EP4130: Final Project

Ugranam Hema Chandar
EE20BTECH11062

Korapala Hrushikesh
EE20BTECH11022

Indian Institute of Technology, Hyderabad — May 1, 2024

Abstract

This project encompasses four primary objectives. Firstly, it aims to compare the effectiveness of existing and new medications. Secondly, it involves clustering patients based on their initial measurements. Thirdly, it seeks to predict PANSS scores at the 18th week. Finally, it addresses the task of classifying patients into groups indicating whether they have passed or failed a test, or if they need further evaluation by a clinical specialist, based on available data. This report discusses the outcomes of applying data science and statistical methodologies to a medical dataset tracking patients diagnosed with schizophrenia over time. The dataset comprises thirty scores that gauge the severity of schizophrenia symptoms on a scale of 1-7 using the Positive and Negative Syndrome Scale (PANSS), alongside information such as country and visitation days. Each assessment records factors like the patient's ID, evaluator's ID, assessment location, assessment day, and PANSS scores.

Introduction

As part of this project, we are analyzing data gathered from five randomized controlled trials involving patients diagnosed with schizophrenia. Initially, patients undergo a screening test to determine their eligibility for further analysis. Subsequently, an anonymized drug is assessed for its effectiveness in treating schizophrenia across all five trials. Patients participating in these trials are monitored for varying durations, depending on the study's criteria, during which their symptoms related to schizophrenia are observed. Upon enrollment, patients undergo baseline measurements, recorded on their initial visit (i.e., visit day 0). At this stage, they are randomly assigned to one of two treatment groups: the treatment group receives the anonymized medication, while the control group is administered the accepted standard medication for schizophrenia treatment. Throughout the study period, patients return for follow-up visits, during which the same measurements are repeated.

0.1 Data Briefing

The below explains the fields of the each csv file (an image extracted from problem statement):

1. *Study* - A character indicating which of the five studies the data represents.
2. *Country* - The country where the assessment was conducted.
3. *PatientID* - An identification number given to each unique patient.
4. *SiteID* - An identification number given to each unique assessment site.
5. *RaterID* - An identification number given to each unique rater.
6. *AssessmentiD* - An idenfication number given to each unique assessment conducted.
7. *TxGroup* - A string corresponding to the patient's (randomly) assigned treatment group.
8. *VisitDay* - An integer corresponding to the number of days that have passed since the baseline assessment.
9. *P1-P7* - The scores corresponding to each of the 7 positive symptoms of the assessment.
10. *N1-N7* - The scores corresponding to each of the 7 negative symptoms of the assessment.
11. *G1-G16* - The scores corresponding to each of the 16 general psychopathology symptoms of the assessment.
12. *PANSS_Total* - The sum of of the ratings across the 30 PANSS items.
13. *LeadStatus* - A string indicating whether the assessment's audit passed, was flagged, or was assigned to a CS (i.e. clinical specialist).

1 Treatment Effect

1.1 Introduction

As mentioned previously, our study involves two distinct groups: the Control group and the Treatment group. These groups were established to assess the effectiveness of a newly proposed medication for treating schizophrenia. Patients assigned to the Control group received the standard or existing medication, while those assigned to the Treatment group were administered the anonymized or newly proposed treatment. Below, we outline the methodology employed to address this inquiry:

1.2 Pre-Analysis

We operate under the assumption that there is no inherent bias in the initial schizophrenia measurements of patients. However, in practice, there may be a tendency to assign patients with more severe symptoms to the control group (receiving standard medication), while those with milder symptoms are allocated to the treatment group (receiving the experimental treatment), and vice versa. This phenomenon, known as data-snooping bias, can skew our analysis since we are retrospectively conducting statistical assessments after the experiment has been conducted. To mitigate this bias, it is imperative to examine the distributions of baseline measurements and ensure they closely align. If the initial measurements exhibit substantial disparities, it becomes challenging to proceed with meaningful comparisons.

As said in the previous section, to plot the distributions of treatment and control groups, I have considered histograms of the given data using **freedman** rule to calculate optimal bin widths for histograms. Also, I have also used **Kernel Density Estimation (KDE)** to plot Kernel density plots using Gaussian Kernel.

