. When PANSS_Total is regressed wrt continent variable it is not statistically significant variable. So for simplicity UK was renamed to France as both the countries are geographically close.
As we have to predict based on time so we will have to include interaction terms in the data wrt VisitDay. Mistakenly assumed the VisitDay parameter tracks time but it is not the only principle component and at any given VisitDay by patient it is same for that day.
So in this approach the covariates are P1-P7, N1-N7, G1-G16, VisitDay, Country, TxGroup and dependent variable is PANSS_Total. With the data from Study A-D the linear model was created with all the covariates and their interaction with VisitDay to account for time. The data from study A –D was split ~80% for training and 20% for testing the model. The model was accurate in predicting the PANSS total for training data, test data and the data from study E but only for 18th week day (126 day). i.e. if the 18th week data covariates are present. In study E, for 18th week P1-P7, N1-N7, G1-G16 were obtained by mean of these covariates by patient for the entire trial. Approach outlined in [2] using 'mice' library was explored to come up with the missing P1-P7,N1-N7,G1-G16 values for the 18th week to predict for 18th week PANSS score. The predicted PANSS was not correlating with different days for any given patient. Which lead to taking a step back and explore the missing component explained in next section.

Successful Approach:

The main component missing in previous approach was failing to track PANSS score wrt time. This is done by lag variables. Lead variables are also used. This approach takes into account scenarios where patient's 18th week visit can be at 200th day of the trial date. Key point to note is VisitDay tracks the day from when the trial starts and is not attached to a patient. Lag variable to track PANSS score wrt previous 2 visits were taken into account. Also difference in number of days from current visit to last visit and last to last visit was tracked. Similarly lead variables were also created that tracks PANSS score of next visit and next to next visit along with difference in number of days wrt current visit. VisitNumber variable was also created. Min, max and average PANSS for given patient through entire trial was also created. All this data transformation was created per patient. All study data from study A through D was clubbed together into 1 file named “Study_ABCD.csv”. P1-P7, N1-N7 and G1-G16 was not considered in the model as it is linear combination to PANSS. Study Name, Patient ID, Assessment ID, Rater ID were also not considered in the model creation. Only LeadStatus with “Passed” entries were considered in model creation but LeadStatus variable was not used in the model. There was ERROR entry in Country column which were also deleted for model creation. Programmatic code details can be seen in **OBJECTIVE 3 PROGRAM 1.**

First linear model was created as outlined in **OBJECTIVE 3 PROGRAM2.** PANSS Total is regressed wrt all the covariates outlined in data transformation in previous paragraph. Wanted to consider the higher order terms in VisitWeek along with its interaction with other covariates in the model hence the exponential and logarithmic variables of VisitWeek. The training data was split between 0.9:0.1 ratios. The reason for this was to consider all the levels of Country variable and VisitNumber as there were entries, in test set and the set on which we have to predict, that were not in training set. The mean squared error (MSE) for training set was **8.551817e-25** and MSE for test set was **8.704427e-25.** From **TABLE3** it is evident that Train set MSE is always lower than Test set MSE.

Before prediction, there was data transformation done to Study E data to make it match to the data transformations done to study A thru D prior to model creation. PatientIDs for which 18th week input data (covariates) was not present in the data set (Study_E), it was created by looking at previous visit or next visit. Code details can be seen in **OBJECTIVE 3 PROGRAM3.**

Bagging, random forest and boosting approaches were also explored. Bagging is special case of a random forest with m=p. In our case total number of predictors are 18. Bagging program can be seen in **OBJECTIVE3 PROGRAM4. OBJECTIVE3 FIG1** shows MSE wrt number of trees. The variable importance chart was also generated as showing in FIG2. This was generated with “varImpPlot” function. From the variable importance we see that NextVisitPANSS and PreviousVisitPANSS are significant. At the time of creation of this **FIG1,** in order to get more accuracy next to next visits PANSS was also obtained in the input data transformation. Training MSE of 0.1182304 and Test MSE of 1.34499 was obtained with bagging approach. For random forest approach we set mtry to 18/3 i.e. 6, where 18 are the total number of predictors in our model. Training MSE of 0.2747294 and Test MSE of 2.088028 was obtained with random forest model. Complete program for random forest is in **OBJECTIVE3 PROGRAM5.**

Lastly boosting approach is explored. The number of trees used in boosting approach is 4000. The interaction depth is set to 2. The smaller the interaction depth the model will learn slowly. We have to aim for having a week learner in the boosting approach. The shrinkage parameter used was 0.25118. It was obtained using **OBJECTIVE3 PROGRAM7** The boosting approach program is in **OBJECTIVE3 PROGRAM6.** Iterations were carried out to obtain shrinkage parameter that results in least train and test MSE. **OBJECTIVE3 FIG3 and FIG4** are the test and train mse wrt varying shrinkage parameter. Training MSE of 0.02681 and test MSE of 0.1835 was obtained using boosting approach.

TEST MSE and TRAIN MSE for all the approaches explored is summarized in TABLE 5. Data from TABLE 5 reveals the linear model train and test mse are lowest.

OBJECTIVE4

In objective 4 we have to classify weather a given assessment done falls under LeadStatus of “Passed” or “Flagged or Assigned to CS” category.
Data transformations that where done in this case are similar to the data transformations done in objective3. In addition to variables transformed in objective 3, for objective 4 the “LeadStatus” was transformed to have 2 levels. 0 for “Passed” and 1 for “Flagged or Assigned to CS”. This is tracked under “FlagOrACSLeadStatus”. From study A - D input data, next 2 visits LeadStatus and last 2 visits LeadStatus was tracked. We have no information of LeadStatus for the data(study E) on which we have to predict the probability. On the study E data we create the LeadStatus for next 2 visits and last 2 visits randomly (i.e. 0 or 1) as an initial state. Another data transformation done is, PANSS Total score for that Visit is predicted using the model created in OBJECTIVE3. Difference of actual PANSS vs predicted PANSS is used in the model. For detailed code of data transformation refer to **OBJECTIVE 4 PROGRAM1**

We find correlation of all the covariates. Correlation information is summarized in OBJECTIVE4 FIG1. All variables that has high correlation should not be used in the interaction term in the classification model. Next we fit “FlagOrACSLeadStatus” status wrt the covariates in **generalized linear model** with family type “binomial”. **OBJECTIVE 4 PROGRAM 2** is for model fit and creating ROC curve for the model. ROC curve in **OBJECTIVE4 FIG2** reveals test ROC is more or less following train ROC.

Data transformation prior to probability prediction was done in same way as done before model creation. As outlined in first paragraph of this objective, the LeadStatus for last 2 and next 2 visits was randomly formed to have a starting point. Also the PANSS Total for given assessment was predicted using model in OBJECTIVE3 and difference variable was created. For detailed code of data transformation refer to **OBJECTIVE 4 PROGRAM3.**

Next we explore classification or probability prediction using Support Vector Machine algorithm. Same data transformations were used in this approach as used in glm above. **OBJECTIVE 4 PROGRAM4** plots ROC on train and test data as shown in **OBJECTIVE4 FIG3.** The SVM model has cost value of 0.1 and linear kernel. A smaller value of cost parameter means we obtain large number of support vectors, because the margin is wider. We perform cross-validation to tune the svm model i.e. come up with optimal value of cost. From **TABLE6** the cost parameter for best model with lowest cross validation error is 0.15. **TABLE7** outlines summary of bestmodel.
With **OBJECTIVE 4 PROGRAM5** we do data transformations on study E and predict probability with SVM model.

Looking at ROC curves for both glm and svm models**, svm model predicts better** than glm model.

OBJECTIVE4 FIG4 and OBJECTIVE4 FIG5 were generated with the model that did not considered lag and lead variables for LeadStatus. This immediately indicated we need to consider lag and lead variables for LeadStatus probability prediction. These ROC curves also imply that Training ROC curve may appear promising but model correctness is revealed by Test data ROC curve.

Future work post project submission.
The svm algorithm prediction is better compared to glm but need to optimize or explore the svm model by using kernel=’radial’ and varying gamma parameter. The optimization for cost function or gamma value identification takes long runtimes and compute. Need to explore if there is another important covariate that is time dependent and should be considered in the model for better accuracy.

APPENDIX: FIGURES

Objective 1:

FIG 1(a):

Average PANSS score per visit day per treatment group.

