# Stats 202: Data Mining and Analysis

Faraz Ahmed Khan

Stanford University

Team Name: Schizophrenics

Submitted to: Dr Linh Tran

Stats 202: Data Mining and Analysis

**Data Treatment:**

We were provided a data set of study, which had five different trails, each trials was labelled as A – E. Each row in the dataset, represented information regarding an assessment, including the patientID, trail label, siteId, country of patient, and then symptoms information regarding that patient. The data of trail A-D was provided completely to us, while we had to predict on trial E. The first and foremost step that we took were to combine data from trial A-D into one dataframe, so that it is easier for us to work on the data as one whole. We did not divide the data into training, test and validation sets at this time, because we wanted to test and trains our models with data coming from all four trial, rather than test data coming from a certain trial and test data from another trail.

After the merging process, we had a total of 20947 observations. We created a new file, merging.csv to store all those observations and then used that file for the any analysis later on.

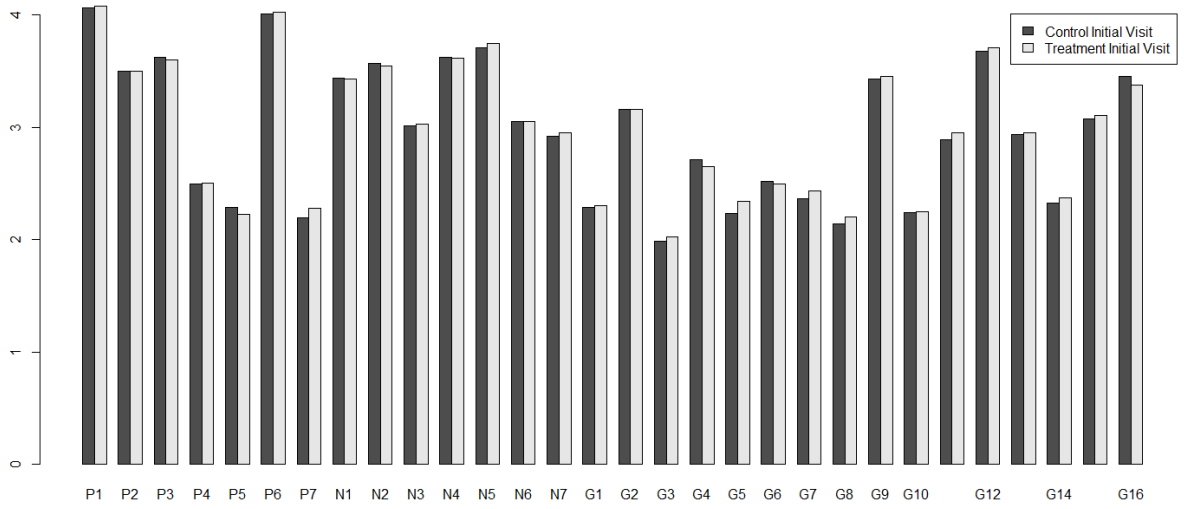
**Objective 1 (Treatment effect):**

We had to identify if the medication had any impact on the patients. Patients were divided into two groups, i.e. control and treatment. The basic objective was to identify if there was any change in behavior with respect to time with each of these groups.

We modelled this into three questions for ourselves.

- If the treatment has any impact in general (on the PANS_Total, i.e. the sum of all symptoms)?
- If the treatment had any impact on any of the specific or specific set of symptoms?
- How do we quantify change to answer the above questions?

We started off with basic analysis and statistics of the data. Of the 2438 patients, 1227 were control and 1211 were treatment, and thus we had almost a 50-50 divide between each group. Furthermore, we tried to analyze the symptoms on the first day of the treatment, as to analyze if there was any bias between each of the groups.




*1*

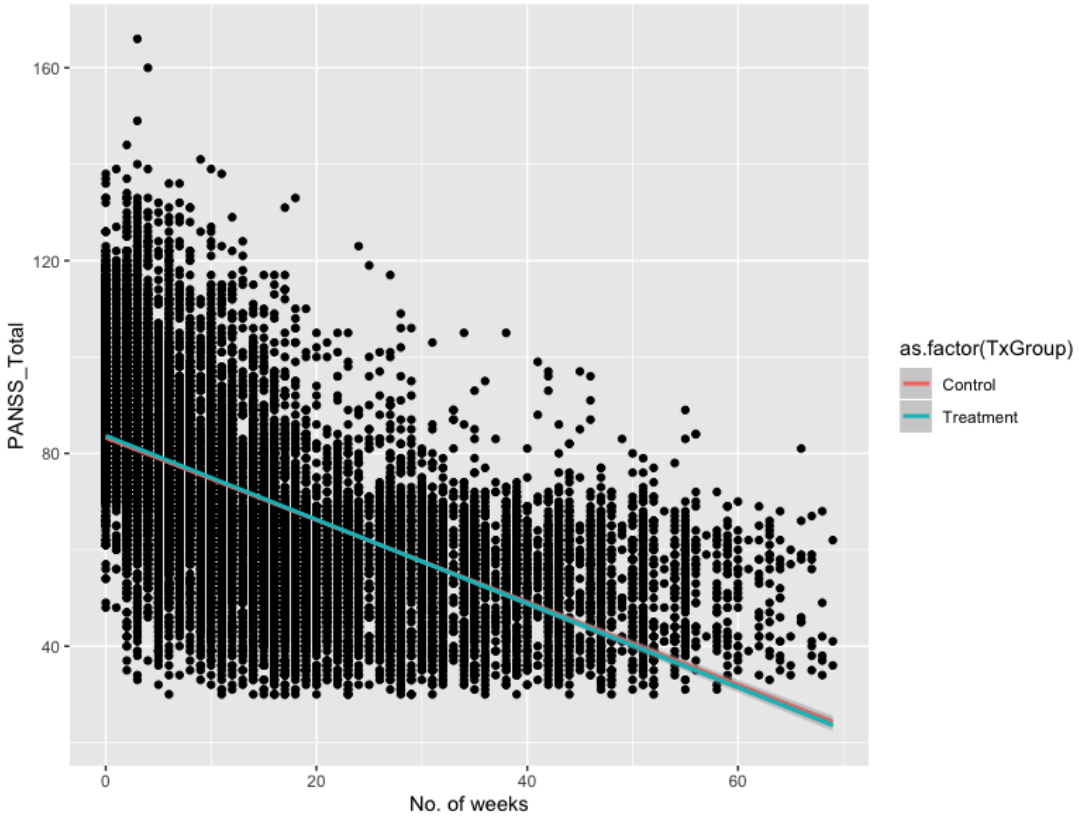
Figure 1 provides an overall picture of the data on the first day of treatment. We notice that both the groups have almost the same means on each of the 30 symptoms. This suggests that both the groups are well randomized and that there is no bias in any of the groups.