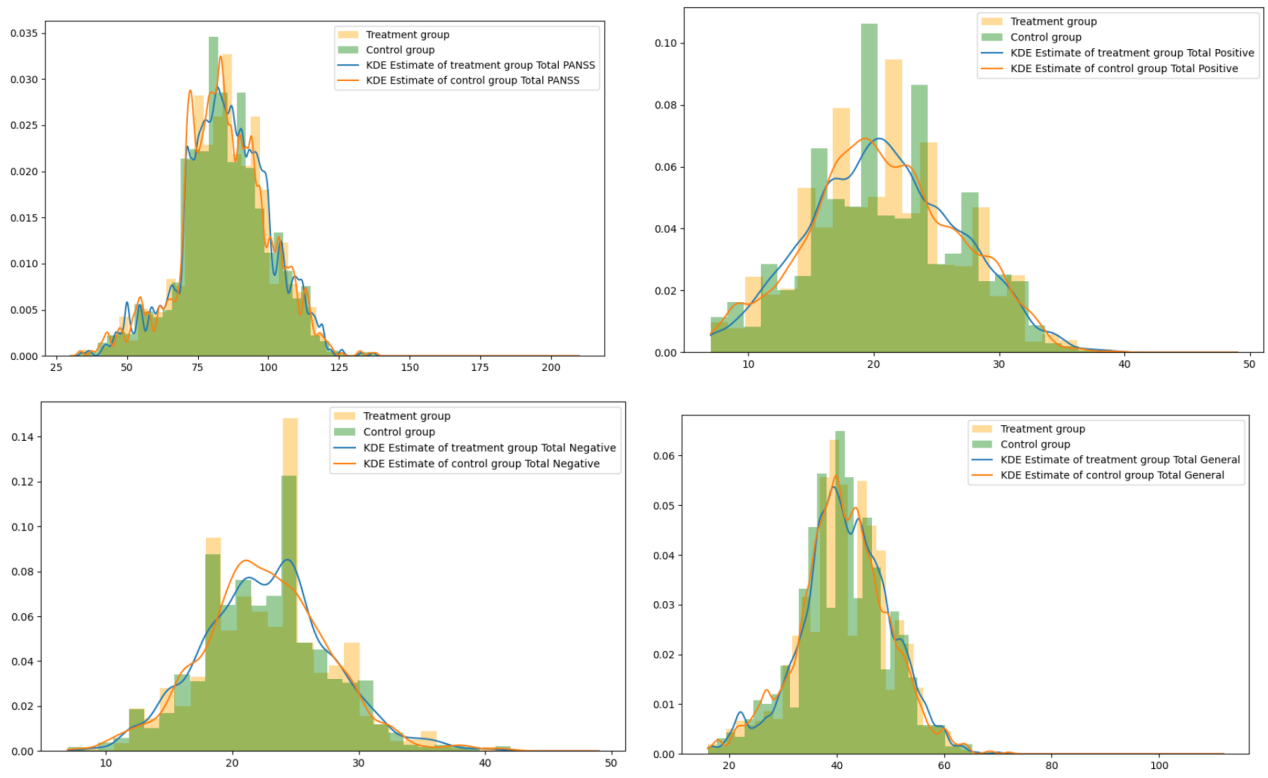

Fig - 1: Initial Distributions histograms as well as KDE plots for different features (a) Total PANSS Score (b) Total positive scale score (c) Total negative scale score (d) Total general scale score

The above are the plots for distributions of total PANSS_score, P_total, N_total, G_total across all studies combined for both treatment and control groups. The histograms and KDE plots of baseline measurement distributions for both groups indicate a significant overlap, suggesting that we begin with nearly unbiased data for both the treatment and control groups. Given this lack of bias in the initial data, we can proceed to compare the data across visit days for both groups.

1.3 Analysis

Since we can objectively compare the two groups, we can utilize data science techniques to assess the efficacy of the newly proposed treatment (for the treatment group) compared to the standard medication (for the control group). Our primary measure of treatment effectiveness focuses on the reduction in PANSS_score, P_total,

N_total, G_total. A decrease in these scores indicates a progression towards normality in symptoms, suggesting that the treatment is effectively addressing the condition.

1.3.1 Using Correlation, LOWESS and Scatter plots

I've analyzed the correlation between various features by plotting a heatmap, showcasing the relationship between PANSS_score, P_total, N_total, G_total, and TxGroup. It's important to note that I disregarded identifiers such as IDs, as they are unique numbers generated but not relevant to the analysis. The heatmap indicates that the correlation between PANSS_score, P_total, N_total, G_total, and TxGroup is almost negligible across all studies.

To determine the statistical significance of this observation, I computed the p-value statistic for the correlation between these measures and TxGroup. Specifically, the Pearson correlation p-value for PANSS_score vs. TxGroup is 0.987 (rounded to three digits). Our null hypothesis posits that there is no correlation between PANSS_Total and TxGroup. Considering the alternative hypothesis that there is a correlation between PANSS_Total and TxGroup, the corresponding p-value (1-0.987) is less than 0.05. Therefore, we reject the alternative hypothesis, concluding that there is no significant correlation between PANSS_Total and TxGroup.

Note : In the above, we have used the fact that, p-value of null hypothesis + p-value of alternate hypothesis = 1 **if and only if** alternate hypothesis is conjugate hypothesis of null hypothesis. In the above case as we are considering alternate hypothesis as any non-zero correlation case which is exactly conjugate for no correlation (null hypothesis).