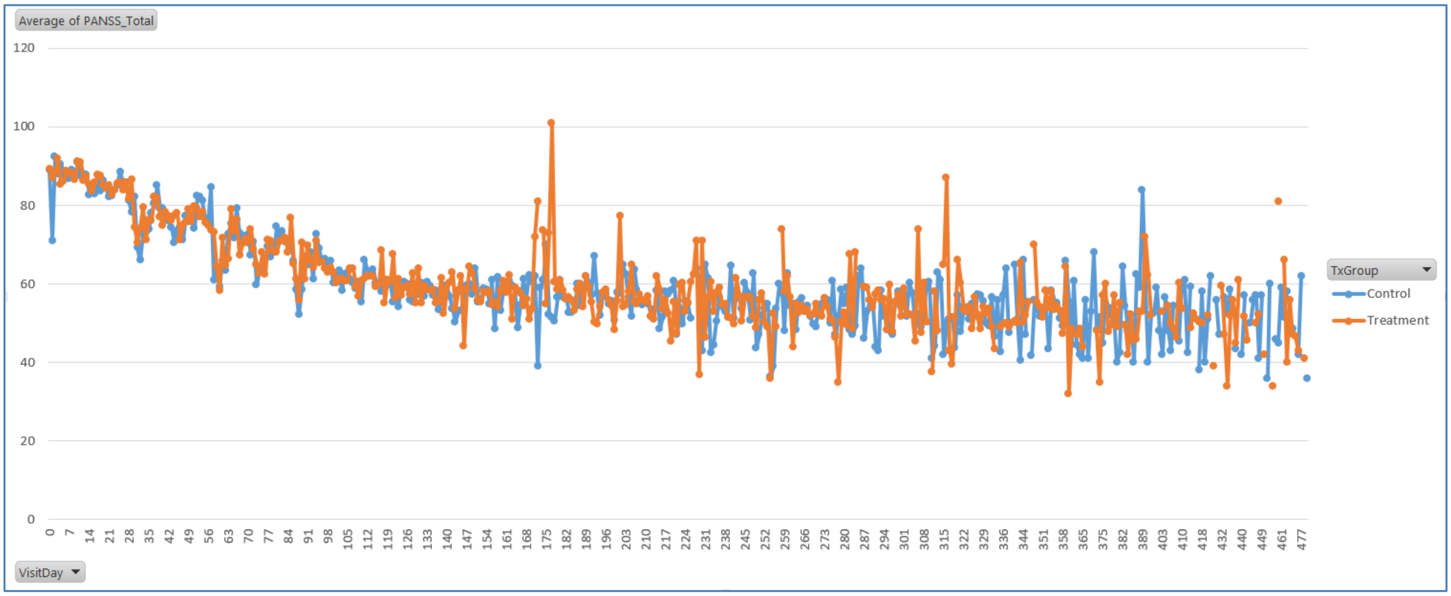


FIG 1(b)

Average PANSS score per visit day per treatment group with Lead Status as Passed.

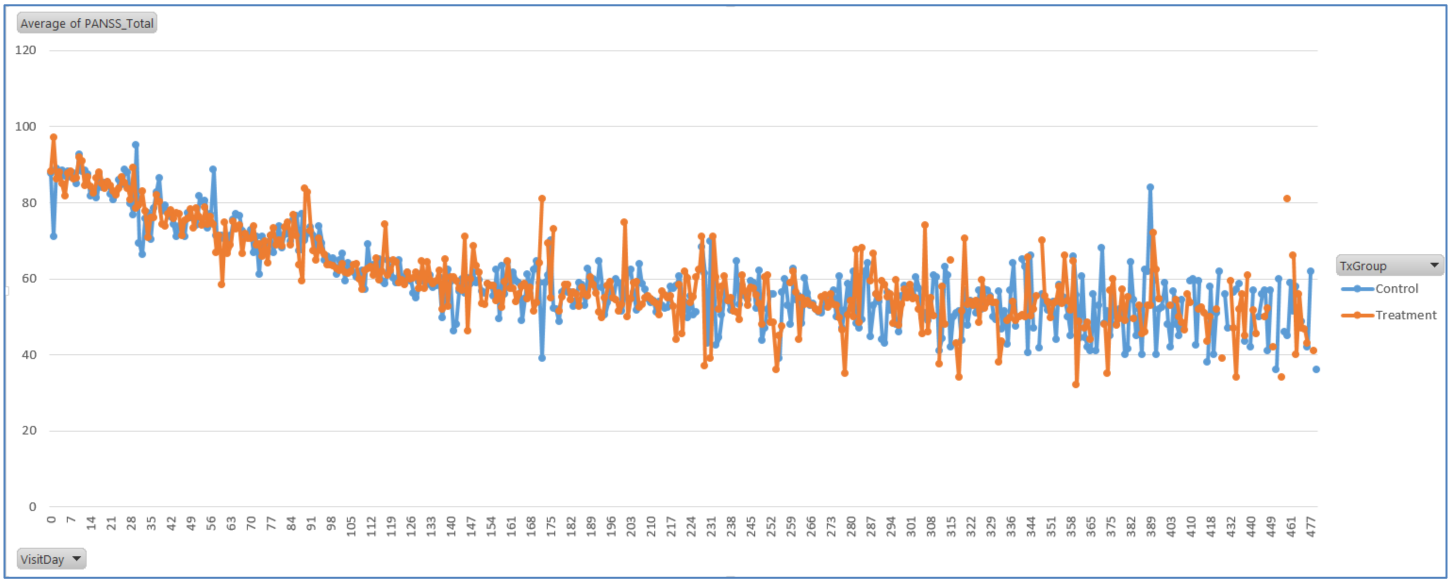


FIG2:

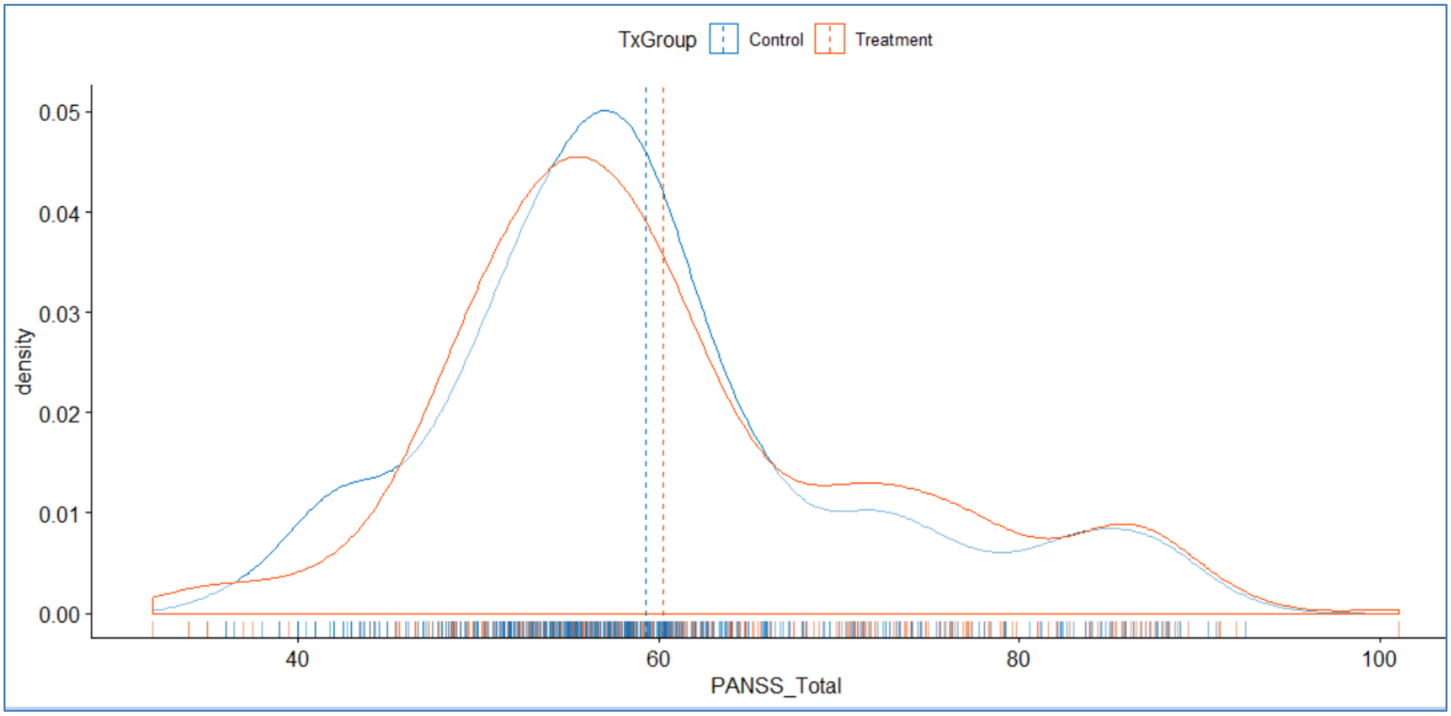

TxGroup
Control
Treatment
0.05
0.04
0.03
0.02
0.01
0.00
density
40
60
80
100
PANSS_Total