We then started with answering the 3rd question, as to how can we quantify the change. We added columns for change for each of the symptoms and PANSS_Total i.e. we had total 31 columns initially for symptoms and PANSS_Total, we produced 31 more columns, each of which recorded a change in each symptoms. So, for observations, whose visit day was 0, the change was 0, but then for observations whose visit day was not 0, the change was measured as the difference from the last visit of that patient, divided by the number of days since that visit, to get a change per day in each. We understood this was not the best way to accommodate for change,

but also realized, a significant change in the means of these changes would help us indentifying the treatment affect.

Now we had the data, to answer the first question. The null hypothesis, that we setup was "there is no affect of medicine on each group"

We measured means of change per day in PANSS_Total in each of the groups, which came out as -0.189 (for control) and -0.184 (for treatment) and the p-value (0.7233f) or the t-test performed on the changes suggested that the change between each of the groups was not significant, which confirms the null hypothesis. To be completely, sure, we tried to regress PANSS_Total onto number of weeks (derived from visit day) and tried to see if there was any visual different between the two groups. As Figure 2 illustrates, the change between the two is negligible and thus, the medication had no impact on the PANSS_Total.




*2*

For question 2, we tried to go through a similar procedure. We tried to calculate the means of the change, and saw that the means were almost equal, i.e. each of the symptoms did not have significant change based on their group. Figure 3 illustrates our analysis.


*3*

Figure 3 illustrates that the mean of change per day is almost equal. We similarly tired to regress each symptoms versus the no.of weeks, but similarly found no different between each of the group[1]. Thus from the analysis conducted, we can conclude there is **no effect of treatment on patients**.

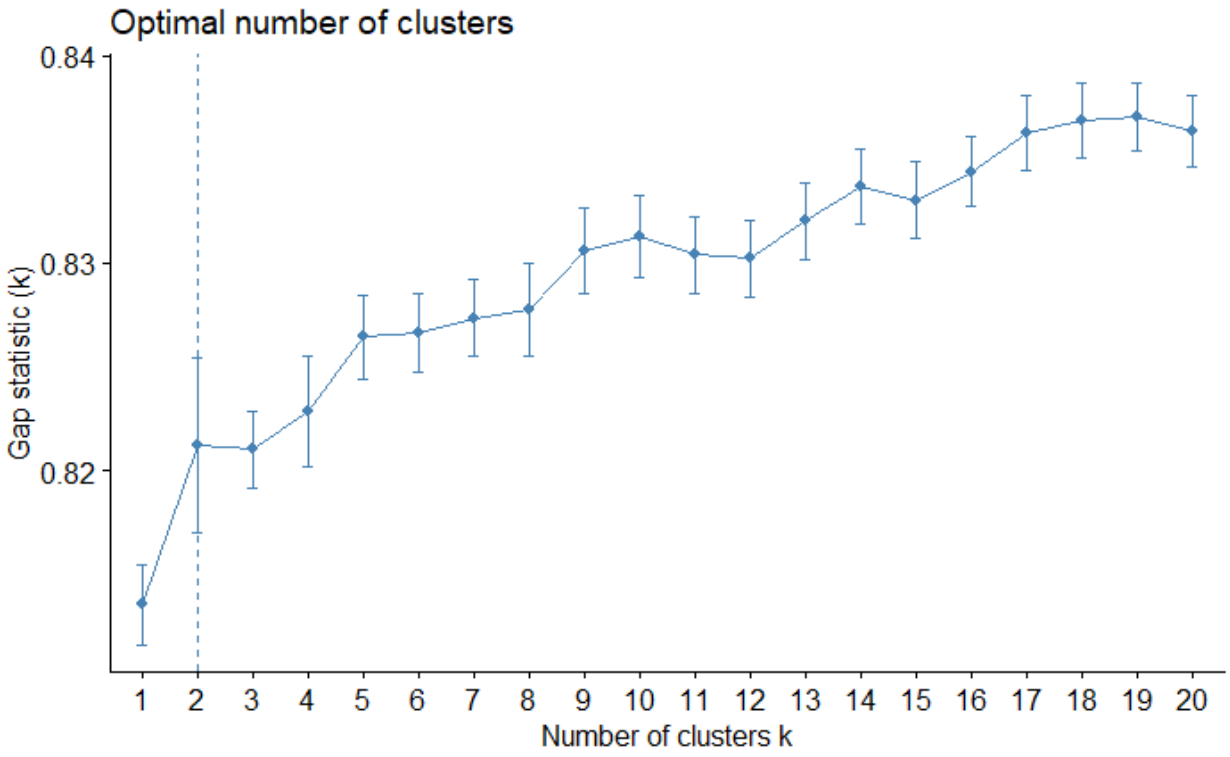
**Objective 2 (Patient segmentation):**

For this objective, we had to divide the patients into k different groups, and justify why we chose that certain k. We perceived this problem as clustering the patients into k groups, where

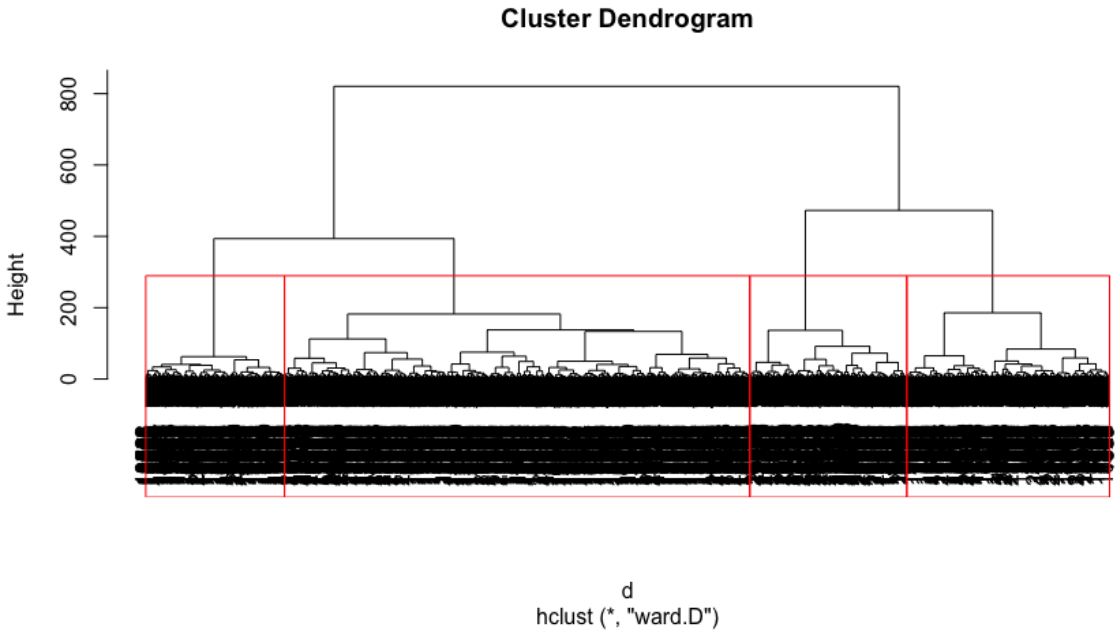
[1] Not attaching figures to avoid cluttering in the report. Figures can be found in the appendix