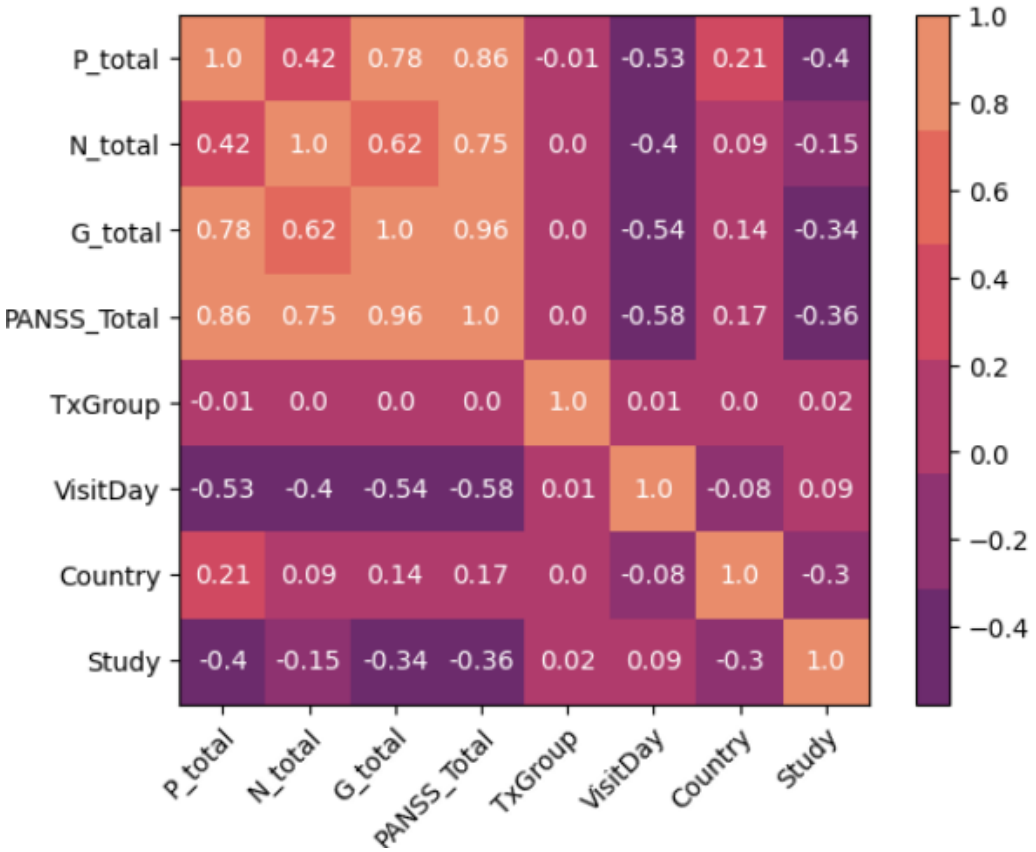

Fig -2 : The correlation map of one feature over other.

Based on the analysis conducted, we deduce that the measurements across all visit days exhibit independence from the two groups, indicating that there is **no discernible improvement with the new treatment** compared to the standard medication. However, when examining individual positive, negative, and general scale measures, the p-values for the null hypothesis were found to be 0.173, 0.482, and 0.584, respectively. These values suggest that we cannot reject the null hypothesis nor the alternative hypothesis, suggesting that the newly proposed treatment may have efficacy on specific subsets of positive, negative, or general symptoms, or vice versa.

Another approach to addressing this issue involves plotting the measures against visit days for both the control and treatment groups, and employing locally weighted scatterplot smoothing (**LOWESS**) plots for analysis.

LOWESS is a non-parametric method which uses locally weighted linear regression (LWR) algorithm to fit smooth curve data depending on local weights. More info about it can be found here [1]. For this project, we are using LOWESS function that is already implemented in "statsmodels" [5] package.

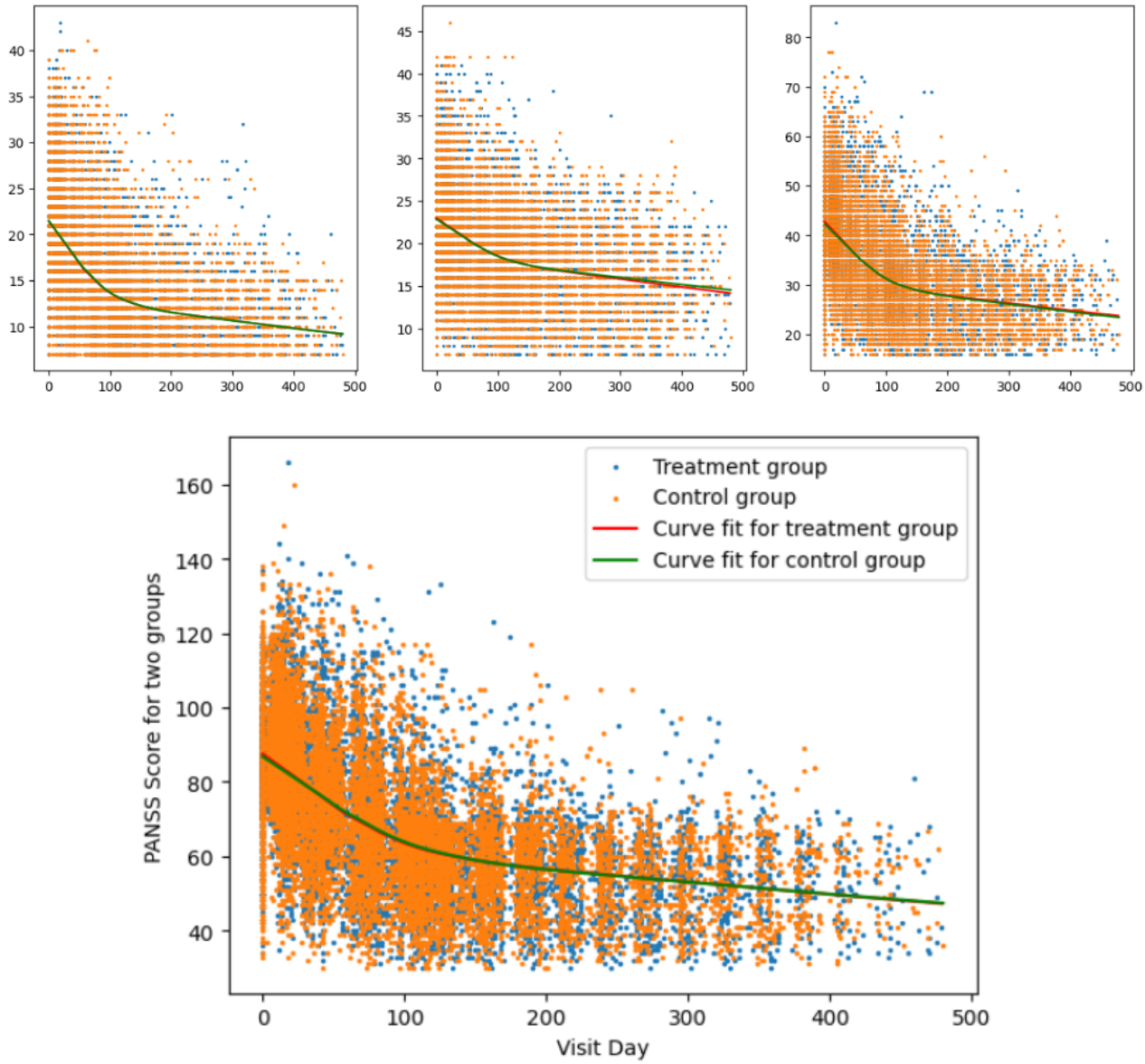

Fig - 3: Scatter and LOWESS plots (a) Total positive, negative and general scales score (from left to right). Red line denotes LOWESS plot for treatment group & green for control group. (b) Total PANSS Score