FIG3

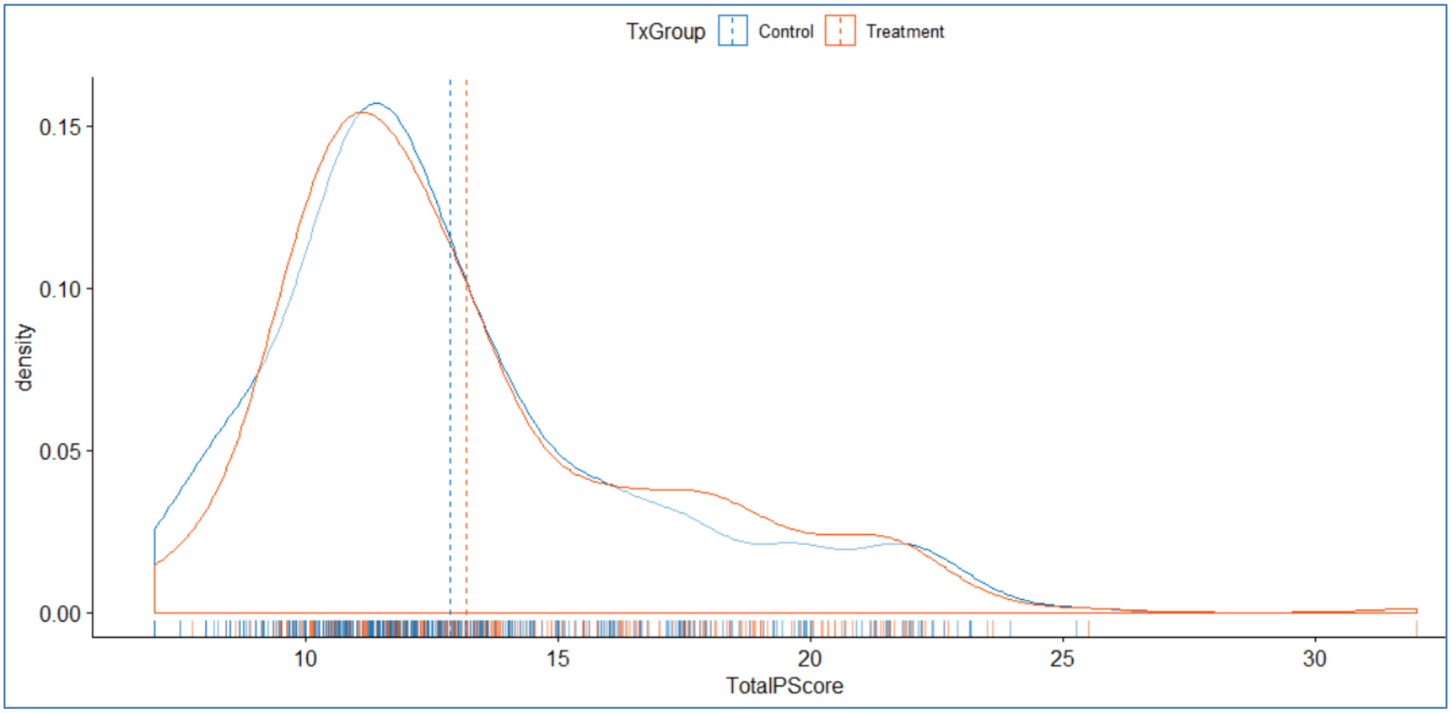

TxGroup
Control
Treatment
0.15
0.10
0.05
0.00
density
10
15
20
25
30
TotalPScore

FIG4

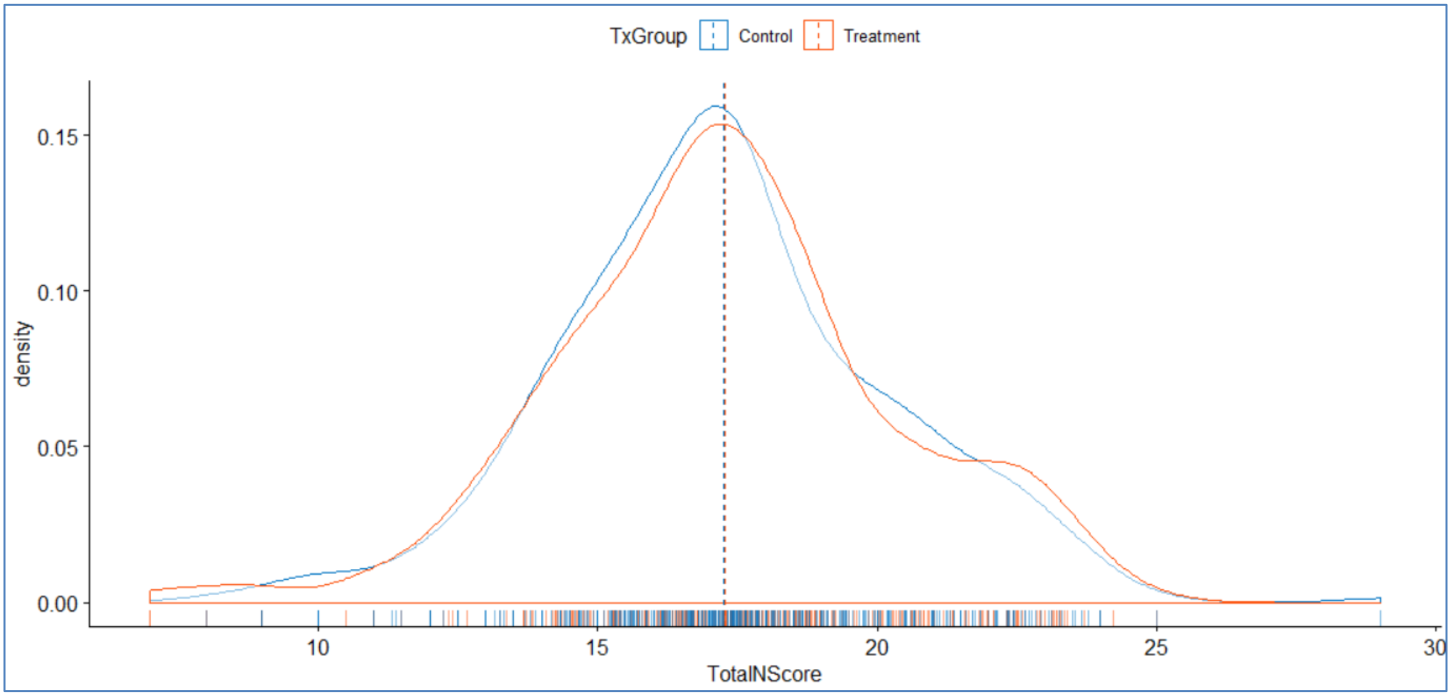

TxGroup
Control
Treatment
density
0.00
0.05
0.10
0.15
10
15
20
25
30
TotalNScore

FIG5

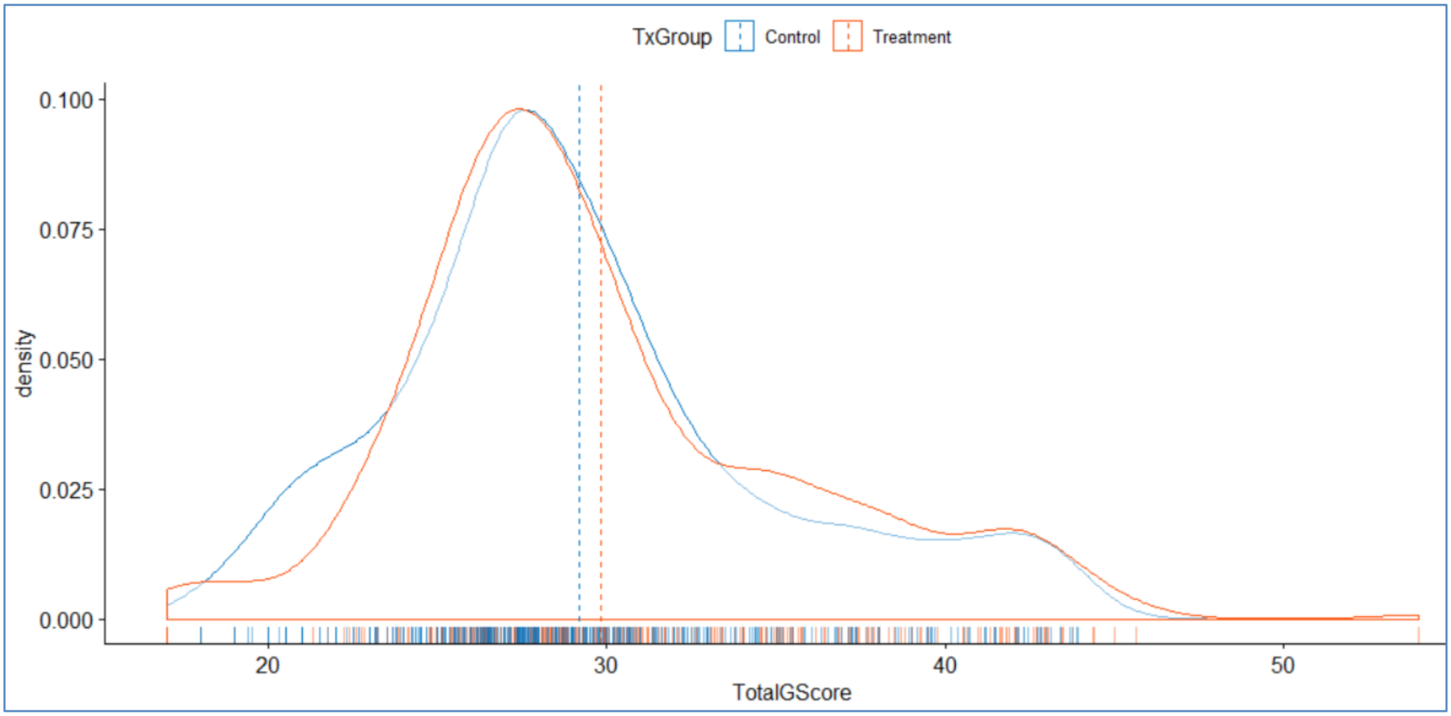
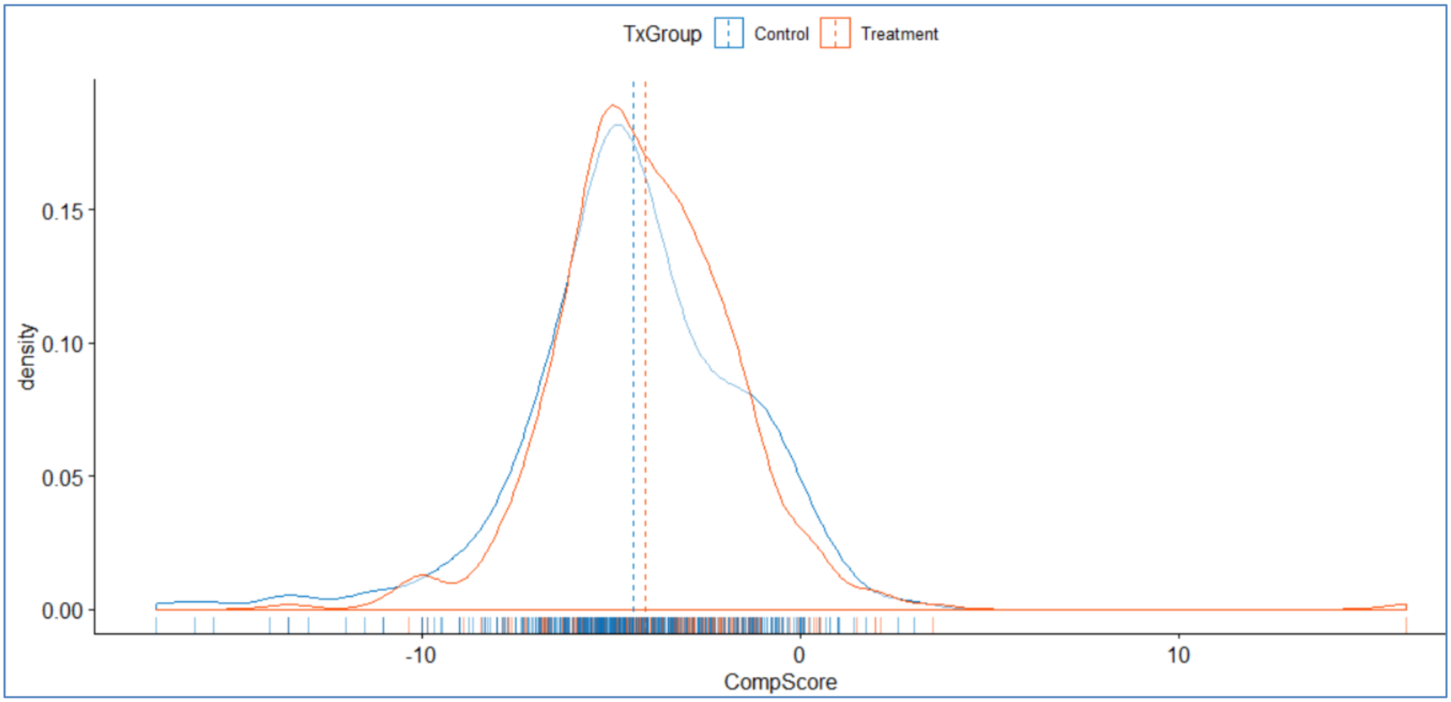

TxGroup
Control
Treatment
density