each of the k clusters could be explain through similar symptoms having similar value(s). For this purpose, we excluded the PANSS_Total, country, site, raterID from our calcutations.

We decided to use Euclidean  distance for our analysis as it gives equal weight to each variable and for us each symptom was important, and that it is scale invariant, suitable for our data as it is an interval scale. We decided to use heirchial clusterting as were unable to pre-determine the number of clusters, and further more, our analysis using kmeans didn't help us identify the number of clusters.




The gap statistic, using kmeans suggested using 2 clusters, but those 2 clusters were extremely as most of the data lied in only 1 cluster, and the outliers lied in the other cluster .Thus we decided to go with hierarchical cluster, using Euclidean distance.




The dndogram was well separated, and cutting the dendogram such that 4 clusters are obtained, gave quite well separated clusters, hence we decided to cut at height 4.

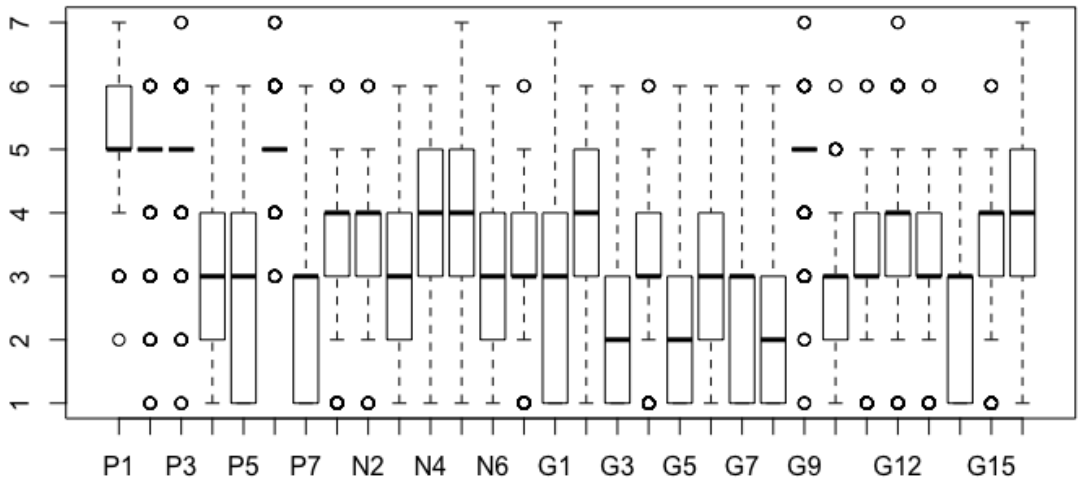
When looking at the clusters obtained after cutting the dendogram, and looking at different subtypes of schizophrenia acknowledged universally, we identified that our clusters aligned well with the acknowledged subtypes[23].

The first cluster, indicted higher values of the positive symptoms and thus is accepted as a positive type schizophrenia, G9 symptom is also considered, as correlated to the positive symptoms, hence it also has a higher value.

---

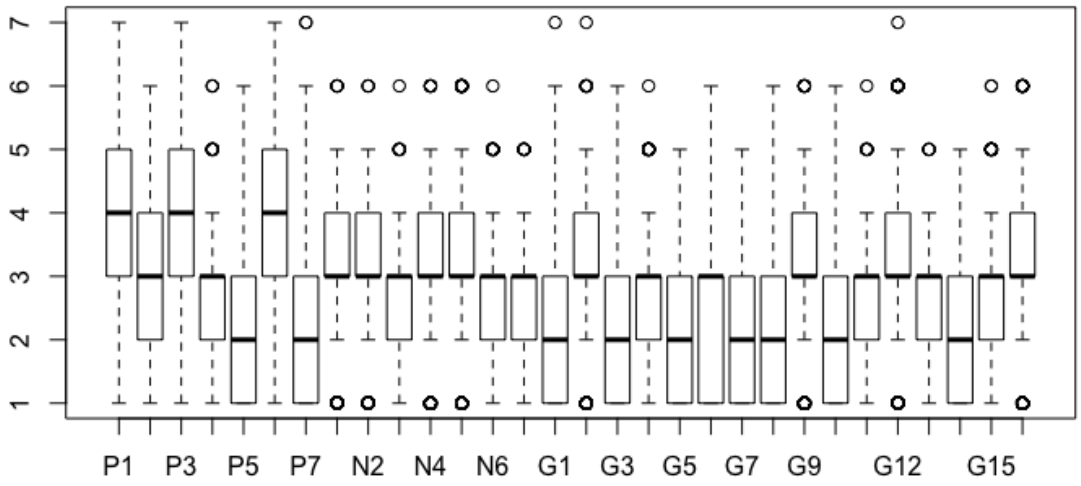
[2] . Lefort-Besnard J, Varoquaux G, Derntl B, Gruber O, Aleman A, Jardri R, et al. Patterns of schizophrenia symptoms: hidden structure in the PANSS questionnaire. Transl Psychiatry. 2018 Oct 30;8(1):1–10.