In order for the treatment to demonstrate efficacy, the measures should exhibit a rapid decrease over the course of the visit days. This implies that curves which remain lower indicate a better response to treatment. From the provided plots, it is evident that the scatter points and LOWESS curves closely overlap each other. Notable deviations in the curves are observed primarily in the Total negative and general scales. Overall, the plots suggest that the two groups exhibit similarities, but there appears to be a slight advantage of the standard medication over the newly proposed medication in addressing general symptoms. Conversely, the newly proposed medication shows a slight advantage over the standard medication in managing negative symptoms.

1.3.2 Curve fitting

In this we will analyse which data fits better to a given model. The model I considered is,

$$\text{Score} = a + b * \text{visitDay} + c * \text{visitDay} * \text{TxGroup} \tag{1}$$

If c $\rightarrow$ zero, we can conclude that there is no improvement in new method. The below are results obtained:

Score	a (p-value)	b (p-value)	c (p-value)
PANSS_Total	$81.65(0)$	$-1.17 \times 10^{-1}(0)$	$-2.13 \times 10^{-4}(0.892)$
P_total	$19.37(0)$	$-3.70 \times 10^{-2}(0)$	$-3 \times 10^{-4}(0.592)$
N_total	$22.01(0)$	$-2.39 \times 10^{-2}(0)$	$-2.24 \times 10^{-4}(0.663)$
G_total	$39.90(0)$	$-5.62 \times 10^{-2}(0)$	$3.12 \times 10^{-4}(0.705)$

In this scenario, the null hypothesis considers the parameter's value to be 0. From the table provided above, it is evident that there is an 89.2% chance that c equals 0 for PANSS_score. Similarly, for P_total, N_total, and G_total, the probabilities are 59.2%, 66.3%, and 70.5%, respectively. This suggests that the PANSS score is nearly independent of the treatment type. However, this finding does not strongly support our claim compared to other methodologies mentioned earlier. Nonetheless, it indicates a higher likelihood of independence between the PANSS score and treatment type.

1.4 Observations and Conclusions

Based on the aforementioned methods, we infer that the anonymized treatment exhibits efficacy comparable to the standard medication, with **no discernible overall improvement**. However, when focusing solely on negative symptoms, the newly proposed treatment demonstrates effectiveness, whereas standard medication appears more beneficial for general symptoms, as indicated by LOWESS plots. Moreover, our curve fitting analysis corroborates these findings, showing a decrease in the probability of treatment dependence on TxGroup, with approximately 40%-50% likelihood. Although we cannot definitively assess individual improvements in each symptom class due to minor deviations, it is evident that the newly proposed treatment offers no significant enhancement in overall symptom alleviation compared to the existing medication.

2 Patient Segmentation

2.1 Pre-Analysis

Given dataset has around 39 features along with three additional features that we introuduced, making it total to 42. This is high dimensional dataset. But we can view atmost 3dimensions. So we have to abstract out all features in maximum of 3 dimensions using **dimensionality reduction**. As we are looking at baseline measurement we ignore visitDay field and as said earlier we will approximate individual scores with P_total, N_total, G_total (some sort of dimensionality reduction as we reduce from 30 dimensions to 3). Regarding considering fields like country, study etc will depend on the groups that we want to segregate data. I want to segregate data into groups to **identify the severity of initial symptoms**. So, I will ignore the other fields like Study (I am doing for all studies combined), ID's like raterId etc and, country, TxGroup due to less correlation with the measures. Now we are left with PANSS_score, P_total, N_total, G_total where we know that PANSS_score is sum of the other three and thus we ended up using P_total, N_total, G_total as our data to consider for clustering.

2.2 Segmentation

Now we figured out that we are using P_total, N_total, G_total features and classifying the initial groups of severity of Schizophrenia. Now we have to determine the value of number of clusters along with the method used for clustering. According to lecture-8 (in course), we have below approaches for clustering:

- **Prototype methods**
  - K-means
  - K-centers
  - D2-clustering
- **Statistical modeling**
  - Mixture modeling by EM algorithm
  - Model Clustering

All the plots along with code for this problem can be found here.

2.3 Statistical modeling Method - Gaussian Mixture Models (GMM)

Gaussian Mixture Models (GMM) are a probabilistic model that assumes a dataset is generated from a mixture of several Gaussian distributions. They can be fit using the expectation-maximization (EM) algorithm.

2.3.1 Deciding on value of K

As we are partitioning into K clusters we have to decide upon the optimal value of K needed. So we can use Akaike Information Criterion(AIC) and Bayesian Information Criterion(BIC) criterion as metrics for GMM.

Note Inertia or silhouette scores aren't reliable when the clusters aren't spherical or have different sizes. So we use a theoretical information criterion, such as BIC and AIC penalize models that have more parameters to learn (e.g., more clusters) and reward models that fit the data well.