0.000
0.025
0.050
0.075
0.100
20
30
40
50
TotalGScore

FIG6


TxGroup
Control
Treatment
density
0.15
0.10
0.05
0.00
-10
0
10
CompScore

OBJECTIVE 2

FIG 1: 3D Plot of k-means clustering with k=2.

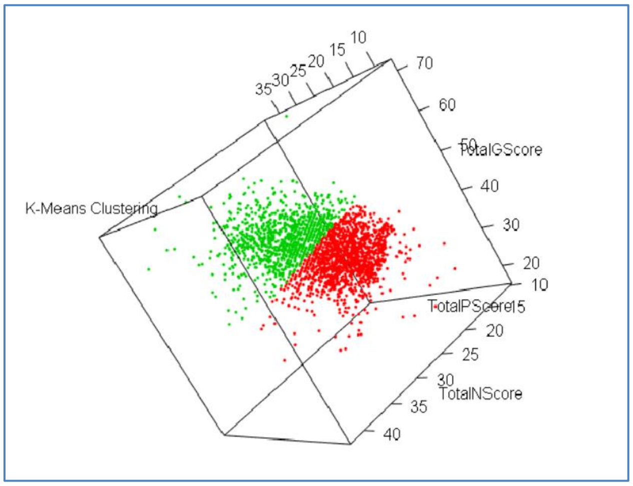


FIG2: 2D Plot of k-means clustering with k=2.

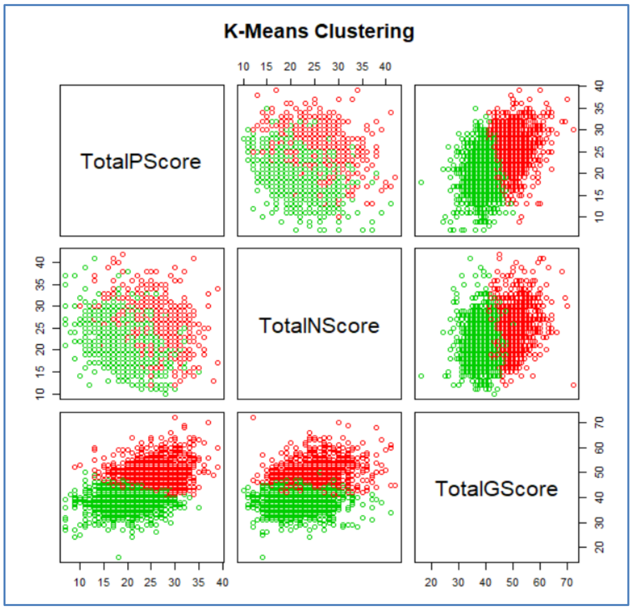


FIG3:

3D Plot of k-means clustering with k=3.