[3] Thokagevistk K, Millier A, Lenert L, Sadikhov S, Moreno S, Toumi M. Validation of disease states in schizophrenia: comparison of cluster analysis between US and European populations. J Mark Access Health Policy [Internet]. 2016 Jun 20;4. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4916257/



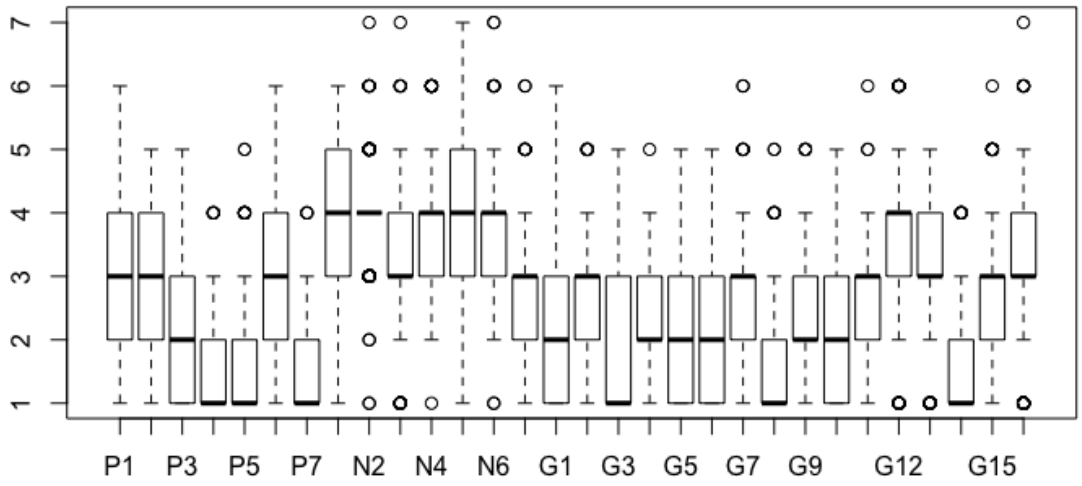

*4 Cluster 1*

The second cluster, indicated towards the early cognitive impairment, this subtype is universally acknowledged as the most common subtypes, and was also our biggest cluster. The symptoms (N5, N7, G13, G11, G12, G15, G16, G4) associated with cognitive behavior had higher values.



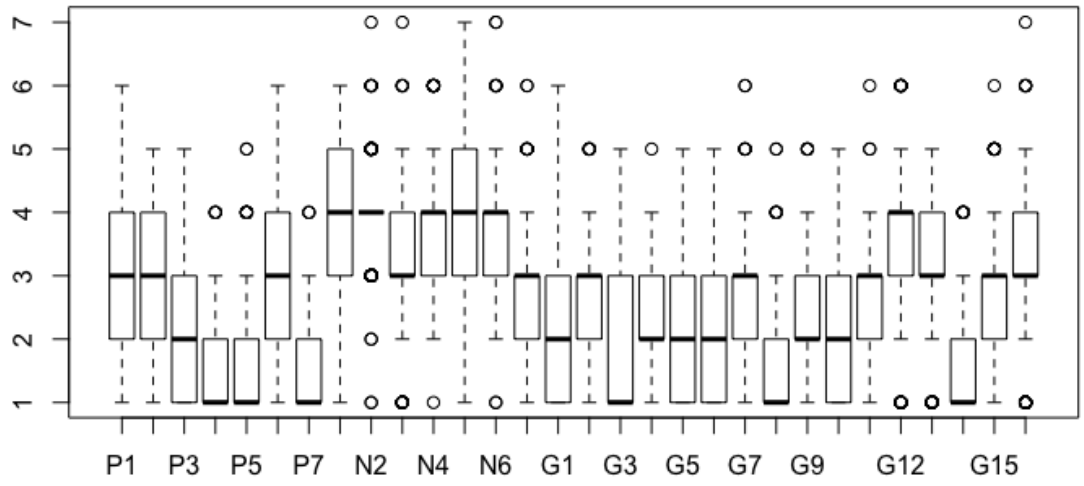

*5 Cluster 2*

Similarly, cluster 3 indicated towards mixed subtype schizophrenia as it had moderate to high values of different symptoms, especially hostility (P7) which is why it's also called subtype schizophrenia hostility.




*6 Cluster 3*

The final subtype called the Negative Schizophrenia Subtype, has higher negative symptom values.



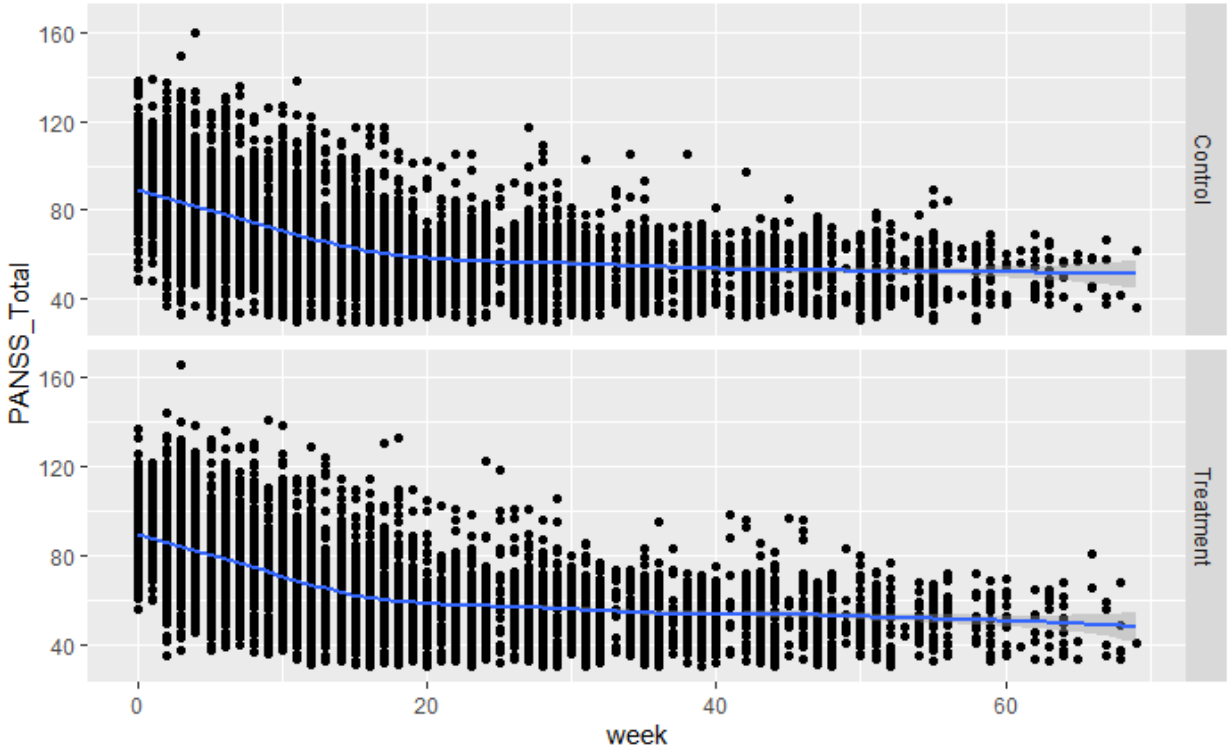

*7 Cluster 4*

**Objective 3 (Forecasting):**

For this talk we had to predict the PANSS_Total for a patients visit on week 18$^{th}$, this task was rather complicated, as the data we received was not segmented into weeks, rather it was given with visitDays, We were unsure of how to cope with this problem, but then we decided to derive the number of weeks into days. We simply added a column to our dataframe, we indicated the week of the assessment, i.e if visitDay was 10, the week was 2. Day 0, was considered as week 0.