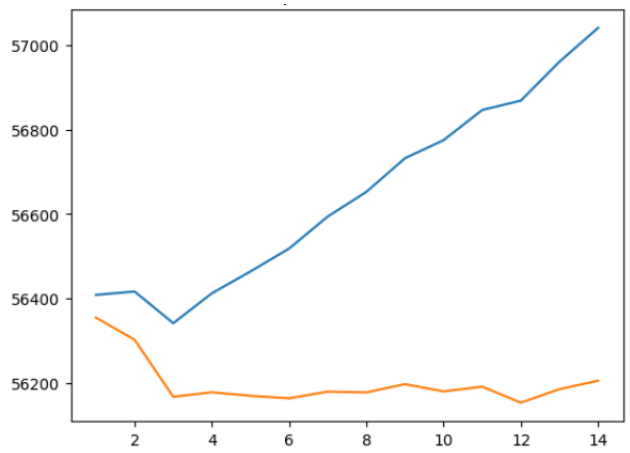

Fig - 5: AIC and BIC vs number of components (k) for GMM model

From the plots, we can clearly see that for minimum BIC we have number of components as 3 and for minimum AIC is 12 (difference is AIC of k=12 and k=3 are almost same too). Now we have conflicting between AIC and BIC criteria.

- Generally both AIC and BIC criteria give same value of k but if they contradict then we consider result with least BIC. (Reference - here) [6].
- The above is due to the fact that GMM does random initialization of inital parameters and AIC is more sensitive to initialization than BIC, so we have to opt for BIC criteria than AIC. (in case of conflicting values of k).

So we have the value of K_{opt} for GMM as 3 (using BIC). Now after plotting and labeling data using those three clusters we got the below 3D plot (3D because we chose 3 features).

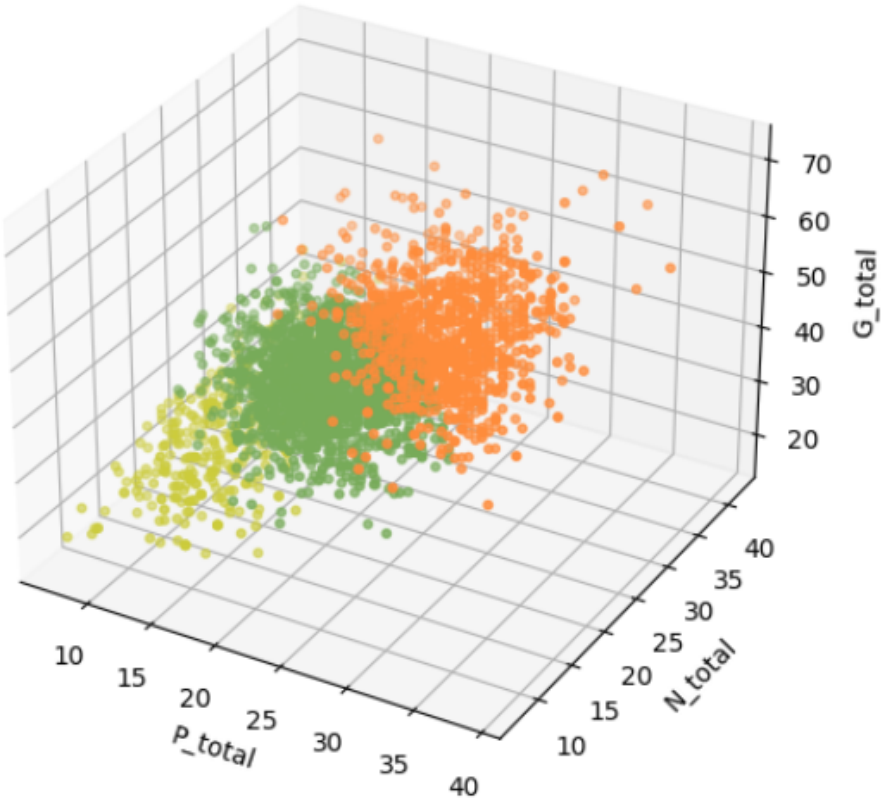

Fig -5: Figure showing 3 clusters of given data detected by GMM Model

2.4 Prototype Method - KMeans

KMeans algorithm aims to partition a set of observations into k clusters, where each observation belongs to the cluster with the nearest mean. The algorithm iteratively adjusts the cluster means until convergence.

2.4.1 Pre-processing

KMeans algorithm is sensitive to mean and variance so we have to normalize data before applying this algorithm. So we use **StandardScaler()** from scikit-learn to normalize all features of data and pass it further.

2.4.2 Deciding on value of K

As we are partitioning into K clusters we have to decide upon the optimal value of K needed. (Reference here [2]).