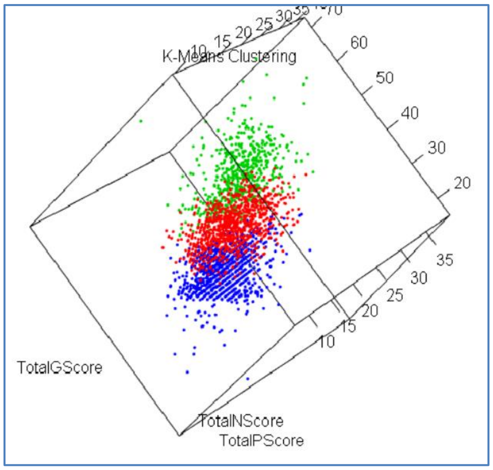


FIG4: 2D Plot of k-means clustering with k=3

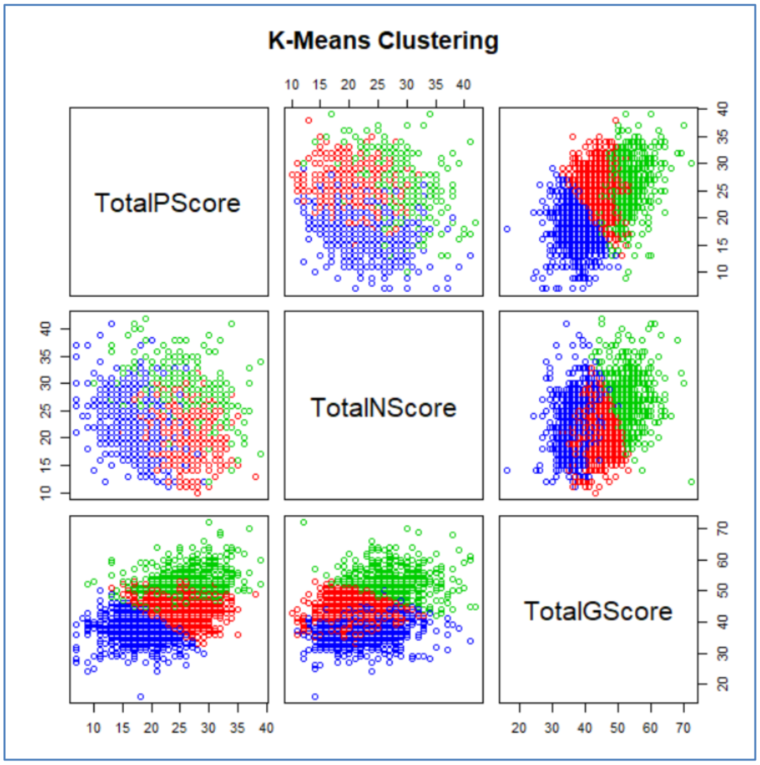
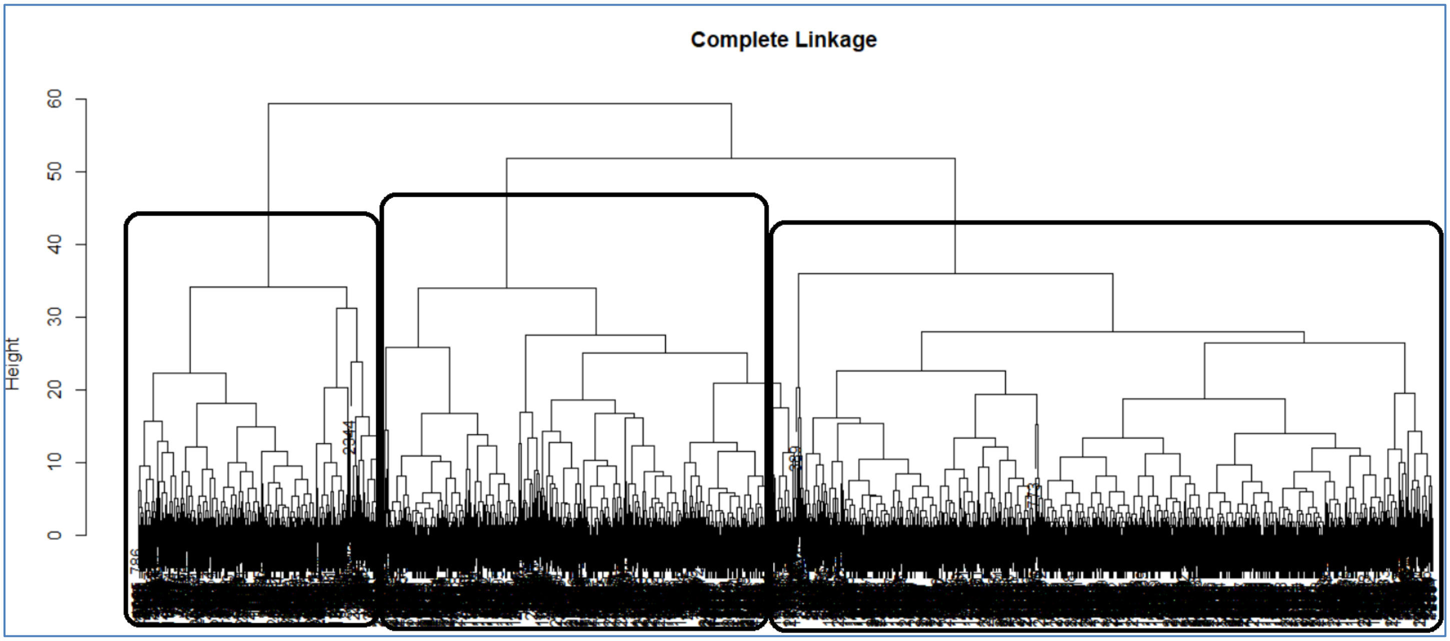


FIG5



OBJECTIVE 3:

FIG1

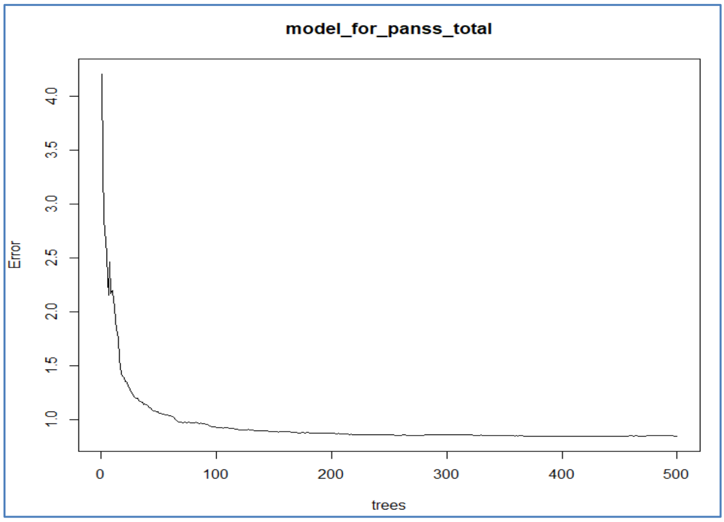


FIG2

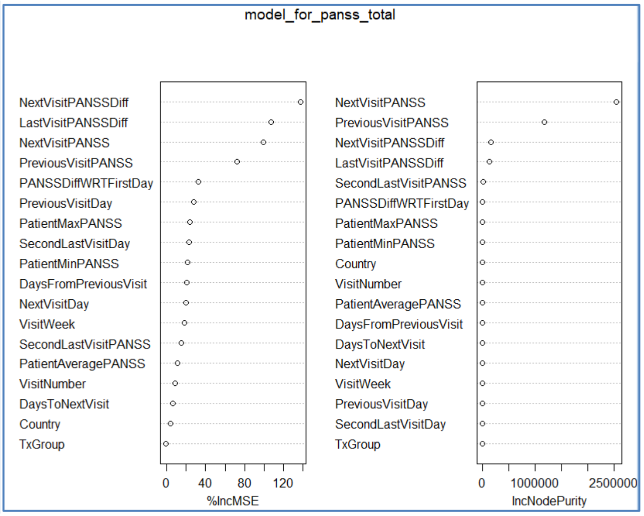


FIG3

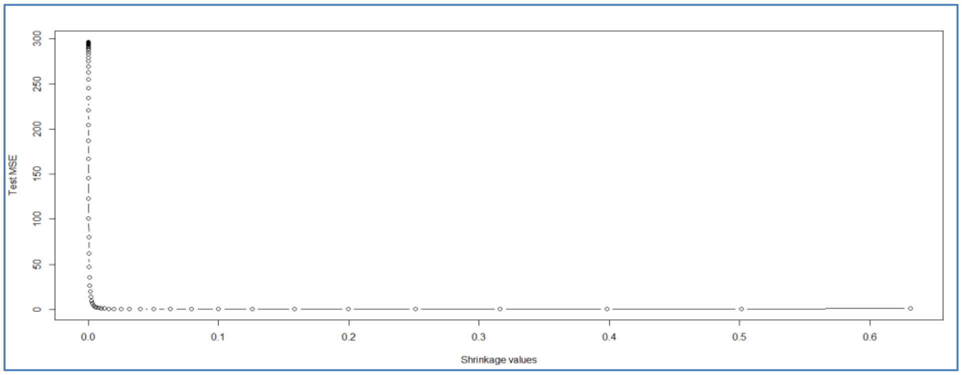
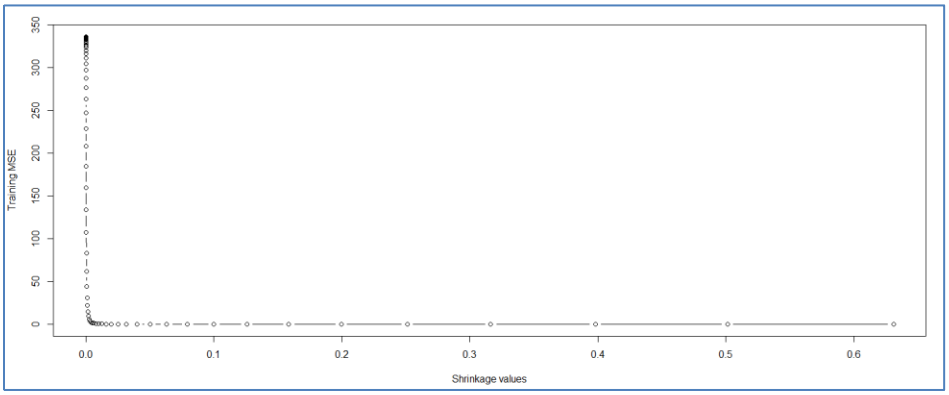

Test MSE
0
50
100
150
200
250
300
0.0
0.1
0.2
0.3
0.4
0.5
0.6
Shrinkage values

FIG4


Training MSE
0
50
100
150
200
250
300
350
0.0
0.1
0.2
0.3
0.4
0.5
0.6
Shrinkage values

OBJECTIVE4

FIG1

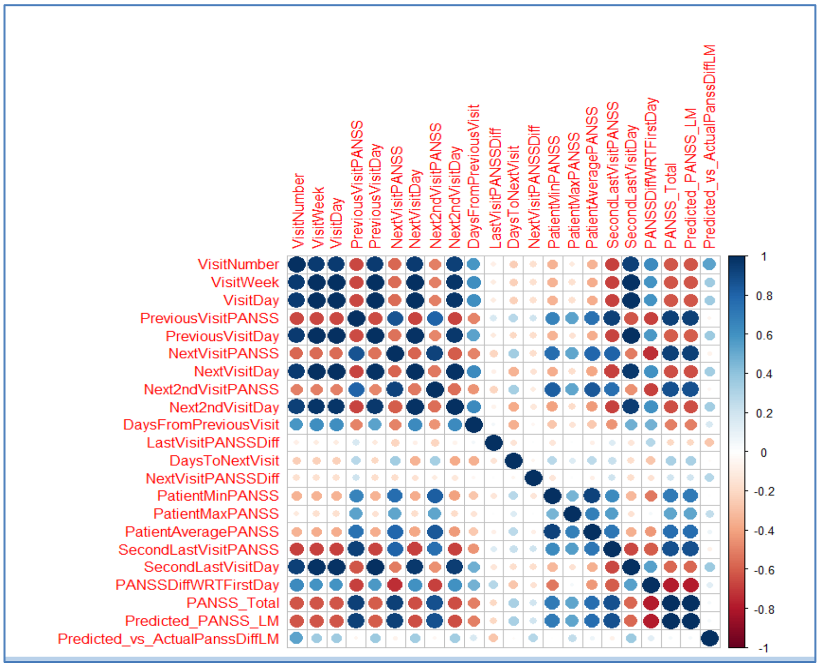


FIG2

GLM train and test ROC. Train ROC is the smooth curve.