Now, we had a data which had week as its last column indicated the week of assessment. We have already derived that treatment and control have no significant impact on PANSS_Total, and the figure 4 below reaffirms that derivation. Thus we did not included treatment group in our calculations going forward. Each assessment was taken by a different rater, and that each study had its own rater and siteId which is why we decided to not included that in the calculations going forward. So, with this, we were only left with the symptoms and the PANSS_Total data, since the panss_total in itself was a sum of the symptoms, we decided to not include individual

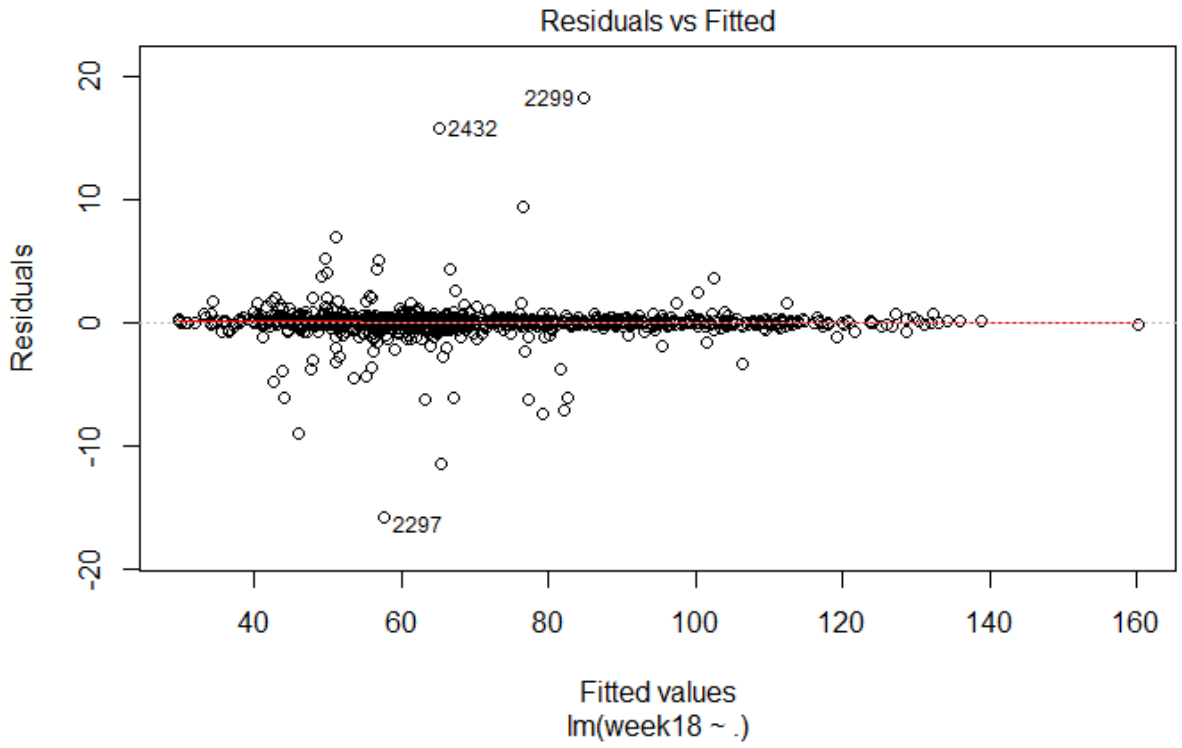
symptoms and only include the PANSS Total for each week. Thus our model was only left with PANSS_Total and the week that assessment was taken in.




*8*

The dataset was divided into 3 sets, i.e. train (60%), test(20%) and validate(20%). We started our modelling with applying a linear regression model on the train data, but for that the data had to be converted into a wide format, as opposed to the long format it was currently in. The problem with the wide format was the missing values, we did not have each patients data for each week, and thus in the long format our matrix had a lot of missing values. These missing values were *missing at random* and the only option we had was to impute it, otherwise it was not

possible to regress on it. We produced a regression model, and model looked very accurate.

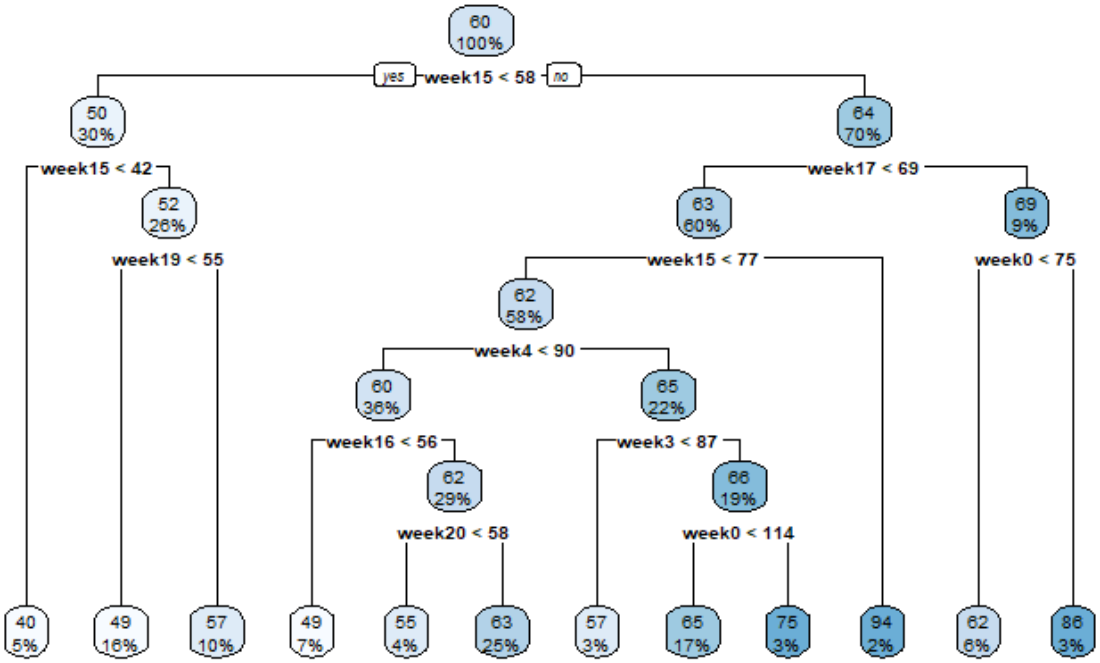



*9*

Figure 9, shows the graph of residuals vs fitted values for that regression model. The same model had 1.37 Test MSE, which was very low, but it was not as meaningful because both the test and train data had to be interpolated to work. This is why this modal performed poorly on the kaggle leaderboard. We also, tried knn regression, but ended up facing a similar issue.