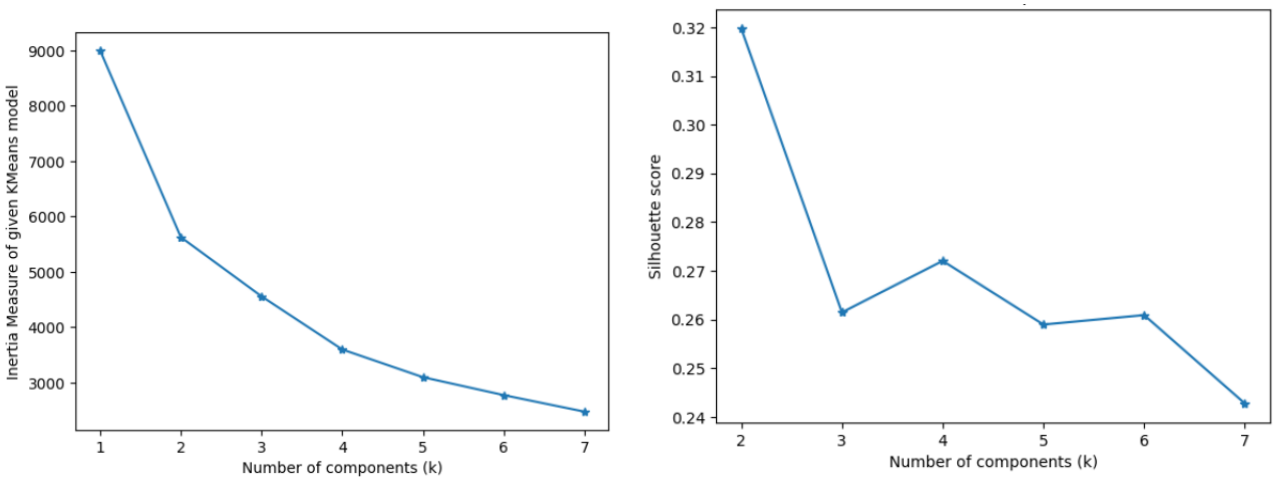

Fig - 6: (a) Inertia vs k (Elbow Method), (b) Silhouette scores vs k where k is number of components

- **Inertia plot** : For optimal value of K, we can consider metric called **inertia** which is the mean squared distance between each instance and its closest centroid. The one with lower inertia is optimal value of k, but from the above we have inertia decreasing down with value of k and thus we have to find an inflexion point called the **elbow**. Elbow is the value of lowest possible value of k where on increase in k wouldn't decrease inertia considerably (saturation phase). From the above curve using elbow method, we conclude that K_{opt} **= 4**.
- **Silhouette plot**: An instance's silhouette coefficient is equal to $\frac{(b-a)}{max(a,b)}$, where a is mean distance to other instances in same cluster (i.e., mean intra-cluster distance) and b is the mean nearest-cluster distance (i.e., the mean distance to the instances of the next closest cluster, defined as the one that minimizes b, excluding the instance's own cluster). When we take mean silhouette coefficient over all the instances we get **Silhouette score**. In general, one with higher silhouette score is better. From Fig-6 (b), we conclude that K_{opt} **= 2** for this method.

As silhouette is better than elbow method (reference [3]), considering K_{opt} **= 2** and plotting clusters, gives below,

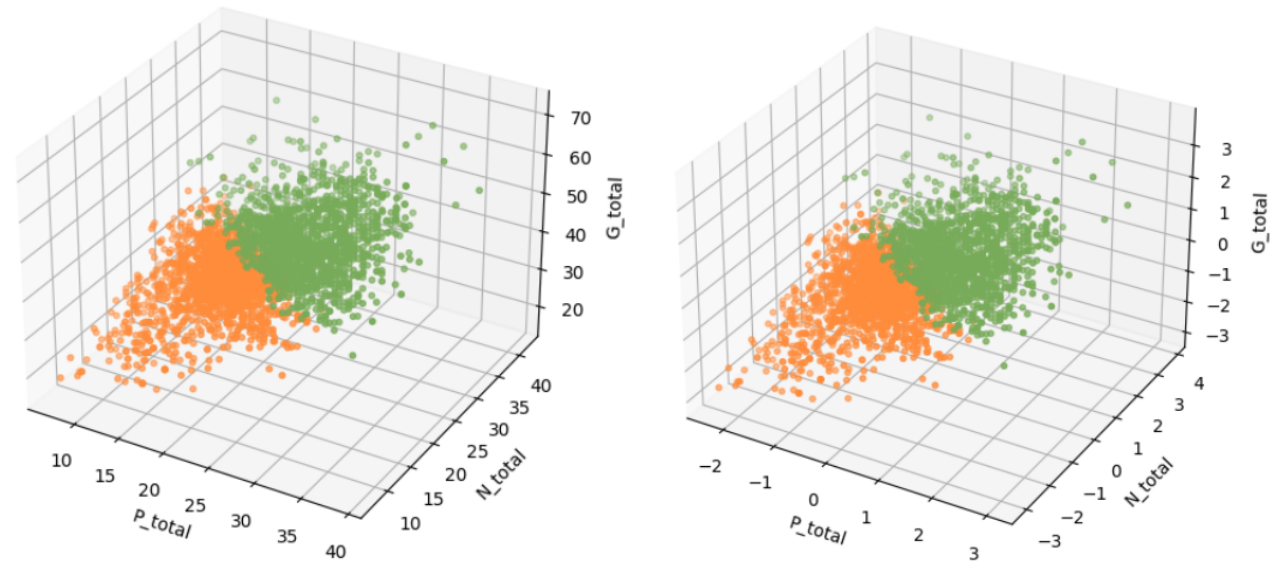

Fig -7: Clusters of data detected by KMeans (a) after feature scaling(initial values), (b) before scaling

2.5 Observations and Conclusions

For given data, GMM classified patients based on P_total, N_total, G_total into three categories (we consider BIC over AIC) whereas KMeans classified patients into 2 categories only (we took Silhouette over Elbow method). Now the main question is, **what do clusters represent?**. As we have clustered using G_total, P_total, N_total our outcome would be saying about severity of Schizophrenia on patients on 0^{th} day of their visits.

- Gaussian Mixture Model has successfully classified the patients into three categories : Orange cluster representing highly effected, green being moderately effected and yellowish-green being least effected.

For GMM	Orange cluster	Yellowish-green cluster	Green cluster
Mean of each component	$(25.57, 25.00, 48.32)$	$(13.11, 20.13, 28.61)$	$(19.45, 21.71, 39.79)$

From the above table, orange cluster is shifted towards higher measures and then green, yellowish-green clusters. This proves that our hypothesis that orange is representing higher severe patients then green(moderate effect) then yellowish-green clusters (least effect).