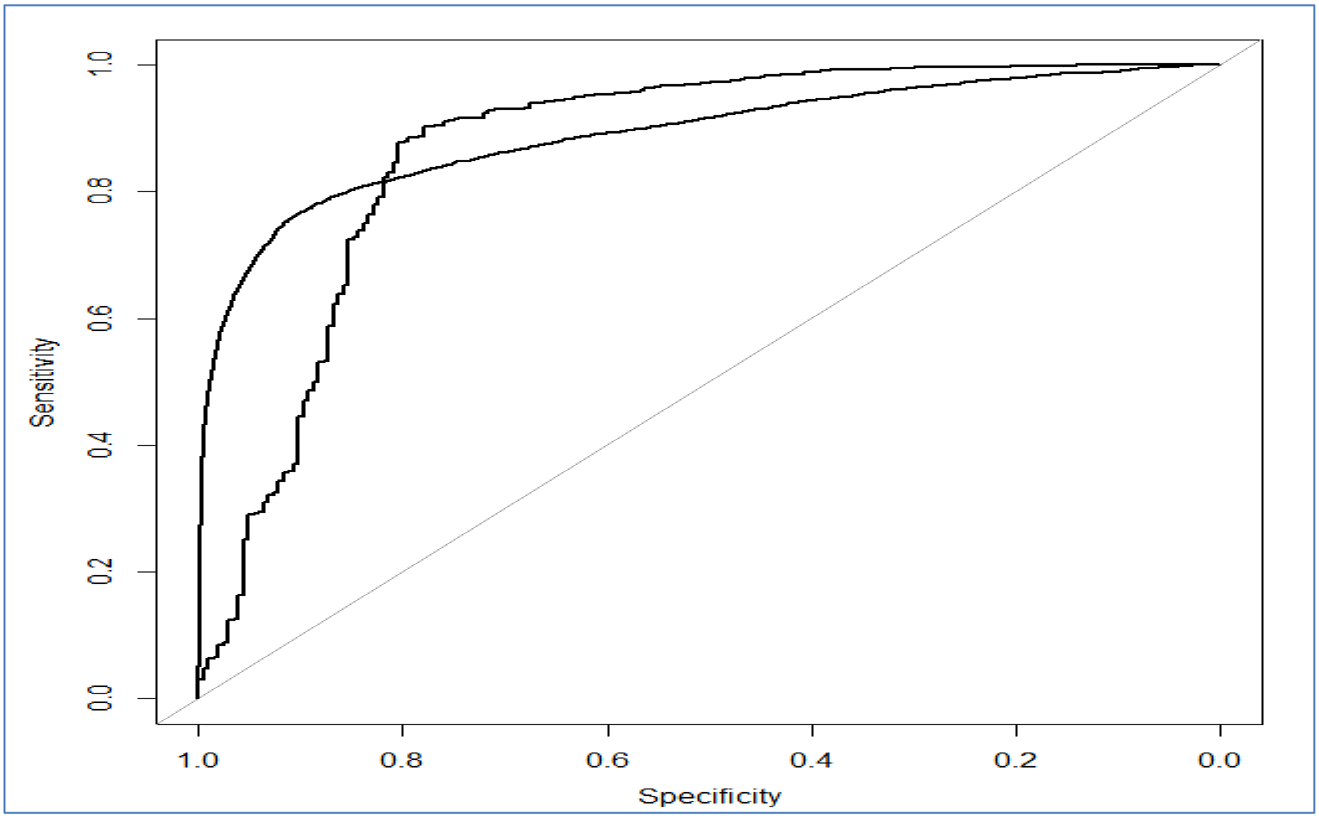


FIG3

SVM ROC curve with linear kernel and cost of 0.1

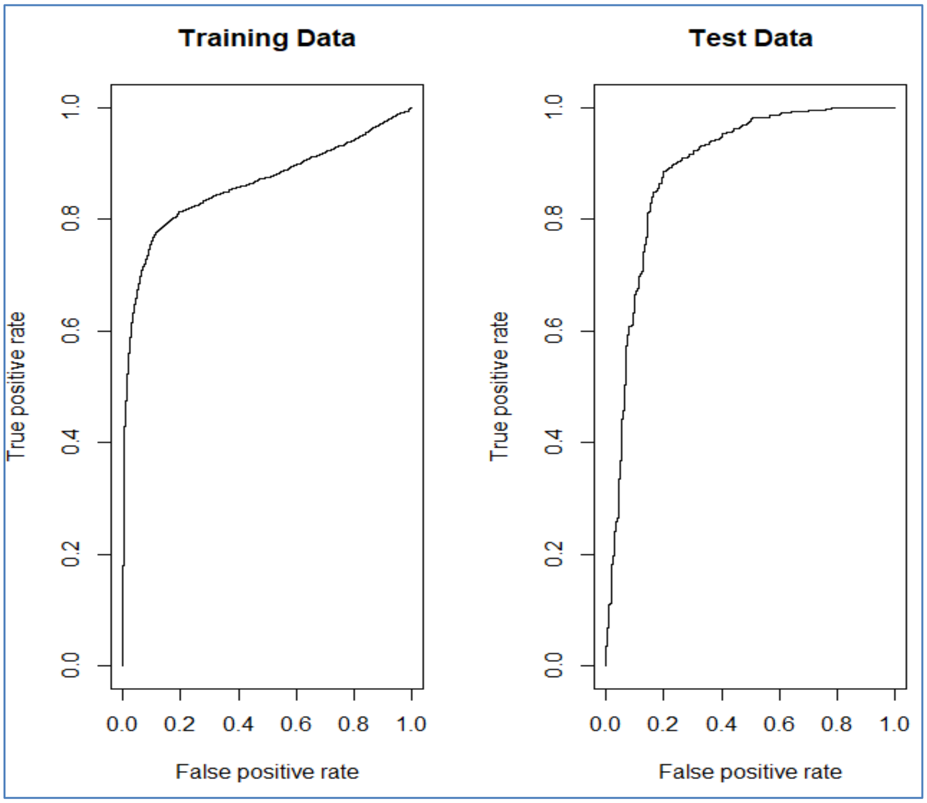


FIG4

SVM ROC curve with no lead or lag LeadStatus variables in the model. The Test ROC curve is linear and reveals the model will predict incorrectly. This also reveals Training ROC curve can be misleading.