We thus decided, that a regression model was not the best way to go about this, as the data would then have to be interpolated and that would not produce accurate results. The only alternative that we found useful was decision trees, as it was able to incorporate the missing values.

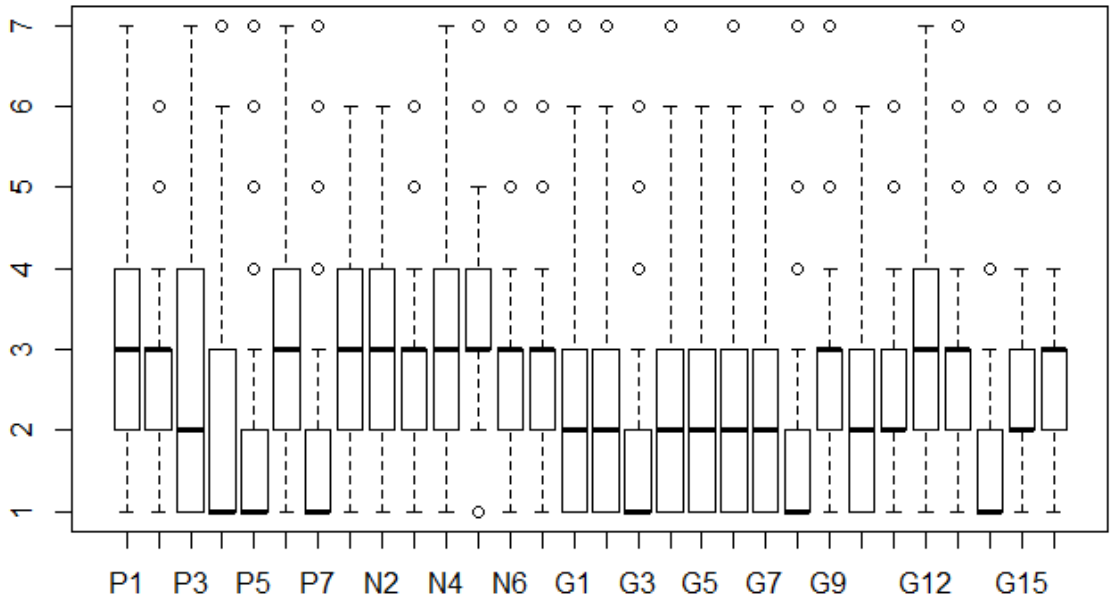
Thus we decided to produce a decision tree, without interpolating the data. We also combined the train and validation data set. The tree produced by the decision tree looked like this,



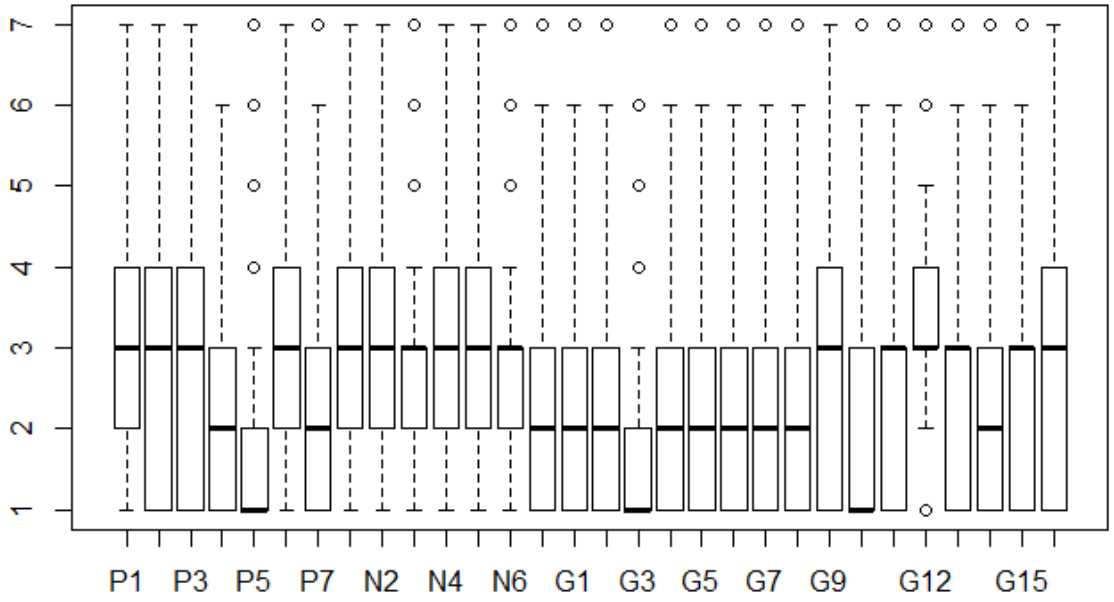

Summary of the tree, indiciated week 17 and week 15 PANSS_Total to be most important factors for determining the week 18$^{th}$ PANSS_Total, and this is expected as these are closer to week 18. This model produced a test MSE of 200, and this was our final model on Kaggle leaderboard.

**Objective 4 (Binary classification):**

The final objective required finding the probability of the assessment being passed or not (flagged or assigned to CS). First and foremost, we tried to observe the general pattern with flagged and passed observations, with respect to the thirty symptoms.