- Doing the same for KMeans where it predicted two classes which can hypothesized as patients suffering Schizoprenia or not (in statistical sense).

For KMeans	Orange cluster	Green cluster
Centre of cluster after feature scaling	$(17.09, 20.57, 35.79)$	$(24.94, 25.02, 47.60)$
Centre of cluster for normalized data	$(-0.62, -0.41, -0.66)$	$(0.70, 0.47, 0.75)$

From the above table, green cluster is shifted towards higher measures and then orange. This proves that our hypothesis that green is representing patients diagnosed as Schizophrenia and orange for patients possibly not having noticeable effect of Schizophrenia.

3 Forecasting

3.1 Approach

In this question, our task involves predicting the 18th-week scores of patients enrolled in Study-E by leveraging data extracted from Studies A to D. Drawing insights from the observed trends among patients in the preceding studies, we aim to forecast the PANSS score of the test subjects.

Initially, Linear Regression might appear as a common solution to this problem. However, to enhance its performance, I made modifications tailored to our specific context.

Given the extensive dataset and numerous features across all studies, a straightforward application of regression might yield inconclusive results. Recognizing the complexity of the data and the temporal aspect (where scores vary with time), a more efficient approach is necessary. Hence, I opted for the double exponential smoothing algorithm, which is well-suited for time series analysis. This algorithm allows us to consider all the data and features effectively, especially given the hints of time series application (where the score depends on days).

After selecting the appropriate method, I proceeded to train the algorithm for predicting the 18th-week scores using the information from Studies A to D.

While alternative methods such as Random Forest Regressor or Support Vector Regressor exist, I chose to evaluate the efficacy of this simpler algorithm in our specific scenario.

3.2 Data Processing and Implementation

I utilized the pandas library to import each file (A to D) as a data frame and merged them into a single large data frame named "train". Additionally, I imported another file as a data frame named "test" (E). Subsequently, both data frames were converted into numpy arrays, which offer efficiency for numerical computations.

The featureExtractor function was implemented to process an input tensor (a multidimensional array) along with a boolean flag called testSet. This function returns a numpy array containing a subset of the input tensor. If testSet is set to False, it returns all elements except the first seven and the last one. Conversely, if testSet is True, it excludes only the first seven elements.

After applying various manipulations (detailed in the code), I condensed all the data into two lists named X and Y. Then, the doubleExponentialMean function was employed, which takes a list of arrays as input along with two parameters: "a" and "gamma". This function implements a double exponential smoothing algorithm to make predictions based on the input data. It iterates over each array in the input list, initializing variables representing the predicted value and gradient (or trend). These variables are updated using a weighted average of the actual value and the previous prediction and gradient, where "a" and "gamma" are the smoothing factors. The function returns a list of predictions for each array in the input list, following the equations:

$$l_t = a * y_t + (1 - a)(l_{t-1} + b_{t-1})$$
$$b_t = \gamma(l_t - l_{t-1}) + (1 - \gamma)b_{t-1}$$
$$s_t = l_t + b_t$$

These equations form the basis of the double exponential smoothing algorithm, with "a" and "gamma" serving as smoothing parameters for the level and trend components, respectively. The term "double exponential

smoothing" denotes the utilization of two smoothing parameters, one for the level component and one for the trend component.

Subsequently, the best parameters were determined using three metrics: MSE (Mean Squared Error), MAE (Mean Absolute Error), and RMSE (Root Mean Squared Error). Although all three metrics are not always necessary, they are commonly used and computed using the functions provided in the sklearn.metrics module.

Using the calculated values of "a" and "gamma", Linear Regression was applied along with the best parameters from sklearn.linearModel to fit the values on X and Y. Subsequently, the predict method was used to forecast the values of the total PANSS score.

Given that the double Exponential method was applied to X and Y initially, followed by Linear Regression, there is a compounded effect, combining the predictions from the linear model and the double exponential algorithm based on labels. This compound effect can be observed in the code implementation.

3.3 Observations: Why Linear Regression works so well here?

In general, the Linear Regressor tends to perform well under specific conditions:

The relationship between the independent and dependent variables is linear, indicating that changes in the independent variables have a consistent effect on the dependent variable. There is no significant correlation among the independent variables, ensuring that each independent variable provides unique information to the model. The observations are independent of each other, meaning that the value of one observation does not influence the value of another observation. Moreover, it's important to ensure that the relationship between independent and dependent variables is linear, as the model assumes this relationship for accurate predictions.

Based on the observations from Questions 1 and 2: The correlation map in Figure 2 indicates that the correlations between other features are very low, which is a favorable condition for linear regression. This suggests that each feature contributes independently to the prediction, enhancing the model's accuracy. Similarly, from Figure 3, the curve fit between the mentioned features appears to be almost linear, which is again conducive to linear regression. This reinforces the suitability of linear regression for the given dataset. Hence, in this case, Linear Regression demonstrates a decent degree of accuracy, benefiting from the favorable conditions outlined above.

3.4 Conclusions and Improvement

In this particular problem, we harnessed the specific trend exhibited by the dataset to leverage the effectiveness of linear regression in conjunction with double exponential smoothing. However, it's worth noting that general methods like Support Vector Regressor and Random Forest Regressor remain viable alternatives, potentially offering improvements if optimal parameters are considered.

4 Binary classification

4.1 Approach

In this particular task, my aim is to determine the likelihood that each patient in Study-E will be flagged, as defined in the introduction. To achieve this, I utilize Studies A to D as training data and Study E as test data.