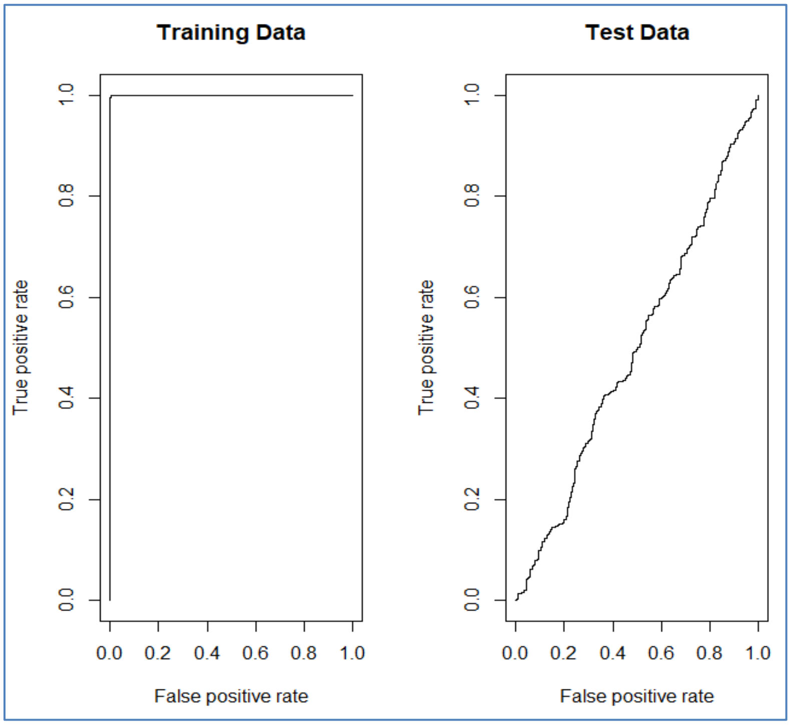
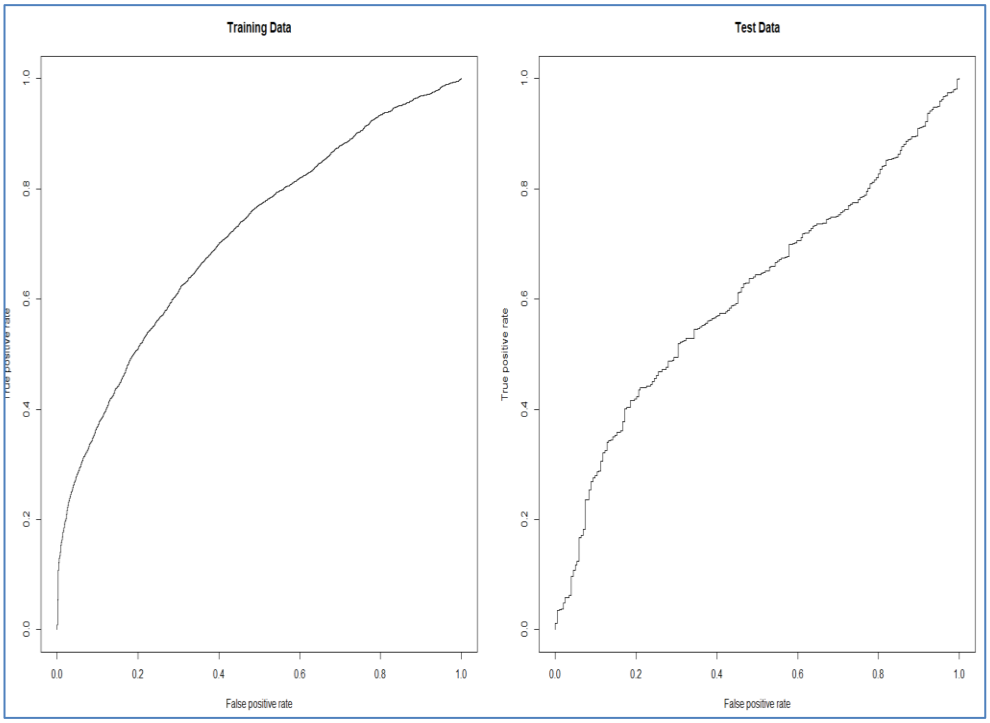


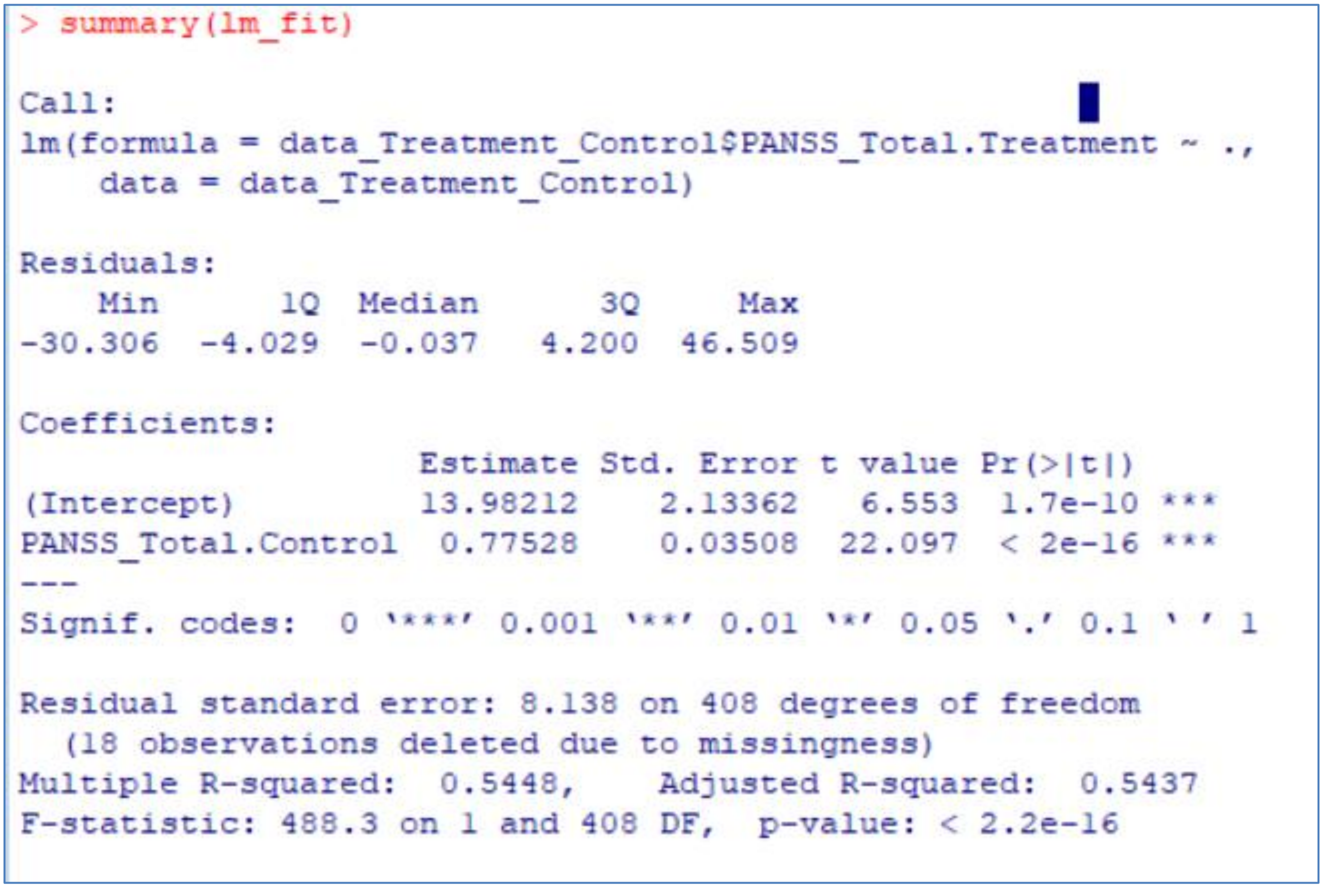
FIG5:

Another e.g of SVM model ROC curve where the lead and lag variables were not considered in the model thus leading to mostly incorrect probability predicitons.



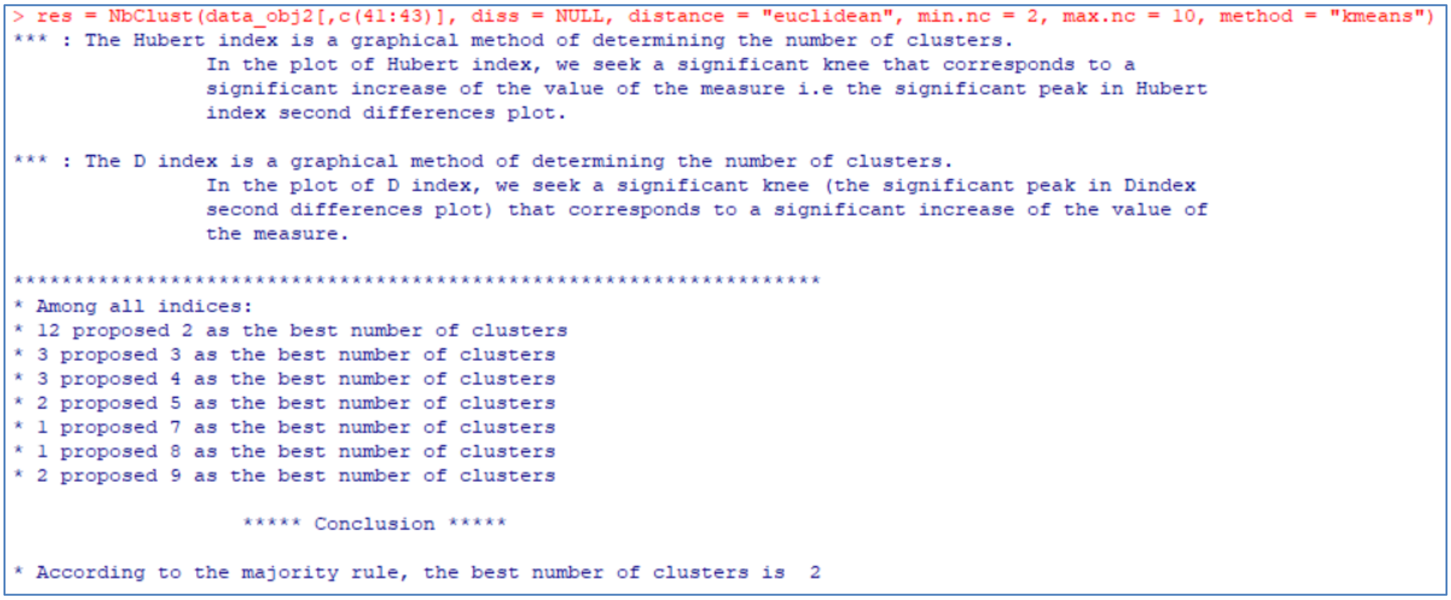
APPENDIX: TABLES

TABLE 1:



```
> summary(lm_fit)

Call:
lm(formula = data_Treatment_Control$PANSS_Total.Treatment ~ .,
    data = data_Treatment_Control)

Residuals:
    Min      1Q  Median      3Q     Max
-30.306  -4.029  -0.037   4.200  46.509

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)         13.98212    2.13362   6.553  1.7e-10 ***
PANSS_Total.Control  0.77528    0.03508  22.097  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.138 on 408 degrees of freedom
  (18 observations deleted due to missingness)
Multiple R-squared:  0.5448,    Adjusted R-squared:  0.5437
F-statistic: 488.3 on 1 and 408 DF,  p-value: < 2.2e-16
```