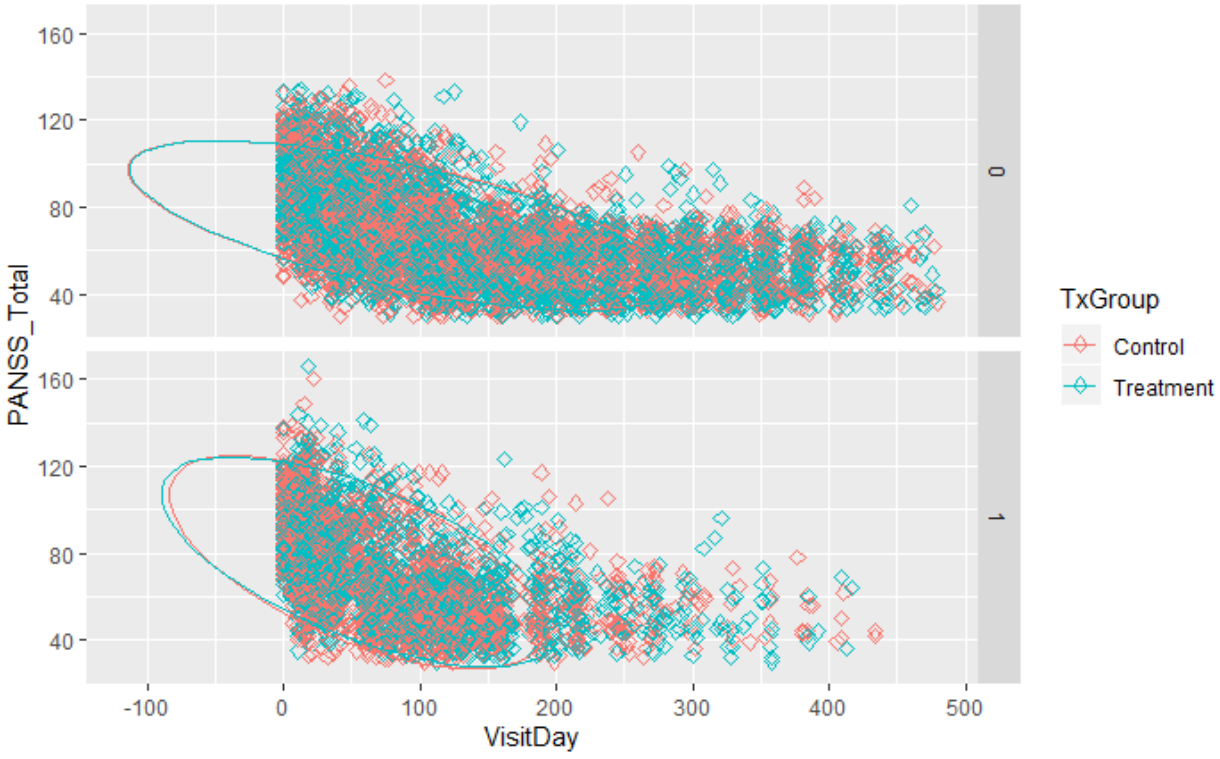

*11:* Boxplot of Symptoms in passed assessments




*10:* Boxplot of Symptoms in flagged or assigned to CS assessments

A pattern that is quite clearly observable (and expected) is that the assessments which are quite close to the mean value are passed, but the observations which are not passed have outlier values in each of the symptoms. This was a nice perspective to start the analysis, and gain further insights into the data.

We then transformed the LeadStatus column to indicate 0 for passed observations, and 1 for flagged and assigned to CS observations. Now we had to decide whether to include visitDay in the model for classification or not. Of all the observations, 24% had been flagged or assignment to a CS, and of all the baseline (first visit) observations, 28% had been flagged or assigned to a CS, the difference was not significant hence, we decided to not include the visit day in our model for classification.