The description of being flagged or assigned to a clinical specialist entails scenarios such as patient assessments lacking coherence, assessments inconsistent with previous ratings, or an impractical trajectory of outcome assessments. To address this, clinical auditing firms are typically engaged to validate patient assessments. Assessments deemed potentially erroneous are either flagged for review or assigned to a clinical specialist for further validation and confirmation.

Given the absence of specific trends to classify patients into each group, I conducted an analysis of the data from Studies A to D, where each patient was labeled. Subsequently, I attempted to predict the probability using a trained machine learning (ML) model.

Given the challenge of identifying the exact features determining the patient's status, I included all columns from country to Lead status, as they all potentially influence the final outcome. Subsequently, I employed these columns to train my model for predicting the probability.

In essence, the core version of this problem involves classifying a patient into a particular group. This version extends the problem to provide the probability that the individual belongs to that group.

Given the classification nature of the problem, the initial consideration would be to employ Logistic Regression. However, due to the large volume of data and numerous features, I deemed this model may not yield optimal results without significant data preprocessing. Consequently, I opted to utilize a decision tree-based GradientBoostClassifier (GBC) for this task, as further elucidated below.

4.2 Why GBC?

First let me give a brief intro to GradientBoostClassifier:- GBC is a ensemble learning method which is used to optimize both regression and classification problems. (We will be focusing on classification here)

- It combines several weaker models to create a strong predictive model.
- Here I'm using the implementation based on decision trees i.e, they are the weaker model.
- The algorithm builds an additive model in a forward stage-wise fashion and allows for the optimization of arbitrary differentiable loss functions. In each stage, regression trees are fit on the negative gradient of the loss function.
- I'm using sklearn's implementation and further information can be gathered from here [4]

Now that we have some basic idea let's see it's advantages:

- Gradient boosting classifier can handle both numerical and categorical features and can perform implicit feature selection.
- Gradient boosting classifier is robust to overfitting and can achieve high accuracy with less data sensitivity
- Gradient boosting classifier can learn from the errors of previous weak learners and improve the predictions iteratively (well this applies to almost all ensemble learners but still I'm mentioning it here).
- Gradient boosting classifier can use different loss functions such as log loss to measure the performance of the mode which I used here for this task.

After all that I was convinced to use GBC on the data to which I fed all the columns after some processing discussed below.

4.3 Data Processing and Implementation

In this section, I'll outline the data processing steps undertaken and elucidate the utilization of Gradient Boosting Classifier (GBC) to predict probabilities. Please refer to the attached code for additional insights and clarification.

4.3.1 Data Preprocessing

- I gathered data from all studies containing both scores and labels, consolidating them into one array for training (Studies A to D) and another for testing (Study E).
- Extracting features and labels from these arrays, I assigned numerical values to the labels using a dictionary. Specifically, "Passed" was labeled as 0, while "Assign to CS" and "Flagged" were labeled as 1. This setup entails a binary classification task, where 0 signifies that the patient passed and 1 indicates that further attention is required.
- Additionally, I utilized a function called "Labels" to generate two lists: "sequences" and "labels." The former comprises feature arrays derived from the scores array for each patient, while the latter contains numerical labels based on the dictionary. Each patient is assigned a label based on their last row in the scores array.
- Employing another function named "preprocess," I padded each array in the "sequences" list with edge values. Subsequently, I slid a window of size 2*windowsize + 1 over each array and flattened the window into a sequence. This process was repeated for each patient to generate additional features based on neighboring scores.

4.3.2 Model Training and Implementation

- The features and labels were divided into training and testing sets using the TrainTestSplit function from the sklearn library, adhering to the commonly accepted ratio of 80-20. Additionally, a fixed random state was set to ensure consistency for further use in 5-fold cross-validation.
- Subsequently, a grid search cross-validation was conducted to identify the optimal hyperparameters for the gradient boosting classifier. This involved creating a grid search cross-validation object, which takes as inputs the classifier, a parameter grid, the number of folds, and the number of parallel jobs. The object then exhaustively explores all possible combinations of parameters within the grid and assesses each combination's performance using cross-validation on the training set. The best parameters and corresponding score, based on accuracy, are returned by the object.

- Due to the substantial size of the dataset, particularly after processing, the grid search was initially performed solely on Study A. The parameters identified as optimal during this process, specifically 'learningrate': 0.01, 'maxdepth': 4, 'nestimators':500, were subsequently utilized for training the model on the remaining studies.
- The gradient boosting classifier was then trained on the training set using the best parameters determined by the grid search. Evaluation was conducted on both the training and testing sets across all studies, employing log loss as the metric. The classifier generates predictions by aggregating weighted votes from all the trees in the ensemble.

4.4 Conclusions and Improvement

So we can see that in the age of ML and Data Processing we use them for predicting the status without actually seeing the patient though it has some errors but we can get a rough idea on where and which patient we have to focus on. Though this is not the only method but one can always use different models and also can be estimated purely using statistics.

5 Codes

Please find the github repositories for the above here:
Codes : Here

References

[1] João Paulo Figueira. "LOESS". In: (2019).

[2] Aurélien Géron. In: *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd Edition, 9th Chapter.

[3] Adria Binte Habib. *Elbow Method vs Silhouette Co-efficient in Determining the Number of Clusters*. MLearning.ai.

[4] *Scikit-learn*.

[5] Jonathan Taylor Josef Perktold Skipper Seabold. *statsmodels 0.13.5*.

[6] Jake Vander Plas. "In Depth: Gaussian Mixture Models". In: *Python Data Science Handbook*.