TABLE2



```
> res = NbClust(data_obj2[,c(41:43)], diss = NULL, distance = "euclidean", min.nc = 2, max.nc = 10, method = "kmeans")
*** : The Hubert index is a graphical method of determining the number of clusters.
                In the plot of Hubert index, we seek a significant knee that corresponds to a
                significant increase of the value of the measure i.e the significant peak in Hubert
                index second differences plot.

*** : The D index is a graphical method of determining the number of clusters.
                In the plot of D index, we seek a significant knee (the significant peak in Dindex
                second differences plot) that corresponds to a significant increase of the value of
                the measure.

*******************************************************************
* Among all indices:
* 12 proposed 2 as the best number of clusters
* 3 proposed 3 as the best number of clusters
* 3 proposed 4 as the best number of clusters
* 2 proposed 5 as the best number of clusters
* 1 proposed 7 as the best number of clusters
* 1 proposed 8 as the best number of clusters
* 2 proposed 9 as the best number of clusters

                   ***** Conclusion *****

* According to the majority rule, the best number of clusters is  2
```

TABLE 3

Training and Test MSE for various linear model iterations.

Training MSE	Test MSE
0.0007197515	0.0008652924
0.0007272261	0.0009265986
8.82185e-26	1.33702e-25
1.874582e-26	2.590885e-26

TABLE 4

Training and Test MSE for various iterations in boosting approach

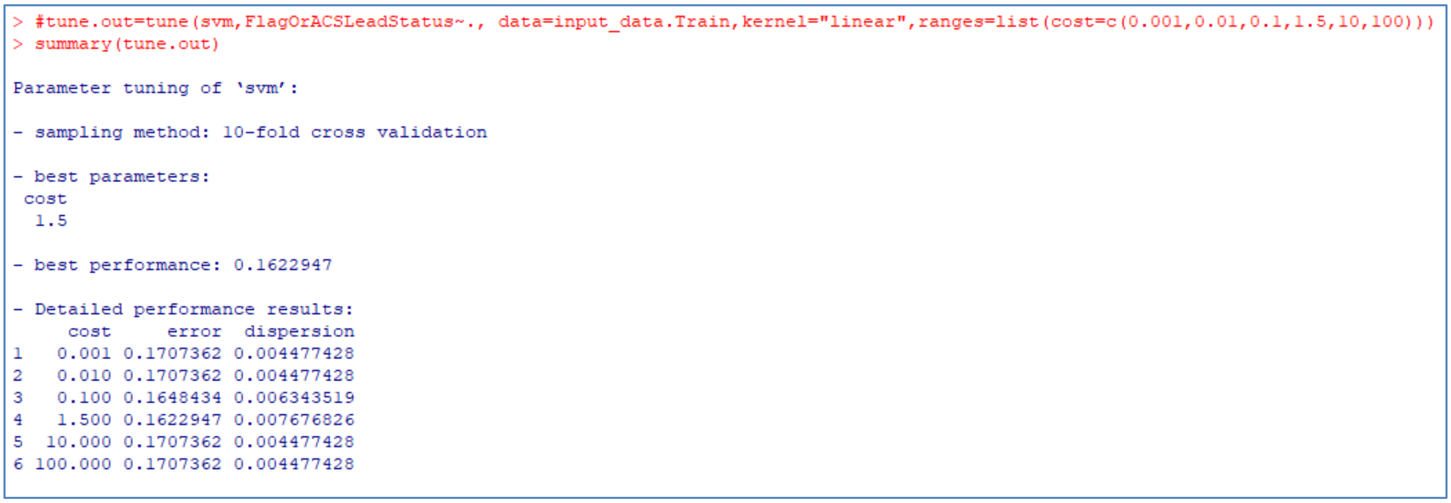
Iteration number	**TRAIN MSE**	**TEST MSE**	**No. of Trees**	**Interaction Depth**	**Shrinkage**
1.	**0.5129537**	**0.7753633**	**500**	**3**	**0.001**
2.	**0.2811479**	**0.5319558**	**1000**	**3**	**0.001**
3.	**0.08392819**	**0.3901659**	**2000**	**3**	**0.2**
4.	**0.06619631**	**0.5275767**	**2000**	**3**	**0.5**
5.	**0.161967**	**0.3700751**	**2000**	**3**	**0.08**
6.	**0.0268199**	**0.1835426**	**4000**	**2**	**0.2511886**

TABLE5

Summary of Train and Test MSE for different approaches used.

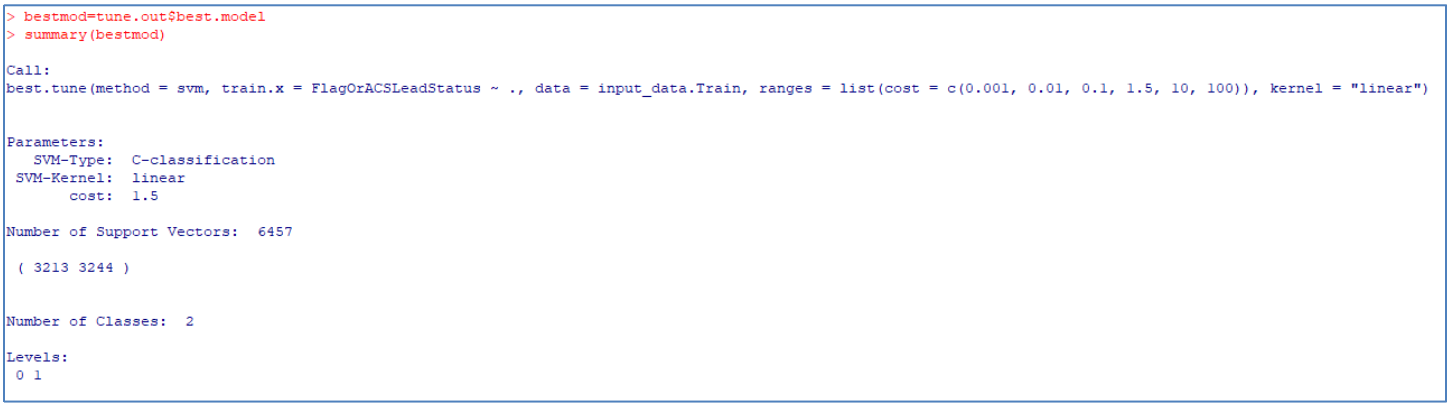
	Train MSE	Test MSE
Linear Model	8.551817e-25	8.704427e-25
Bagging	0.1182304	1.34499
Random Forest	0.2747294	2.088028
Boosting	0.0268199	0.1835426

TABLE6



```
> #tune.out=tune(svm,FlagOrACSLeadStatus~., data=input_data.Train,kernel="linear",ranges=list(cost=c(0.001,0.01,0.1,1.5,10,100)))
> summary(tune.out)

Parameter tuning of 'svm':

- sampling method: 10-fold cross validation

- best parameters:
 cost
  1.5

- best performance: 0.1622947

- Detailed performance results:
     cost     error  dispersion
1   0.001 0.1707362 0.004477428
2   0.010 0.1707362 0.004477428
3   0.100 0.1648434 0.006343519
4   1.500 0.1622947 0.007676826
5  10.000 0.1707362 0.004477428
6 100.000 0.1707362 0.004477428
```

TABLE7



```
> bestmod=tune.out$best.model
> summary(bestmod)

Call:
best.tune(method = svm, train.x = FlagOrACSLeadStatus ~ ., data = input_data.Train, ranges = list(cost = c(0.001, 0.01, 0.1, 1.5, 10, 100)), kernel = "linear")


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  linear
       cost:  1.5

Number of Support Vectors:  6457

 ( 3213 3244 )


Number of Classes:  2

Levels:
 0 1
```

REFERENCES:

- **[1]:** The Positive and Negative Syndrome Scale (PANSS) for Schizophrenia (Kay et al 1987) page-265
- **[2]:** https://www.analyticsvidhya.com/blog/2016/03/tutorial-powerful-packages-imputing-missing-values/