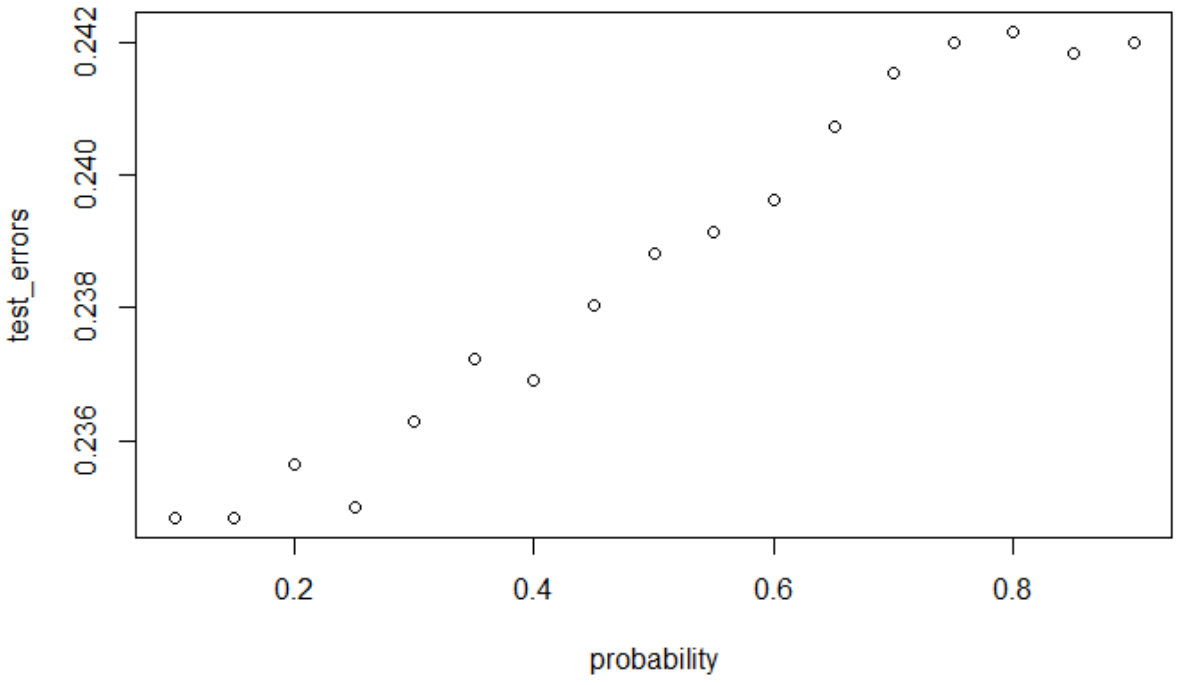



*12*

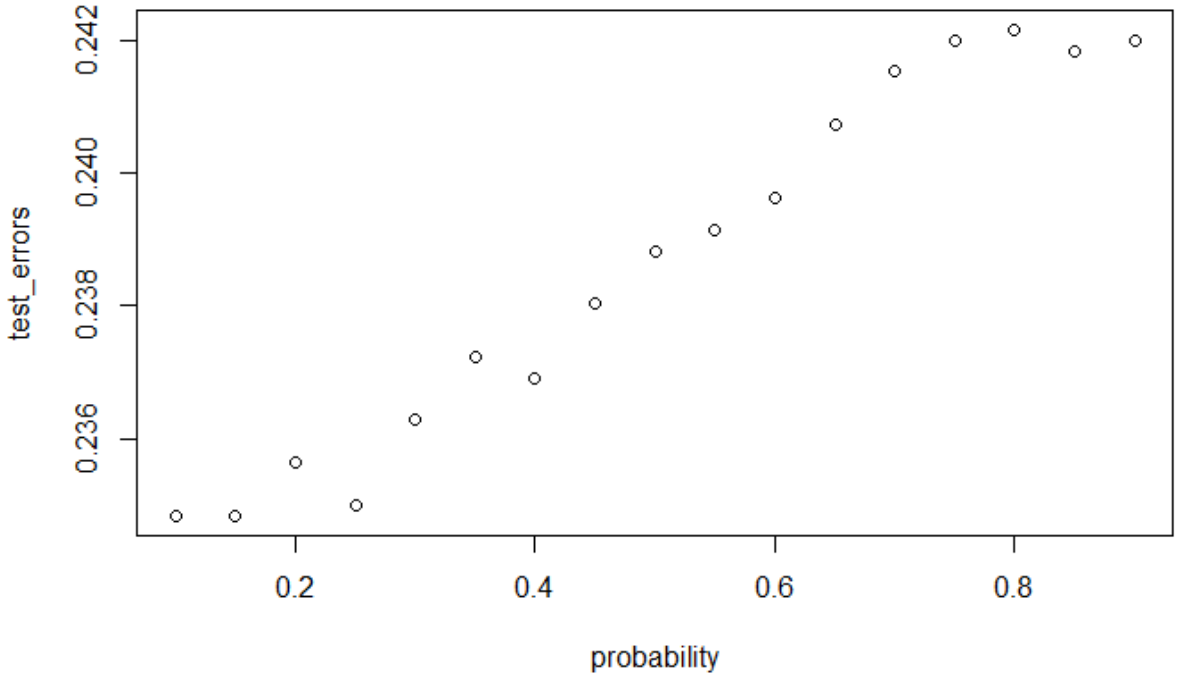
Figure 12 displays PANSS_Total vs VisitDay, and this again confirms what we had already discovered that, the outliers (specially those having higher values) had a higher chance of being flagged or assigned to CS. We could have included PANSS_Total in the final model, but since we were including its parts (the symptoms) we decided to not include the symptoms. Also including the PANSS_Total did not have a significant impact on the misclassification rate, hence we decided to not include it. We have already established how the TxGroup has no impact on the treatment and panss score, the TxGroup also did have no impact on the observation being passed or not (figure 12).

With this information, we understood that the observation being passed or not, relied more on the symptoms value rather than anything else, so we restricted our domain of analysis to each of the 30 symptoms.

We divided the dataframe into two sets, train (70%) and test(30%), and started our analysis with the basic LDA and logistic regression, each of produced misclassification rate of around 24% on the test data.

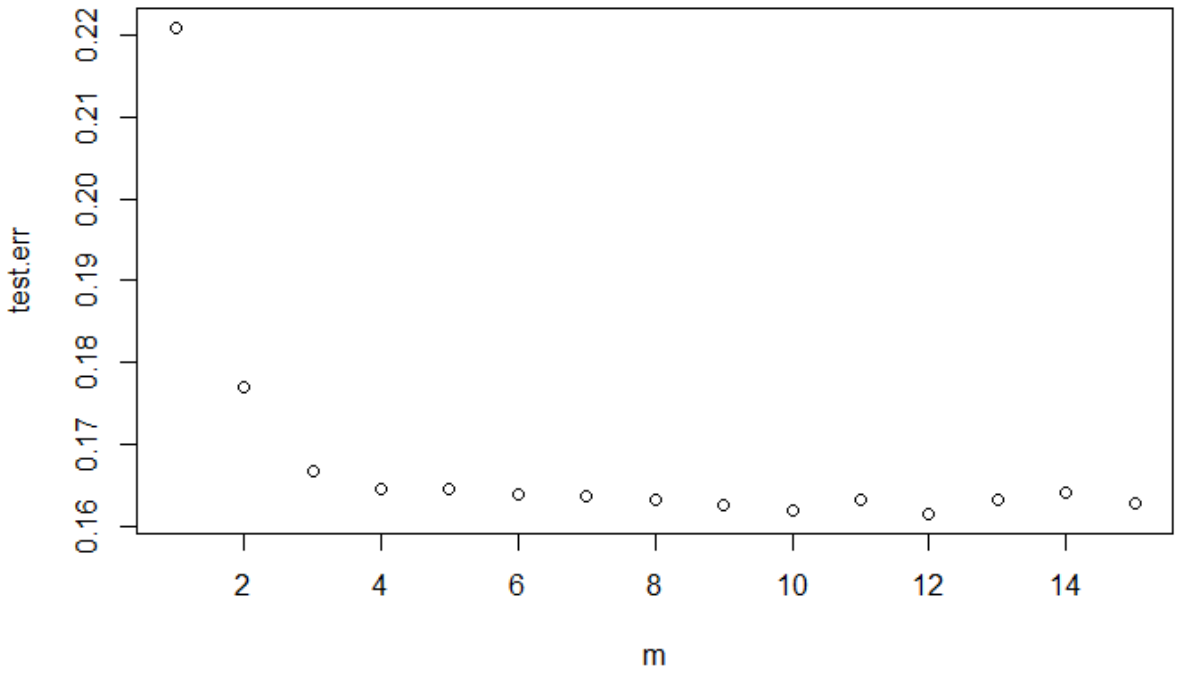



*13 Probabiliy vs Test Errors for logistic regresion*




*14 Probabiliy vs Test Errors for LDA*

Since the problem involved a lot of variables, and that the interaction between the variables was also important for our model, we decided to implement random forests and bagging (special case of random forest) as our model and achieved better results, figure 11, shows the results of random forest on our dataset for 1000 trees. The hyperparameter was choosen after various comparing similar graphs, such as 15, to see which fared better on the test data.




*15 Test Errors vs m for Random Forest for 1000 trees*

Bagging for the same data, produced an error rate of 16.5% which was slightly greater than the error rate produced by the best random forest model.

Thus we went on to upload the best random forest model to Kaggle for the leaderboard.

Since the treebased method, produced a much better output, we also wanted to try out, boosting to see if it fares better in this scenario. Boosting did not work with LeadStatus being a factor, so we had to convert it to a character column. Boosting when implemented, also provided similar errors rates as those of random forests with different hyperparaters. Boosting with hyper parameters (shrinkage = 0.10 and interaction depth = 5) produced the best outputs on test data (Test MCE = 17.6%).